INTERPRETING IDR AS A PETROV-GALERKIN METHOD*

VALERIA SIMONCINI† AND DANIEL B. SZYLD‡

Abstract. The IDR method of Sonneveld and van Gijzen [*SIAM J. Sci. Comput.*, 31:1035–1062, 2008] is shown to be a Petrov-Galerkin (projection) method with a particular choice of left Krylov subspaces; these left subspaces are rational Krylov spaces. Consequently, other methods, such as BiCGStab and ML(s)BiCGStab, which are mathematically equivalent to some versions of IDR, can also be interpreted as Petrov-Galerkin methods. The connection with rational Krylov spaces inspired a new version of IDR, called Ritz-IDR, where the poles of the rational function are chosen as certain Ritz values. Experiments are presented illustrating the effectiveness of this new version.

1. Introduction. Recently, Sonneveld and van Gijzen [23] brought renewed interest in the Induced Reduction Method (IDR) first mentioned in [28] thirty years ago. In [23], and in the subsequent publications [21], [22], and [27], the authors discuss the IDR method as a completely new family of Krylov subspace methods with short recurrences; see also [7]. In this contribution, we describe the IDR class of methods as a classical Krylov subspace method satisfying a Petrov-Galerkin condition using an appropriately chosen left subspace. These left subspaces are rational Krylov spaces. The connection with rational Krylov spaces inspired us to look a new possible values for the poles of the defining rational functions. We present a new version of IDR, where the poles of the rational function are chosen as certain Ritz values. We call this new version Ritz-IDR, and we present several experiments illustrating its effectiveness.

The IDR method as presented in [21], [23] appears to be very competitive in a host of problems. This is due in part to the way the IDR method is implemented. In the interpretation given in this article, we do not deal with implementation issues; we merely discuss some theory behind this class of methods.

In the next section, we briefly describe the classical Petrov-Galerkin condition in the context of Krylov subspace methods; see further, e.g., [14], [16]. In Section 3 we present and analyze a set of nested subspaces, which we use as the left spaces for our interpretation of IDR(s) as a Petrov-Galerkin method. We relate these sets to the results on IDR in [21], [22], [23], and [27], in Section 4 and we show how some of the results in these references follow from the interpretation of IDR as a Petrov-Galerkin method. In section 5, we extend our analysis to the more general class of IDR(s, ℓ) methods.

Since both BiCGStab [25] and ML(s)BiCGStab [29] are mathematically equivalent to certain versions of IDR(s)(see [7], [21], [23]) our interpretation of the IDR class of methods as Petrov-Galerkin methods allows as to conclude that BiCGStab and ML(s)BiCGStab are also Petrov-Galerkin methods.

In Section 6 we describe the new version Ritz-IDR, and in Section 7 we present our numerical experiments illustrating the effectiveness of the proposed new version.

2. Petrov-Galerkin methods. The goal of the IDR method studied in this article is to solve $n \times n$ large sparse nonsymmetric linear systems of equations of the form

$$Ax = b. \tag{2.1}$$

*This version dated 10 April 2010

†Dipartimento di Matematica, Università di Bologna, Piazza di Porta S. Donato, 5, I-40127 Bologna, Italy; and CIRSA, Ravenna (valeria@dm.unibo.it).

‡Department of Mathematics, Temple University (038-16), 1805 N. Broad Street, Philadelphia, Pennsylvania 19122-6094, USA (szyld@temple.edu).

Given an initial vector x_0, and the corresponding initial residual $r_0 = b - Ax_0$, Krylov subspace methods are defined by choosing at the mth iteration an approximation x_m such that $x_m - x_0$ lies in the Krylov subspace (of dimension m) defined by

$$\mathcal{K}_m = \mathcal{K}_m(A, r_0) = \text{span}\{r_0, Ar_0, A^2r_0, \ldots, A^{m-1}r_0\}. \tag{2.2}$$

Thus, the residual $r_m = b - Ax_m = r_0 + A(x_m - x_0)$ lies in the next Krylov subspace space $\mathcal{K}_{m+1}$. Note that these spaces are nested, i.e., $\mathcal{K}_m \subset \mathcal{K}_{m+1}$. Unless there is a polynomial p of degree less than n such that $p(A)r_0 = 0$, this sequence of subspaces keeps growing until $\mathcal{K}_n = \mathbb{C}^n$.

Given a set $\{\mathcal{L}_m\}$ of nested subspaces, the Petrov-Galerkin condition consists of imposing that the residual r_m be orthogonal to the subspace $\mathcal{L}_m$, that is finding the only element $x \in x_0 + \mathcal{K}_m$ such that

$$b - Ax \perp \mathcal{L}_m. \tag{2.3}$$

Different choices of the subspaces $\mathcal{L}_m$ give rise to different iterative methods. For example, when $\mathcal{L}_m = \mathcal{K}_m$, one obtains the Full Orthogonalization Method (FOM), which reduces to the Conjugate Gradient method (CG) when A is symmetric and positive definite. Similarly, when $\mathcal{L}_m = A\mathcal{K}_m$, one has a minimal residual method, such as GMRES. When $\mathcal{L}_m = \mathcal{K}_m(A^*, \tilde{r}_0)$, for some initial (left) residual $\tilde{r}_0$, one has BiCG. See, e.g., [14], [16], for descriptions of these methods, and references of the original papers presenting them.

3. Analysis of the left spaces. We begin with some notation and definitions. Let $\sigma(A)$, $\mathcal{N}(A)$, and $\mathcal{R}(A)$ denote the spectrum, the null space, and the range of A, respectively. Let A^* denote the conjugate transpose of A. Let $P = [p_1, p_2, \ldots, p_s] \in \mathbb{C}^{n\times s}$. For every $j = 1, 2, \ldots$, define the *block Krylov subspace*

$$\mathcal{K}_j(A^*, P) = \mathcal{R}([P, A^*P, \ldots, (A^*)^{j-1}P]) = \sum_{i=1}^{s} \mathcal{K}_j(A^*, p_i). \tag{3.1}$$

Consider the polynomial

$$\Omega_j(t) = (1 - \omega_j t)\cdots(1 - \omega_1 t), \quad \omega_i \neq 0,\ i = 1, \ldots, j, \tag{3.2}$$

of degree j, and let $\Omega_0(t) = 1$. If $1/\omega_i \notin \sigma(A)$, $i = 1, \ldots, j$, then the matrix $\Omega_j(A)^*$ is invertible, and for $j \geq 1$, $(\Omega_j(A)^*)^{-1} = (\Omega_{j-1}(A)^*)^{-1}((I - \omega_j A)^*)^{-1}$. We construct now the sequence of subspaces

$$\mathcal{W}_j = (\Omega_j(A)^*)^{-1}\mathcal{K}_j(A^*, P), \tag{3.3}$$

where the scalars ω_i are appropriately chosen.

We claim that the $\mathcal{W}_j$ spaces are the left spaces of IDR when seen as a Petrov-Galerkin method, and this is shown in Section 4. The following lemma summarizes some facts about these spaces.

LEMMA 3.1. *Assume that $\Omega_j(A)^*$ is nonsingular. Then*

(i) $\mathcal{W}_j = \mathcal{K}_j(A^*, (\Omega_j(A)^*)^{-1}P)$;

(ii) $d_j := \dim \mathcal{W}_j = \dim \mathcal{K}_j(A^*, P)$; *analogously, from* $\text{rank}\, P = s$, *it follows that* $\text{rank}(\Omega_j(A)^*)^{-1}P = s$.

(iii) $\mathcal{W}_{j+1} = (\Omega_{j+1}(A)^*)^{-1}\mathcal{K}_j(A^*, P) + \mathcal{R}(((I - \omega_{j+1}A)^*)^{-1}P)$.

(iv) $\mathcal{W}_j = \mathcal{R}([(\Omega_j(A)^*)^{-1}P, (\Omega_{j-1}(A)^*)^{-1}P, \ldots, ((I-\omega_1 A)^*)^{-1}P])$.

Proof. The relation in (i) follows from the fact that A^* commutes with any rational function in A^*. The relation in (ii) follows by considering a basis of $\mathcal{K}_j(A^*, P)$, and noticing that for $\Omega_j(A)^*$ nonsingular, the matrix $(\Omega_j(A)^*)^{-1}$ maps linearly independent vectors onto linearly independent vectors - a basis of $\mathcal{W}_j$. The same argument can be used to see that $\operatorname{rank}(\Omega_j(A)^*)^{-1}P = s$.

To show (iii) we conveniently write the space $\mathcal{K}_j(A^*, P)$ as

$$\mathcal{K}_j(A^*, P) = \mathcal{R}([P, (I-\omega_1 A)^* P, (I-\omega_1 A)^*(I-\omega_2 A)^* P, \ldots, \prod_{k=1}^{j-1}(I-\omega_k A)^* P]),$$

so that

$$(\Omega_{j+1}(A)^*)^{-1}\mathcal{K}_{j+1}(A^*, P) = \mathcal{R}([(\Omega_{j+1}(A)^*)^{-1}P, \prod_{k=2}^{j+1}((I-\omega_k A)^*)^{-1}P, \ldots, ((I-\omega_{j+1}A)^*)^{-1}P]).$$

In addition

$$(\Omega_{j+1}(A)^*)^{-1}\mathcal{K}_j(A^*, P) = \mathcal{R}([(\Omega_{j+1}(A)^*)^{-1}P, \prod_{k=2}^{j+1}((I-\omega_k A)^*)^{-1}P, \ldots, ((I-\omega_j A)^*)^{-1}((I-\omega_{j+1}A)^*)^{-1}P]).$$

Comparing the last two displayed formulas the result follows.

Similarly, for (iv), we can write $\mathcal{K}_1(A^*, P) = \mathcal{R}(P)$ and for $j \geq 2$

$$\mathcal{K}_j(A^*, P) = \mathcal{R}([P, (I-\omega_j A)^* P, (I-\omega_j A)^*(I-\omega_{j-1} A)^* P, \ldots, \prod_{k=2}^{j}(I-\omega_k A)^* P]),$$

and computing $(\Omega_j(A)^*)^{-1}\mathcal{K}_j(A^*, P)$ the expression in (iv) follows directly. ∎

We are ready to prove that the sets $\mathcal{W}_j$ are nested.

THEOREM 3.2. *Let $A \in \mathbb{R}^{n\times n}$ be nonsingular. Let $\Omega_j(t)$ be a polynomial of degree j of the form* (3.2)*, with $1/\omega_i \notin \sigma(A)$, $i = 1, \ldots, j$. Let $P = [p_1, p_2, \ldots, p_s] \in \mathbb{R}^{n\times s}$ be of rank s, such that*

$$\mathcal{K}_j(A^*, P) \subsetneq \mathcal{K}_{j+1}(A^*, P), \tag{3.4}$$

unless $\mathcal{K}_j(A^*, P) = \mathbb{R}^n$. Consider the sets $\mathcal{W}_j$ defined in (3.3)*. Then, for each $j \geq 1$, we have that $\mathcal{W}_j \subset \mathcal{W}_{j+1}$, and this inclusion is strict unless $\mathcal{W}_j = \mathbb{C}^n$. Furthermore,*

$$d_{j+1} - d_j \leq d_j - d_{j-1} \leq s. \tag{3.5}$$

Proof. Using Lemma 3.1(iv) for j and $j+1$ it follows that $\mathcal{W}_j \subset \mathcal{W}_{j+1}$. The fact that this inclusion is strict (unless $\mathcal{W}_j = \mathbb{C}^n$) follows from the first equality of Lemma 3.1(ii) and the hypothesis (3.4), which implies $d_j < d_{j+1}$, except when $\mathcal{K}_j(A^*, P) = \mathbb{C}^n$, in which case $\mathcal{W}_j = (\Omega_j(A)^*)^{-1}\mathcal{K}_j(A^*, P) = \mathbb{C}^n$.

To complete the proof, we use the formulation given by Lemma 3.1(iv) and the fact that $\dim \mathcal{R}([(\Omega_{j+1}(A)^*)^{-1}P]) = s$. Therefore we have that

$$\dim[(\Omega_{j+1}(A)^*)^{-1}\mathcal{K}_{j+1}(A^*, P)] \leq \dim[(\Omega_j(A)^*)^{-1}\mathcal{K}_j(A^*, P)] + s,$$

from which the second inequality in (3.5) follows. The first inequality in (3.5) stems from the fact that once some vectors are collinear in $\mathcal{W}_j$, they continue to be collinear in $\mathcal{W}_{j+1}$. □

The hypothesis (3.4) ensures that P is chosen so that the columns of $(A^*)^j P$ do not become linearly dependent with respect to powers of A^* less or equal to j, so that the block Krylov subspaces (3.1) keep growing with j. In block Krylov subspace methods, one needs to monitor the space for the possible emergence of this collinearity; see, e.g., [6], [15] and references therein. Of course, any condition that implies (3.4) can be used as hypothesis. For example, it would suffice to require that none of the vectors p_i, $i = 1, \ldots, s$, lie in any proper invariant subspace of A^*, cf. the remark after Theorem 2.1 in [23]. We mention that there are practical ways to deal with this issue of collinearity if it occurs, and this is taken into account in the description of IDR, e.g., in [23].

Our characterization given in Lemma 3.1(iv) emphasizes that the left subspace $\mathcal{W}_j$ (sometimes referred to as the space of "shadow residuals") is in fact a *rational* Krylov subspace. These subspaces have been successfully used in the solution of eigenvalue problems and other related problems; see, e.g., [1] and references therein. We will use this fact in our motivation of the new version of IDR in Section 6.

4. IDR viewed as Petrov-Galerkin. The original description of the IDR method [28], and its recent resurgence IDR(s) [23] is given in terms of shrinking subspaces $\mathcal{G}_j$, $j = 0, 1, \ldots$ The residuals r_m are chosen in $\mathcal{K}_{m+1}(A, r_0)$ (see, e.g., (3.1) in [23]), and in addition, such that

$$r_m \in \mathcal{G}_j, \text{ with } j = \lfloor m/(s+1) \rfloor. \tag{4.1}$$

Note that every $(s+1)$ iterations the space where these residuals reside changes. The first space $\mathcal{G}_0$ is large, and it is suggested in [23] and elsewhere to be chosen as a maximal Krylov subspace $\mathcal{G}_0 = \mathcal{K}_n(A, r_0)$. A full rank $n \times s$ matrix P is given, and the subsequent spaces defined as

$$\mathcal{G}_j = (I - \omega_j A)(\mathcal{G}_{j-1} \cap \mathcal{N}(P^*)).$$

These spaces are shown to be nested, and their dimension is decreasing, and this dimension reduction gives the name of the method: Induced Reduction Method.

We show now that the requirement that (4.1) holds is equivalent to the following Petrov-Galerkin condition

$$r_m \perp \mathcal{W}_j, \text{ where } j = \lfloor m/(s+1) \rfloor. \tag{4.2}$$

To that end, we first note that one can derive an alternative definition of the spaces $\mathcal{G}_j$, as done, e.g., in [21, Theorem 4.2], where it is shown that

$$\mathcal{G}_j = \{\Omega_j(A)v \mid v \perp \mathcal{K}_j(A^*, P)\}, \tag{4.3}$$

or, what is the same

$$\mathcal{G}_j = \Omega_j(A)[\mathcal{K}_j(A^*, P)]^{\perp}; \tag{4.4}$$

see also [7].

THEOREM 4.1. *It holds that $\mathcal{G}_j^{\perp} = \mathcal{W}_j$, that is, $r \perp \mathcal{W}_j$ if and only if $r \in \mathcal{G}_j$.*

Proof. Let $B = \Omega_j(A)$ and $\mathcal{S} = K_j(A^*, P)^\perp$. Since $B\mathcal{S} = \{Bv \mid v \in \mathcal{S}\}$, we can write

$$\begin{aligned}\mathcal{G}_j^\perp \overset{(4.4)}{=} (B\mathcal{S})^\perp &= \{w \mid w^*Bv = 0, v \in \mathcal{S}\} \\ &= \{w \mid (B^*w)^*v = 0, v \in \mathcal{S}\} \\ &= \{B^{-*}y \mid y^*v = 0, v \in \mathcal{S}\} \\ &= B^{-*}\{y \mid y^*v = 0, v \in \mathcal{S}\} = B^{-*}\mathcal{S}^\perp \overset{(3.3)}{=} \mathcal{W}_j. \quad \square\end{aligned}$$

Theorem 4.1 provides a proof that the IDR(s) method of Sonneveld and van Gijzen is indeed a Petrov-Galerkin method. Some of the results we have shown in Section 3 have counterparts in the IDR literature. Our Theorem 3.2 corresponds to the "IDR Theorem" given, e.g., in [23, Theorem 3.1] or [21, Theorem 2.2]. Our observation in Lemma 3.1(ii) corresponds to [21, Corollary 4.5].

In this context, the IDR(s) method consists of finding at the mth iteration, an approximation $x_m \in x_0 + \mathcal{K}_m(A, r_0)$ such that for $m > s$ the residual $r_m = b - Ax_m$ satisfies (4.2). To complete the formulation of the IDR(s) method, one needs to provide a choice of the first s iterates, and a choice of the scalars ω_j, $j = 1, \ldots$. For the first s iterates it suffices to compute $\{x_1, \ldots, x_s\}$ by any iterative method: for example, s steps of GCR are suggested in [21], whereas s Chebyshev iterations are used in [27].

The condition (4.2) suggests that a new scalar ω_{j+1} should be computed every $s+1$ iterations, so as to complete a new polynomial Ω_{j+1}. Following [25], in [23] this choice is given by

$$\omega_{j+1} = \operatorname{argmin}_{\omega \in \mathbb{R}} \|r_m - \omega A r_m\|, \quad \text{with} \quad m = j(s+1) + s. \tag{4.5}$$

Once this ω_{j+1} is computed, a new space $\mathcal{W}_{j+1} = (\Omega_{j+1}(A)^*)^{-1}\mathcal{K}_{j+1}(A^*, P)$ can be used so that

$$r_{(j+1)(s+1)+k} \perp \mathcal{W}_{j+1}, \quad \text{for } k = 0, \ldots, s. \tag{4.6}$$

Since $r_m \in \mathcal{K}_{m+1}(A, r_0)$, it follows that $w := r_m - \omega_{j+1}Ar_m \in \mathcal{K}_{m+2}(A, r_0)$. Unless additional hypotheses are imposed, w does not necessarily satisfy the optimality condition (4.6). To comply with this requirement, the original IDR method in [23] builds the next residual as $w := v - \omega_{j+1}Av$, where v is obtained by making the residual r_m orthogonal to P, in the direction of $\mathcal{G}_j$. The s subsequent residuals do satisfy (4.6).

On the other hand, the implementation proposed in [27] requires that at each cycle j the residual vectors satisfy the additional hypothesis

$$r_{j(s+1)+k} \perp p_i, \quad i = 1, \ldots, k, \; k = 0, \ldots, s;$$

see [27, (13)]. For $k = 0$ the residual $r_{j(s+1)}$ is not enforced to be orthogonal to any of the vectors p_i, whereas for $k = s$ it holds that $r_{j(s+1)+s} \perp \mathcal{R}(P)$. This latter condition is sufficient to fully satisfy orthogonality with respect to $\mathcal{W}_{j+1}$. We formalize this result in the following proposition.

PROPOSITION 4.2. *For $m = j(s+1) + s$, $j \geq 0$, assume that $r_m \perp \mathcal{R}(P)$. Then for any ω_{j+1} such that $I - \omega_{j+1}A$ is nonsingular, $r_{m+1} = r_m - \omega_{j+1}Ar_m$ satisfies $r_{m+1} \perp \mathcal{W}_{j+1}$.*

Proof. For $j = 0$ and using $r_s \perp \mathcal{R}(P)$ and $r_{s+1} = (I - \omega_1 A)r_s$, we obtain $0 = r_s^* P = r_{s+1}^*((I-\omega_1 A)^*)^{-1}P$, that is, $r_{s+1} \perp \mathcal{W}_1$. For $j \geq 1$ we can use $r_{j(s+1)+s} \perp \mathcal{W}_j$ and write

$$0 = r_{j(s+1)+s}^* \mathcal{W}_j = r_{j(s+1)+s+1}^*((I - \omega_{j+1}A)^*)^{-1}\mathcal{W}_j,$$

that is, $r_{j(s+1)+s+1} \perp (\Omega_{j+1}(A)^*)^{-1}\mathcal{K}_j$. Recall that

$$0 = r_{j(s+1)+s}^* P = r_{j(s+1)+s+1}^*((I - \omega_{j+1}A)^*)^{-1}P,$$

so that $r_{j(s+1)+s+1} \perp \mathcal{R}(((I - \omega_{j+1}A)^*)^{-1}P)$. Using Lemma 3.1(iii) we have that $(\Omega_{j+1}(A)^*)^{-1}\mathcal{K}_j + \mathcal{R}(((I - \omega_{j+1}A)^*)^{-1}P) = \mathcal{W}_{j+1}$ so that $r_{j(s+1)+s+1} \perp \mathcal{W}_{j+1}$. ∎

Under the condition $r_m \perp \mathcal{R}(P)$, the result of Proposition 4.2 together with (4.6) imply that all generated residuals $r_{j(s+1)+k}$, $k = 0, \ldots, s$ are orthogonal to the shadow spaces $\mathcal{W}_j$, completing the requirement of a standard Petrov-Galerkin procedure.

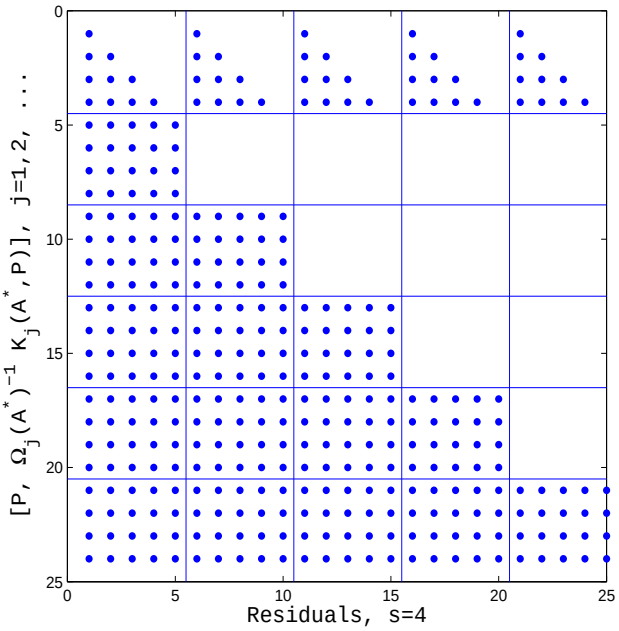

FIG. 4.1. *Sparsity pattern of the matrix* $[P, \mathcal{W}_j]^*[r_0, r_1, \ldots]$ *for* $s = 4$.

A summary of the orthogonality properties of the computed vectors under the hypotheses of Proposition 4.2 is sketched in Figure 4.1, where the sparsity pattern of the matrix $[P, \mathcal{W}_j]^*[r_0, r_1, \ldots]$ for $s = 4$ is reported; the plot was obtained with the IDR code available in the IDR website associated with [27].

In [7], [23] it is shown that both BiCGStab [25] and ML(s)BiCGStab [29] are mathematically equivalent to some versions of IDR. More precisely, the iterands from BiCGStab and ML(1)BiCGStab coincide with those of IDR(1) every other step. Therefore, the equivalence given in Theorem 4.1 implies that BiCGStab and ML(s)BiCGStab are also Petrov-Galerkin methods, with the rational Krylov subspaces $\mathcal{W}_j$ as left subspaces.

5. Discussion on IDR(s,ℓ). It was noted with the BiCGStab method, that the use of polynomials of the form (3.2) works well for matrices with real spectra. Indeed, let $r_m = p_m(A)r_0$, where $p_m(t) = \Omega_j(t)q(t)$ is the residual polynomial. If the roots of Ω_j approximate some eigenvalues of A, the residual polynomial in A applied to r_0 gives a vector with small components in the direction of the corresponding eigenvectors. But if the matrix has complex spectrum, it is very unlikely that a polynomial with real roots in A can accomplish this; see, e.g., [5], [17]. Thus, in these two last references, it was suggested to use a polynomial $\Omega(t)$ with complex roots. In BiCGStab(ℓ) [17], the residual polynomial is $p_m(t) = \Omega_{j,\ell}(t)q(t)$ with

$$\Omega_{j,\ell}(t) = \phi_j(t)\phi_{j-1}(t)\dots\phi_1(t), \tag{5.1}$$

and each $\phi_k(t)$ is a polynomial of degree ℓ with $\phi_k(0) = 1$, $k = 1,\dots,j$; cf. (3.2), with $\ell = 1$. The polynomial $\phi_j(t)$ is chosen so that the next residual has minimum norm; cf. (4.5) for $\ell = 1$. Let μ_i be the roots of $\Omega_{j,\ell}$, $i = 1,\dots,j\ell$. In the following we assume that $\mu_i \notin \sigma(A)$, $i = 1,\dots,j\ell$, so that the matrix $\Omega_{j,\ell}(A)$ is nonsingular.

Taking these properties of BiCGStab(ℓ) into account, one can similarly extend the IDR(s) method to a Petrov-Galerkin method with the left block subspaces being

$$\mathcal{W}_{j,\ell} := (\Omega_{j,\ell}(A)^*)^{-1}\mathcal{K}_{j\ell}(A^*, P). \tag{5.2}$$

It would then be natural to call this, the IDR(s,ℓ) method. This method is the Petrov-Galerkin description of the IDRStab method presented by Sleijpen and van Gijzen in [22], where the sets $\mathcal{G}_j^{\perp}$ given there are in fact identical to $\mathcal{W}_{j,\ell}^{\perp}$; this follows from Theorem 4.1. See also the related GBi-CGSTAB(s,L) method presented by Tanio and Sugihara in [24].

We do not pursue further this IDR(s,ℓ) method, except to present some properties of the sets (5.2). Let $d_{j,\ell} := \dim \mathcal{W}_{j,\ell}$. Using the same argument as in Lemma 3.1 (ii) it follows that $d_{j,\ell} = \dim \mathcal{K}_{j\ell}(A^*, P)$. The next result is an "IDR Theorem" for the sets (5.2), i.e., for the IDR(s,ℓ) method, and it follows in the same manner as Theorem 3.2 using the following identity

$$\mathcal{K}_{j\ell}(A^*, P) = \mathcal{R}([P, \phi_j(A)^*P, \phi_j(A)^*\phi_{j-1}(A)^*P, \dots, \phi_j(A)^*\cdots\phi_2(A)^*P]).$$

THEOREM 5.1. *Let $A \in \mathbb{R}^{n\times n}$ be nonsingular. Let $\Omega_{j,\ell}(t)$ be a polynomial of degree $j\ell$ of the form* (5.1)*, with roots μ_i such that in $\mu_i \notin \sigma(A)$, $i = 1,\dots,j\ell$. Let $P = [p_1, p_2, \dots, p_s] \in \mathbb{R}^{n\times s}$ be full rank, such that $\mathcal{K}_k(A^*, P) \subsetneq \mathcal{K}_{k+1}(A^*, P)$,unless $\mathcal{K}_k(A^*, P) = \mathbb{R}^n$, $k = 1,\dots$. Consider the sets $\mathcal{W}_{j,\ell}$ defined in* (5.2)*. Then, for each $j \geq 1$, we have that $\mathcal{W}_{j,\ell} \subset \mathcal{W}_{j+1,\ell}$, and this inclusion is strict unless $\mathcal{W}_{j,\ell} = \mathbb{C}^n$. Furthermore, $d_{j+1,\ell} - d_{j,\ell} \leq d_{j,\ell} - d_{j-1,\ell} \leq s + \ell$.*

6. Ritz-IDR: a new version. The introduction of the rational Krylov subspace $\mathcal{W}_j$ suggests that the poles, the reciprocals of the parameters ω_j could be chosen following more general strategies than the local minimization employed in (4.5). First introduced for approximating interior eigenpairs [12], [13], rational Krylov spaces have significantly increased their popularity in the past decades, showing their effectiveness in a variety of applications far beyond eigenvalue approximation; see, e.g., [2], [3], [4]. The a-priori selection of poles, however, remains a critical step in the performance of projection-type methods based on rational Krylov subspaces. In addition, the approximation problem usually guides such a choice. In our setting, the space

$$\mathcal{W}_j = \mathcal{R}([((I - \omega_1 A)^*)^{-1}P, \dots, (\Omega_j(A)^*)^{-1}P, (\Omega_{j-1}(A)^*)^{-1}P])$$

is the test space, and not the approximation space; nonetheless, the choice of the parameters ω_j may be related to the spectral properties of A. Indeed, intuitively speaking, if the poles are close to some of the eigenvalues of A^*, and P is sufficiently general, then $\mathcal{W}_j$ will be rich in the corresponding eigencomponents. Making the residual orthogonal to such a space will eventually dampen eigenvector (or invariant subspace) components. The IDR method can thus benefit from an explicit inclusion of spectral information in the test space, whenever this is not naturally obtained with the computed ω's. Indeed, for some real vector v, the parameters in (4.5) are obtained as

$$\omega_j = \frac{v^*Av}{(Av)^*(Av)} = \frac{(A^{-1}w)^*w}{w^*w}, \quad w = Av$$

and thus, the $\omega_j \in \mathbb{R}$ belong to the field of values $W(A^{-*})$ of A^{-*}, which is the same as $W(A^{-1})$ for real A. In general, however, $W(A) \neq (W(A^{-1}))^{-1}$, and in fact the two convex sets may differ substantially, therefore it may be possible that the reciprocals $1/\omega_j$ do not even belong to $W(A)$; we refer to [9] and references therein for a characterization of $W(A^{-1})$. Therefore, for problems where $W(A^{-1})^{-1}$ significantly differs from $W(A)$, more effective ω_j may be chosen. We mention that strategies to switch to some values of ω_j other than (4.5) are given in [18], [19] for BiCGStab-related methods, and that the switching idea is implemented in the IDR version reported in [23], but this implementation is not publicly available.

Here we propose to substitute the parameters ω_j with the Ritz values obtained by a preliminary generation of a small Krylov subspace of fixed dimension m_0. The Ritz values are then applied cyclically as the IDR iteration proceeds. Our numerical experience suggested not to take all Ritz values, but only a significant portion of them, corresponding to the largest (in magnitude) 75%. For indefinite problems, better results were observed with Leja (as opposed to decreasing) ordering; see, e.g., [11] for a description of the Leja ordering. We call the resulting method Ritz-IDR.

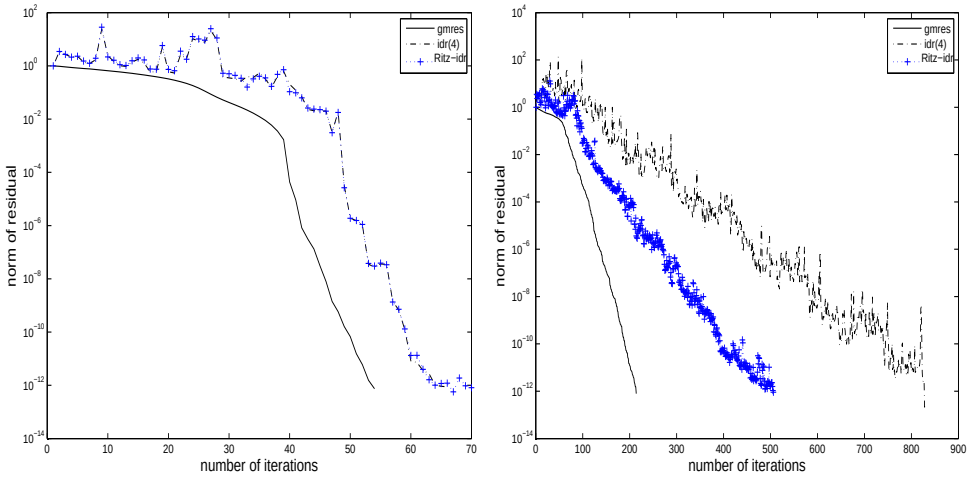

FIG. 6.1. *Convergence history for the operator* $L(u) = -\Delta u + \beta(u_x + u_y)$ *for various methods and* $\beta = 50$ *(left) and* $\beta = 1000$ *(right).*

The following two examples visually describe the discussed phenomenon. We consider the 400×400 matrix stemming from the finite difference discretization of the

operator $L(u) = -\Delta u + \beta(u_x + u_y)$ on the unit square. The right-hand size is the vector of all ones. The two plots of Figure 6.1 show the convergence performance of GMRES, standard IDR(4) and Ritz-IDR for $\beta = 50$ (left) and $\beta = 1000$ (right) using the largest 15 Ritz values (in magnitude) of the 20 computed; $s = 4$ is used in Ritz-IDR. The IDR performance highly deteriorates for large β. It is also remarkable that for $\beta = 50$, the convergence curves of IDR and Ritz-IDR are almost indistinguishable.

The next plots display the different spectral settings associated with the two considered values of β. Figure 6.2 shows the field of values[1] of A and A^{-1} for $\beta = 50$ (left) and $\beta = 1000$ (right), together with the Ritz values obtained with a Krylov subspace of dimension $m_0 = 20$ (circles) and used in Ritz-IDR, and the values ω_j^{-1} (asterisks) computed by the IDR method in the runs shown in Figure 6.1. The crosses are the eigenvalues of A. While for moderate β the values of ω_j^{-1} are contained in $W(A)$, for $\beta = 1000$ they fall far away from $W(A)$, providing no benefit to the method. The similar location of the Ritz and IDR values for $\beta = 50$ also appears to justify the very close convergence behavior of Ritz-IDR and IDR in this case. We mention in passing that the fact that $W(A)$ is larger than the convex hull of the eigenvalues in Figure 6.2 stems from the fact that the matrices are non-normal.

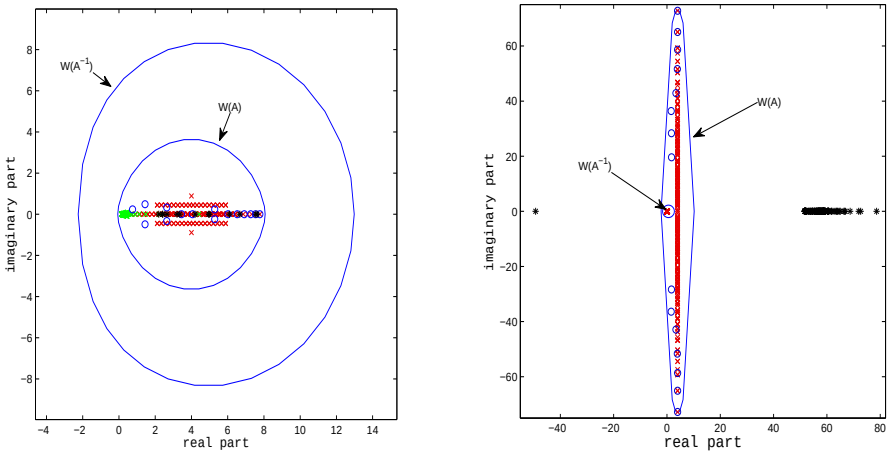

FIG. 6.2. *Spectral information for operator* $L(u) = -\Delta u + \beta(u_x + u_y)$ *with* $\beta = 50$ *(left) and* $\beta = 1000$ *(right).*

In terms of storage, the new Ritz-IDR version needs m_0 long vectors of storage. In our experiments with $m_0 = 20$, this is not much more than the 16 vectors needed for IDR(s) for $s = 4$. On the other hand, our implementation of Ritz-IDR uses complex arithmetic, which implies that memory allocations are doubled.

7. Numerical experiments. In this section we report on our numerical experience with IDR(s) and its variant Ritz-IDR presented in the previous section. Our implementation is based on the code available on the IDR site [26]. To evaluate the optimality of the short-term recurrence, we also include results with full GMRES, which is otherwise too expensive, both in terms of CPU time and memory requirements, in the examples shown. In some cases, the performance of restarted GMRES

[1] All fields of values were approximated numerically with the function `fv.m` of the Test Matrix Toolbox for Matlab [8].

is also discussed, as the current method of choice in many application problems. This experimental setting significantly differs from those reported in the IDR literature, where the major competitor is BiCGStab(ℓ). We mention that no comparisons are presented with IDRStab [22] or other IDR versions since those implementations are not publicly available.

Unless otherwise stated, the new variant uses $s = 4$, which is therefore no longer a parameter to be set. Except for Example 7.5, the parameters ω's in Ritz-IDR were obtained as $\omega_j = \theta_j^{-1}$, where θ_j were taken to be the 15 largest (in absolute values) eigenvalues of the matrix $V_{m_0}^* A V_{m_0}$, where the columns of V_{m_0} stored the orthonormal basis after $m_0 = 20$ Arnoldi iterations. We remark that our implementation of the new variant is in general more expensive, in terms of CPU time per iteration, than the original IDR method. This is due to the fact that complex ω's may be used, enforcing complex arithmetic computation throughout. A more sophisticated real implementation that combines complex conjugate ω's may be considered.

In all examples, the right-hand side was $b = \mathbf{1}$, normalized to have unit norm.

EXAMPLE 7.1. We consider the benchmark parameterized example available in the IDR website [26], whose matrix is given as

$$A = A_0 - \gamma I + (Pe)B$$

where A_0 is the symmetric and positive definite discretization of the Poisson operator in a square, and B is the bidiagonal matrix with -1 and 1 on the lower and upper diagonals, respectively. We first consider the data for the example suggested in [26], namely $Pe = 0.1$, $\gamma = 1$ and matrix size $n = 400$. The behavior of IDR(4) is similar to that reported in the literature. The resulting convergence is shown in the left plot of Figure 7.1, together with that of full GMRES.

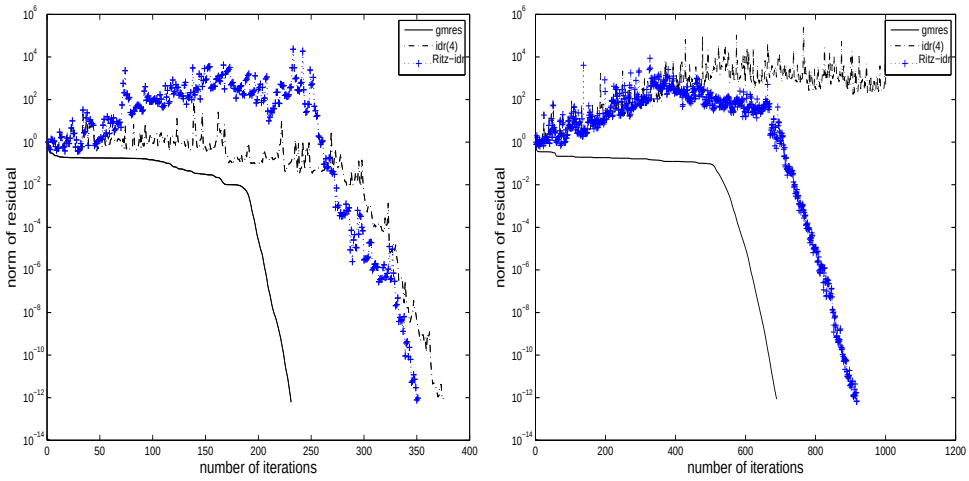

FIG. 7.1. *Example 7.1. Convergence history for various methods and two parameter settings.*

The right plot displays the convergence history for the same problem, with $Pe = 1$, $\gamma = 0.5$ and size $n = 1600$. All other parameters and variables were kept the same. We readily notice that the new version of IDR converges whereas the original method continues to oscillate for many more iterations. It eventually converges to a comparable residual norm after more than 1400 iterations. We also report that while

full GMRES required 43.89 seconds to converge, with large memory requirements, the new variant of IDR only took 0.36 seconds. It is also important to notice that restarted GMRES with $m = 20$ fully stagnates.

EXAMPLE 7.2. We consider the centered finite difference discretization of the operator $L(u) = -\Delta u + \beta(u_x + u_y + u_z)$ in the unit cube with homogeneous Dirichlet boundary conditions, giving rise to a matrix of size $n = 8000$. Figure 7.2 shows the convergence history of all methods for $\beta = 100$ (left) and $\beta = 500$ (right). These plots confirm that high nonsymmetry of the problem, in this case corresponding to larger convection, is problematic for the IDR method.

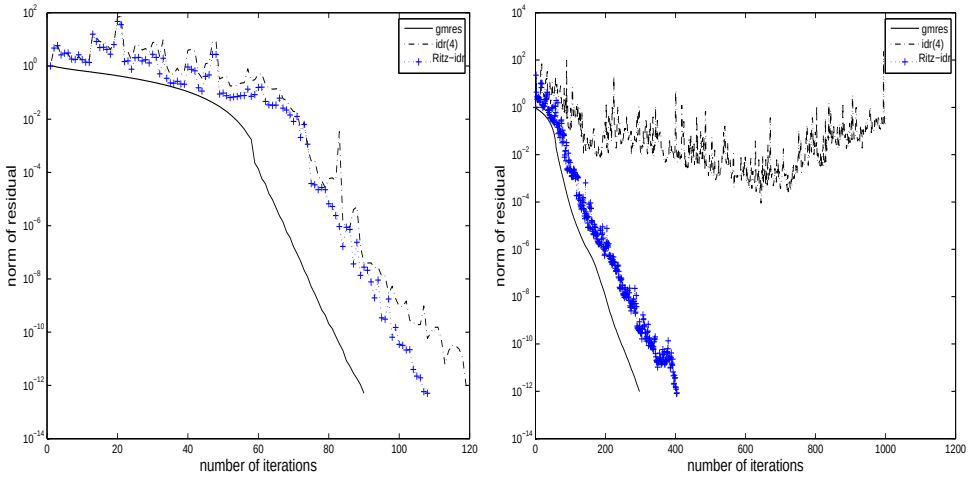

FIG. 7.2. *Example 7.2. Convergence history for various methods and two parameter settings:* $\beta = 100$ *(left),* $\beta = 500$ *(right).*

We also report that for the more difficult problem, $\beta = 500$, GMRES employed 27.26 seconds to converge, whereas the new IDR version took only 0.79 seconds. For completeness we also report the numerical results for other methods not displayed in the plots: Restarted GMRES with $m = 20$ required 2.85 seconds to achieve the same residual norm, after 20 restarts, for a total of matrix-vector multiplies similar to that of the new variant of IDR. The following table summarizes the performance of the original method IDR(s) for larger values of s, keeping in mind that IDR(s) requires the allocation of $3s+4$ long vectors:

s	iter	CPU time
8	812	1.23
12	591	1.18
20	417	1.37

The table shows that a larger block in the left space may cause higher CPU times, in spite of a lower number of iterations. We conclude this example by noticing that the convergence curve for $s = 20$ is similar to that of the new variant, although the approximation procedure is rather different.

EXAMPLE 7.3. We consider the centered finite difference discretization of the operator $L(u) = -\Delta u + 1000u_z$ in the unit cube with homogeneous Dirichlet boundary conditions, giving rise to a matrix of size $n = 8000$. Figure 7.3 shows the behavior of IDR(4) and of Ritz-IDR, compared to GMRES. The results are completely analogous

to those for the less asymmetric operator.

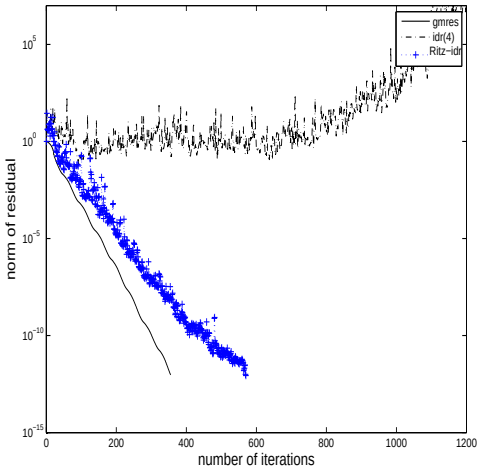

FIG. 7.3. *Example 7.3. Convergence history for various methods and operator* $L(u) = -\Delta u + 1000u_z$ *in the unit cube with homogeneous Dirichlet boundary conditions.*

EXAMPLE 7.4. We consider the Benchmark matrix SHERMAN5 from the Matrix Market [10]. This is a highly indefinite, 3312×3312 matrix. For both IDR methods $s = 4$ was used. The Ritz-IDR method does not work well on this example, as shown in the left plot of Figure 7.4. No improvements were obtained by allowing more Ritz values in a larger space. Similar negative results were obtained for other highly indefinite problems.

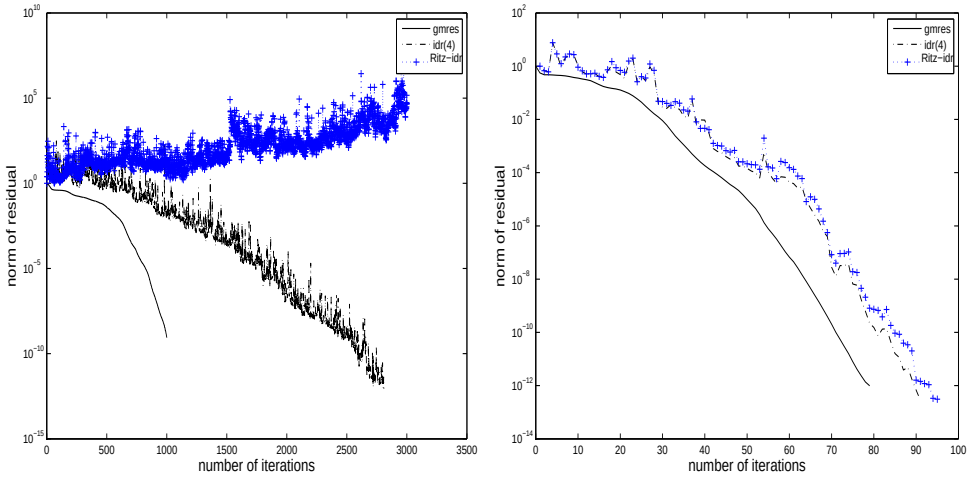

FIG. 7.4. *Example 7.4. Convergence history for various methods and matrix* SHERMAN5. *Left: original problem. Right: ILU*(10^{-2}) *preconditioning.*

The right plot shows the performance of all methods when preconditioning is applied. Here the preconditioner is obtained by an incomplete LU decomposition with threshold tolerance 10^{-2} on the shifted matrix $A + I$. All methods behave roughly

the same, in terms of number of iterations, although the original IDR is bound to be cheaper if CPU time is considered.

EXAMPLE 7.5. We conclude with another example from the Matrix Market, the 2395×2395 matrix ADD20 [10]. This matrix has only two nonreal eigenvalues, and the real spectral interval is given by $[5.9\,10^{-5}, 0.73]$, with many eigenvalues clustered in the lower part of the interval. We believe that this is the reason why we found more effective to take *all* $m_0 = 20$ Ritz values in this case, using Leja ordering. The convergence history of the methods for $s = 4$ is reported in Figure 7.5, to achieve a final residual norm less than 10^{-8}, as used in [21]. The performance of the two IDR-type methods is similar. The peaks in the Ritz-IDR residuals are related to the presence of the small Ritz values.

Due to the special spectrum, we also considered varying m_0, and the number of iterations for Ritz-IDR to converge is reported in the following table, where all m_0 Ritz values are used during the iterations:

m_0	iter
20	509
30	389
40	380

Apparently, 30 Ritz values are sufficient to significantly improve convergence, while a higher number only marginally improves the computational costs.

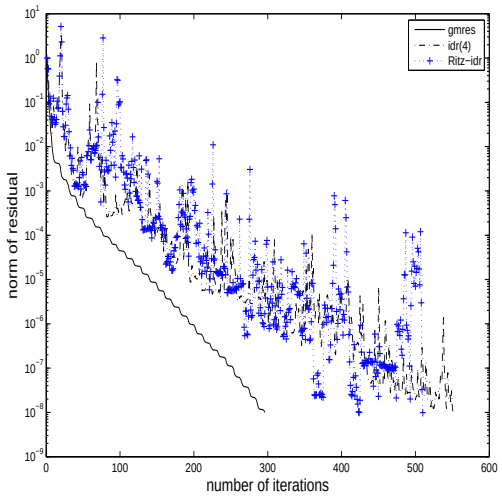

FIG. 7.5. *Example 7.5. Convergence history for various methods and matrix* ADD20.

8. Conclusions. We have shown that the recently explored family of IDR methods may be viewed as Petrov-Galerkin methods with respect to a (left) block rational Krylov subspace, whose block size corresponds to the method parameter s. The left space is not explicitly generated, therefore no matrix solves need to be performed. Nonetheless, the IDR methods can fully exploit the implicitly imposed orthogonality.

We have also suggested a different way of computing the poles of this left space, which may overcome the difficulties encountered when using a small block size (that is, a small value of s) for building the left block subspace. The new variant seems to work well on convection-diffusion problems with definite spectrum. Further investigation,

possibly in the selection of the poles, is required to improve its performance when dealing with highly indefinite problems; cf. [20, §4.2.2] for some considerations on similar issues.

Acknowledgements. This research commenced during a visit of the second author to the Università di Bologna. The warm hospitality received is greatly appreciated. The research of the second author is supported in part by the U.S. Department of Energy under grant DE-FG02-05ER25672. We thank Martin van Gijzen for helpful discussions on the IDR method. We also thank him, as well as Martin Gutknecht, Jans-Peter Zemke, and two referees for comments on an earlier version of the paper.

REFERENCES

[1] Zhaojun Bai, James Demmel, Jack Dongarra, Axel Ruhe, and Henk A. van der Vorst, editors. *Templates for the Solution of Algebraic Eigenvalue Problems: a Practical Guide*. SIAM, Philadelphia, 2000.

[2] Bernhard Beckermann and Lothar Reichel. Error estimation and evaluation of matrix functions via the Faber transform. *SIAM Journal on Numerical Analysis*, 2009. In press.

[3] Vladimir Druskin, Leonid Knizhnerman, and Mikhail Zaslavsky. On optimal convergence rate of the rational Krylov subspace reduction for electromagnetic problems in unbounded domains. *SIAM Journal on Numerical Analysis*, 47:953–971, 2009.

[4] Kyle A. Gallivan, Eric Grimme, and Paul Van Dooren. A rational Lanczos algorithm for model reduction. *Numerical Algorithms*, 12:33–63, 1996.

[5] Martin H. Gutknecht. Variants of BiCGStab for matrices with complex spectrum. *SIAM Journal on Scientific Computing*, 14:1020–1033, 1993.

[6] Martin H. Gutknecht. Block Krylov space methods for linear systems with multiple right-hand sides: an introduction. In Abul Hasan Siddiqi, Iain S. Duff, and Ole Christensen, editors, *Modern Mathematical Models, Methods and Algorithms for Real World Systems*, pages 420–447, New Delhi, 2007. Anamaya Publishers.

[7] Martin H. Gutknecht. IDR explained. *Electronic Transactions on Numerical Analysis*, 36:126–148, 2010.

[8] Nicholas J. Higham. The Matrix Computation Toolbox. Available at `http://www.ma.man.ac.uk/~higham/mctoolbox`.

[9] Michiel E. Hochstenbach, David A. Singer, and Paul F. Zachlin. On the field of values and pseudospectra of the inverse of a large matrix. CASA report 07-06, Department of Mathematics, TU Eindhoven, The Netherlands, February 2007. Submitted.

[10] Matrix Market. A repository of test data for numerical linear algebra. Available at `http://math.nist.gov/MatrixMarket/`.

[11] Lothar Reichel. The application of Leja points to Richardson iteration and polynomial preconditioning. *Linear Algebra and its Applications*, 154-156:389–414, 1991.

[12] Axel Ruhe. Rational Krylov sequence methods for eigenvalue computation. *Linear Algebra and its Applications*, 58:391–405, 1984.

[13] Axel Ruhe. The rational Krylov algorithm for nonsymmetric eigenvalue problems. III: complex shifts for real matrices. *BIT Numerical Mathematics*, 34:165–176, 1994.

[14] Yousef Saad. *Iterative Methods for Sparse Linear Systems*. The PWS Publishing Company, Boston, 1996. Second edition, SIAM, Philadelphia, 2003.

[15] Valeria Simoncini. A stabilized QMR version of block BiCG. *SIAM Journal on Matrix Analysis and Applications*, 18:419–434, 1997.

[16] Valeria Simoncini and Daniel B. Szyld. Recent computational developments in Krylov subspace methods for linear systems. *Numerical Linear Algebra with Applications*, 14:1–59, 2007.

[17] Gerard L. G. Sleijpen and Diederik R. Fokkema. BiCGStab(ℓ) for linear equations involving unsymmetric matrices with complex spectrum. *Electronic Transactions on Numerical Analysis*, 1:11–32, 1993.

[18] Gerard L. G. Sleijpen and Henk A. van der Vorst. Maintaining convergence properties of BiCGstab methods in finite precision arithmetic. *Numerical Algorithms*, 10:203–223, 1995.

[19] Gerard L. G. Sleijpen and Henk A. van der Vorst. An overview of approaches for the stable computation of hybrid BiCG methods. *Applied Numerical Mathematics*, 19:235–254, 1995.

[20] Gerard L. G. Sleijpen, Henk A. van der Vorst, and Diederik R. Fokkema. BiCGStab(ℓ) and other hybrid Bi-CG methods. *Numerical Algorithms*, 7:75–109, 1994.

[21] Gerard L.G. Sleijpen, Peter Sonneveld, and Martin B. van Gijzen. Bi-CGSTAB as an induced dimension reduction method. *Applied Numerical Mathematics*, 2009. In press.

[22] Gerard L.G. Sleijpen and Martin B. van Gijzen. Exploiting bi-CGSTAB(ℓ) strategies to induce dimension reduction. Technical Report 09-02, Department of Applied Mathematical Analysis, Delft University of Technology, 2009.

[23] Peter Sonneveld and Martin B. van Gijzen. IDR(s): a family of simple and fast algorithms for solving large nonsymmetric linear systems. *SIAM Journal on Scientific Computing*, 31:1035–1062, 2008.

[24] Masaaki Tanio and Masaaki Sugihara. GBi-CGSTAB(s, L): IDR(s) with higher-order stabilization polynomials. Technical Report METR 2009–16, Department of Mathematical Informatics, University of Tokyo, 2009.

[25] Henk A. van der Vorst. Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM Journal on Scientific Computing*, 13:631–644, 1992.

[26] Martin B. van Gijzen. IDR website, 2009. Available at `http://ta.twi.tudelft.nl/nw/users/gijzen/IDR.html`.

[27] Martin B. van Gijzen and Peter Sonneveld. An elegant IDR(s) variant that efficiently exploits bi-orthogonality properties. Technical Report 08-21, Department of Applied Mathematical Analysis, Delft University of Technology, 2008.

[28] Pieter Wesseling and Peter Sonneveld. Numerical experiments with a multiple grid and a preconditioned Lanczos type method. In Reimund Rautmann, editor, *Approximation methods for Navier-Stokes problems (Proceedings of the symposium held by the International Union of Theoretical and Applied Mechanics (IUTAM) at Universität Padeborn, Germany, 9-15 September 1979)*, volume 771 of *Lecture Notes in Mathematics*, pages 543–562. Springer, Berlin, New York, 1980.

[29] Man-Chung Yeung and Tony Chan. ML(K)BiCGSTAB: A BiCGSTAB variant based on multiple Lanczos starting vectors. *SIAM Journal on Scientific Computing*, 21:1263–1290, 1999.