History

IDR basics

Case $s=1$

Case $s>1$

Conclusions
History: References


History: References (cont’d)


Received May 16, 1997. Introduce ML(k)BiCGSTAB — a version of BiCGSTAB with “multiple left (shadow) residuals”. Astonishing numerical results.
History: References (cont’d)

P. Sonneveld and M. B. van Gijzen, IDR(s): a family of simple and fast algorithms for solving large nonsymmetric systems of linear equations, Report 07-07, Department of Applied Mathematical Analysis, Delft University of Technology.
Generalizing IDR ≈ IDR(1) to IDR(s). Detailed description, connection to BiCGSTAB; very good numerical results.

G. Sleijpen, P. Sonneveld, and M. B. van Gijzen, Bi-CGSTAB as an induced dimension reduction method, Report 08-07, Department of Applied Mathematical Analysis, Delft University of Technology.
Partly different view; partly different notation. Introduce variation of ML(k)BiCGSTAB; but no comparison.
IDR(s) basics: the setting

Given: linear system \( \mathbf{Ax} = \mathbf{b} \in \mathbb{C}^N \), initial approx. \( \mathbf{x}_0 \). Let:

\[
\mathbf{r}_0 \equiv \mathbf{b} - \mathbf{Ax}_0,
\]
\[
\mathcal{K}_m \equiv \mathcal{K}_m(\mathbf{A}, \mathbf{r}_0) \equiv \text{span} \{ \mathbf{r}_0, \mathbf{Ar}_0, \ldots, \mathbf{A}^{m-1} \mathbf{r}_0 \},
\]

\[\nu\] such that \( \mathcal{G}_0 \equiv \mathcal{K}_\nu \) invariant,

\( \mathcal{S} \subset \mathbb{C}^N \) linear subspace of dimension \( N - s \),

for \( j = 1, 2, \ldots \): choose \( \omega_j \neq 0 \) and let

\[
\mathcal{G}_j \equiv (\mathbf{I} - \omega_j \mathbf{A})(\mathcal{G}_{j-1} \cap \mathcal{S}),
\]

for \( n = n_j, \ldots, n_{j+1} - 1 \):

choose \( \mathbf{x}_n \) such that

\[
\mathbf{r}_n \in \mathcal{G}_j \cap (\mathbf{r}_0 + \mathbf{A}\mathcal{K}_n).
\]

Note: Typically \( n_{j+1} := n_j + s + 1 \).
IDR(s) basics: the spaces $\mathcal{G}_j$ (case $s = 1$)

$G_0 = \mathbb{R}^3$

$G_0 \cap S = S$

$G_2 \cap S = \{0\} = \{v_5\}$

$G_j :\equiv (I - \omega j A)(G_{j-1} \cap S)$
IDR(s) basics: the first three steps (case $s = 1$)

$G_0 = R^3$

$I - \omega_1 A$

$G_1 \cap S = S$

$r_0$

$r_1$

$r_2 = (I - \omega_1 A)v_1$

$r_3$

$v_1$

$v_2$

$v_3$
**IDR(s) basics: all the residuals (case $$s = 1$$)**

$$G_0 = R^3$$

$$G_0 \cap S = S$$

$$G_2 \cap S = \{0\} = \{v_5\}$$

$$r_2 = (I - \omega_1 A)v_1$$

$$r_1$$

$$r_0$$

$$r_4$$

$$r_3$$

$$r_5$$

$$v_1$$

$$v_2$$

$$v_3$$

$$v_4$$

$$G_1$$

$$G_2$$

$$G_1 \cap S$$

$$I - \omega_1 A$$

$$I - \omega_2 A$$

$$G_0$$

$$G_1$$

$$G_2$$

$$S$$

$$r_1$$

$$r_0$$

$$r_4$$

$$r_3$$

$$r_5$$

$$v_1$$

$$v_2$$

$$v_3$$

$$v_4$$

$$v_5$$

$$G_0 \cap S = S$$

$$G_2 \cap S = \{0\} = \{v_5\}$$

$$r_2 = (I - \omega_1 A)v_1$$

$$r_1$$

$$r_0$$

$$r_4$$

$$r_3$$

$$r_5$$

$$v_1$$

$$v_2$$

$$v_3$$

$$v_4$$

$$v_5$$

$$G_0$$

$$G_1$$

$$G_2$$

$$S$$
IDR(s) basics: IDR theorem

Recall: \( G_j \equiv (I - \omega_j A)(G_{j-1} \cap S) \), \( r_n \in G_j \cap (r_0 + A\mathcal{K}_n) \).

Genericness assumption: \( S \cap G_0 \) contains no eigenvector of \( A \).

**Theorem (IDR Theorem (Wes/Son80, Son/vGi07))**

\( G_j \subsetneq G_{j-1} \) unless \( G_{j-1} = \{0\} \).
IDR(s) basics: IDR theorem

Recall: \( G_j \equiv (I - \omega_j A)(G_{j-1} \cap S) \), \( r_n \in G_j \cap (r_0 + A \mathcal{K}_n) \).

Genericness assumption: \( S \cap G_0 \) contains no eigenvector of \( A \).

**Theorem (IDR Theorem (Wes/Son80, Son/vGi07))**

\[ G_j \subsetneq G_{j-1} \text{ unless } G_{j-1} = \{0\} \.

Consequently: \( G_j = \{0\} \) for some \( j \leq N \).

Can expect: \( r_n = 0 \) once \( j = N \), that is, \( n = (s + 1)N \).

But typically: \( r_n = 0 \) once \( n = N \), because residuals turn out to be linearly independent.

Hence: IDR Thm. strongly underestimates convergence rate.
**IDR(s) basics: what’s different?**

Most currently used KSS (= Krylov subspace solvers) are based on a different kind of “induced dimension reduction”:

$$r_n \in \mathcal{L}_n^\perp \cap (r_0 + A\mathcal{K}_n(A, r_0)),$$

where, e.g.,

- $$\mathcal{L}_n = \mathcal{K}_n(A, r_0)$$ (CG),
- $$\mathcal{L}_n = A\mathcal{K}_n(A, r_0)$$ (CR, GCR, GMRES),
- $$\mathcal{L}_n = \mathcal{K}_n(A^*, \tilde{r}_0)$$ (BiCG).

What differs now is the special recursion for $$\{G_j\}.$$
IDR(s) basics: what’s different?

Most currently used KSS (= Krylov subspace solvers) are based on a different kind of “induced dimension reduction”:

$$ r_n \in L_n^\perp \cap (r_0 + A\mathcal{K}_n(A, r_0)) , $$

where, e.g.,

$$ L_n = \mathcal{K}_n(A, r_0) \quad (CG) , $$

$$ L_n = A\mathcal{K}_n(A, r_0) \quad (CR, GCR, GMRES) , $$

$$ L_n = \mathcal{K}_n(A^*, \tilde{r}_0) \quad (BiCG) . $$

What differs now is the special recursion for \( \{G_j\} \).

\( G_j \) is not the orthogonal complement of a Krylov subspace.

However, we will see that \( G_j \) is the image of an orthogonal complement of a Krylov subspace.
IDR(s) basics: recursions for \( \{r_n\} \)

Recall: \[ G_j \equiv (I - \omega_jA)(G_{j-1} \cap S), \quad r_n \in G_j \cap (r_0 + AK_n). \]

Assume: \( r_n \notin r_0 + AK_{n-1} \) (\( \forall n \)).

\[ r_{n+1} := (I - \omega_jA)v_n, \quad v_n \in G_{j-1} \cap S \cap (r_0 + AK_n), \]

\[ v_n := r_n - \sum_{k=1}^{K(n)} \gamma_k^{(n)} \Delta r_{n-k} = r_n - \Delta R_n c_n, \quad (1) \]

where \( K(n) \leq n, \)

\[ \Delta r_n \equiv r_{n+1} - r_n, \]

\[ \Delta R_n \equiv \begin{bmatrix} \Delta r_{n-1} & \cdots & \Delta r_{n-K(n)} \end{bmatrix}, \]

\[ c_n \equiv \begin{bmatrix} \gamma_1^{(n)} & \cdots & \gamma_{K(n)}^{(n)} \end{bmatrix}. \]

“Consistency condition” for residuals is implicit in (1).
Recall: \[ v_n = r_n - \sum_{k=1}^{K(n)} \gamma_k^{(n)} \Delta r_{n-k} = r_n - \Delta R_n c_n \in G_j \cap S. \] (1)

Since \( \text{dim} \, S = N - s \), there is \( P \in \mathbb{C}^{N \times s} \) s.t. \( S^\perp = \mathcal{R}(P) \):

\[ v_n \in S \iff v_n \perp S^\perp = \mathcal{R}(P) \iff P^*v_n = 0. \]

To achieve this, the term \( \Delta R_n c_n \) in (1) must be the oblique projection of \( r_n \) into \( \mathcal{R}(\Delta R_n) \) along \( S \).
Recall: \[ v_n = r_n - \sum_{k=1}^{K(n)} \gamma_{k}^{(n)} \Delta r_{n-k} = r_n - \Delta R_n c_n \in G_j \cap S. \quad (1) \]

Since \( \dim S = N - s \), there is \( P \in \mathbb{C}^{N \times s} \) s.t. \( S^\perp = R(P) \):

\[ v_n \in S \iff v_n \perp S^\perp = R(P) \iff P^* v_n = 0. \]

To achieve this, the term \( \Delta R_n c_n \) in (1) must be the oblique projection of \( r_n \) into \( R(\Delta R_n) \) along \( S \).

In order that this projection is uniquely defined, we need \( P^* \Delta R_n \) to be nonsingular. Then

\[ v_n = r_n - \Delta R_n \left( P^* \Delta R_n \right)^{-1} P^* r_n = r_n - \Delta R_n c_n. \]

We need \( K(n) = s \ (\forall n) \) to make \( P^* \Delta R_n \) a square matrix.
Recall: \[ \mathbf{r}_{n+1} = (\mathbf{I} - \omega_j \mathbf{A}) \mathbf{v}_n \] (2)

Here, \( \omega_j \) is fixed for \( n + 1 = n_j, \ldots, n_{j+1} \).

So, only for \( n + 1 = n_j \), we may choose \( \omega_j \) s.t. \( \|\mathbf{r}_{n+1}\| \) is minimal among all \( \mathbf{r} \) of the form \( \mathbf{r} = (\mathbf{I} - \omega_j \mathbf{A}) \mathbf{v}_n \), i.e., \( \mathbf{r} \perp \mathbf{A} \mathbf{v}_n \):

\[
\omega_j \equiv \frac{\langle \mathbf{A} \mathbf{v}_n, \mathbf{v}_n \rangle}{\|\mathbf{A} \mathbf{v}_n\|^2}.
\]
Recall: \[ r_{n+1} = (I - \omega_j A) v_n \] (2)

Here, \( \omega_j \) is fixed for \( n + 1 = n_j, \ldots, n_j+1 \).

So, only for \( n + 1 = n_j \), we may choose \( \omega_j \) s.t. \( \|r_{n+1}\| \) is minimal among all \( r \) of the form \( r = (I - \omega_j A) v_n \), i.e., \( r \perp Av_n : \)

\[
\omega_j \equiv \frac{\langle Av_n, v_n \rangle}{\|Av_n\|^2}.
\]

Note: Inserting (1) in (2) yields

\[ r_{n+1} := r_n - \Delta R_n c_n - \omega_j A(r_n - \Delta R_n c_n) \] (3)

which means that IDR(s) with \( K(n) = s \) is a \((k, \ell)\)-step method (G. ’89_NM) with \( k = \ell = s + 1 \).

Recall: CG, BiCG are \((2, 1)\)-step methods, ORTHOMIN(k) is a \((k, 1)\)-step method.
IDR(s) basics: recursions for \( \{x_n\} \)

**Note 1:** \( v_n \in r_0 + A\mathcal{K}_n \implies \exists x'_n \text{ s.t. } v_n = b - Ax'_n, \)
i.e., \( v_n \) is the residual of an “intermediate” iterate \( x'_n \in x_0 + \mathcal{K}_n. \)

**Note 2:** \( \Delta r_n = -A \Delta x_n, \quad \Delta R_n = -A \Delta R_n, \)

Hence:

\[
\begin{align*}
v_n &:= r_n - \Delta R_n c_n \implies x'_n := x_n - \Delta X_n c_n, \quad (4) \\
r_{n+1} &:= (I - \omega_j A) v_n \implies x_{n+1} := \omega_j v_n + x'_n. \quad (5)
\end{align*}
\]
IDR(s) basics: recursions for \( \{x_n\} \)

**Note 1:** \( v_n \in r_0 + A\mathcal{K}_n \implies \exists x'_n \text{ s.t. } v_n = b - Ax'_n \), i.e., \( v_n \) is the residual of an “intermediate” iterate \( x'_n \in x_0 + \mathcal{K}_n \).

**Note 2:** \( \Delta r_n = -A \Delta x_n, \quad \Delta R_n = -A \Delta R_n \),

Hence:

\[
\begin{align*}
  v_n &:= r_n - \Delta R_n c_n \implies x'_n := x_n - \Delta X_n c_n, \quad (4) \\
  r_{n+1} &:= (I - \omega jA) v_n \implies x_{n+1} := \omega j v_n + x'_n. \quad (5)
\end{align*}
\]

There are several ways to rearrange these four recursions and to combine them with the iterate-residual relationships; see [Sle/Son/vGi08]. Also [Son/vGi07] use different, but equivalent recursions that can be viewed as implementation of (3) and its analog for \( x_{n+1} \). (Even in their “prototype algorithm”.)
IDR(s) basics: characterization by orthogonality

**Theorem (Son/vGi07, Sle/Son/vGi08)**

Let \( \Omega_0(t) \equiv 1 \), \( \Omega_j(t) \equiv (1 - \omega_1 t) \cdots (1 - \omega_j t) \in \mathcal{P}_j^\circ \), where \( \mathcal{P}_j^\circ \equiv \{ \text{polyns. of degree } \leq j \text{ that are 1 at 0} \} \). Then

\[
\mathcal{G}_j = \left\{ \Omega_j(A)w \mid w \perp \mathcal{K}_j(A^*, P) \right\} = \Omega_j(A) \mathcal{K}_j(A^*, P)_{\perp} \equiv \mathcal{L}_j = \mathcal{L}_j^\perp.
\]

**Note:** \( \mathcal{L}_j \) is the left-hand side (LHS) block Krylov space that appears in the block Lanczos process with LHS block size \( s \).

**Note:** We may have Lanczos breakdowns and a collapsing block Krylov space (which requires deflation).
Since $r_n \in G_j \cap (r_0 + AK_n)$, and since the residual polynomials must have full degree, we have for $n = n_j, \ldots, n_{j+1} - 1$:

$$r_n = \Omega_j(A) w_n, \quad w_n \in (r_0 + AK_{n-j}) \cap L_j^+, \quad w_n \notin K_{n-j}. \quad (6)$$
Since \( r_n \in G_j \cap (r_0 + A \mathcal{K}_n) \), and since the residual polynomials must have full degree, we have for \( n = n_j, \ldots, n_{j+1} - 1 \):

\[
    r_n = \Omega_j(A) w_n, \quad w_n \in (r_0 + A \mathcal{K}_{n-j}) \cap \mathcal{L}_j^\perp, \quad w_n \notin \mathcal{K}_{n-j}.
\]

(6)

Generically, \( n_{j+1} - n_j = s + 1 \), so, for fixed \( j \), we hope for \( s + 1 \) linearly independent vectors \( w_n \) with \( n_j \leq n < n_{j+1} \).

Generically, for \( n = n_j = j(s + 1) \), where \( w_n \in (r_0 + A \mathcal{K}_{js}) \) and \( w_n \perp \mathcal{L}_j \) with \( \dim \mathcal{L}_j = js \), there is a unique \( w_n \) satisf. (6).
Since $r_n \in G_j \cap (r_0 + AK_n)$, and since the residual polynomials must have full degree, we have for $n = n_j, \ldots, n_{j+1} - 1$:

$$r_n = \Omega_j(A) w_n, \quad w_n \in (r_0 + AK_{n-j}) \cap L_j^\perp, \quad w_n \not\in K_{n-j}. \tag{6}$$

Generically, $n_{j+1} - n_j = s + 1$, so, for fixed $j$, we hope for $s + 1$ linearly independent vectors $w_n$ with $n_j \leq n < n_{j+1}$.

Generically, for $n = n_j = j(s + 1)$, where $w_n \in (r_0 + AK_{js})$ and $w_n \perp L_j$ with $\dim L_j = js$, there is a unique $w_n$ satisf. (6).

Cost of computing $r_{n_j} \in r_0 + AK_{n_j}$: $n_j + 1$ MVs with $A$.

But the $s$ other vectors $w_n$ are not uniquely determined by (6).
Since $r_n \in G_j \cap (r_0 + A\mathcal{K}_n)$, and since the residual polynomials must have full degree, we have for $n = n_j, \ldots, n_{j+1} - 1$:

$$r_n = \Omega_j(A) w_n, \quad w_n \in (r_0 + A\mathcal{K}_{n-j}) \cap L_j^\perp, \quad w_n \notin \mathcal{K}_{n-j}. \quad (6)$$

Generically, $n_{j+1} - n_j = s + 1$, so, for fixed $j$, we hope for $s + 1$ linearly independent vectors $w_n$ with $n_j \leq n < n_{j+1}$.

Generically, for $n = n_j = j(s + 1)$, where $w_n \in (r_0 + A\mathcal{K}_{js})$ and $w_n \perp L_j$ with dim $L_j = js$, there is a unique $w_n$ satisf. (6).

Cost of computing $r_{n_j} \in r_0 + A\mathcal{K}_{n_j}$: $n_j + 1$ MVs with $A$.

But the $s$ other vectors $w_n$ are not uniquely determined by (6).

Note: The larger $s$, the larger is $L_j$ and the smaller is $G_j$.

Note: For $n = n_j - 1$ and $n = n_j$, the polynomials associated with $w_n$ have the same degree: $w_n \in r_0 + A\mathcal{K}_{js}$. 
\( s = 1: \text{IDR}(1) \sim \text{BICGSTAB} \)

In this case, the recursions simplify since \( K(n) = s = 1 \) and every other set of vectors \((w_n, r_n, v_{n-1}, x_n, \ldots)\) is uniquely determined — up to the choice of the parameters \(\omega_j\).

If the latter are chosen as in \(\text{BICGSTAB}\),

\[
\begin{align*}
      r_{2j} &= r_j^{\text{STAB}}, & x_{2j} &= x_j^{\text{STAB}}, & w_{2j} &= r_j^{\text{BICG}},
\end{align*}
\]

where \(r_j^{\text{BICG}}\) is the \(j\)th residual of \(\text{BICG}\): \(r_j^{\text{BICG}} = \rho_j(A)r_0\).

Recursions (4) and (5), with \(\gamma_n \equiv \gamma_1^{(n)} = \langle p, r_n \rangle / \langle p, \Delta r_{n-1} \rangle\):

\[
\begin{align*}
      v_n &:= (1 - \gamma_n)r_n + \gamma_n r_{n-1}, & x'_n &:= (1 - \gamma_n)x_n + \gamma_n x_{n-1}, \\
      r_{n+1} &:= (I - \omega_j A)v_n, & x_{n+1} &:= x'_n + \omega_j v_n.
\end{align*}
\]
\( s = 1: \) polynomial recursions

\[
\begin{align*}
  r_n &= \Omega_j(A)w_n = \\
        &\begin{cases} 
          \Omega_j(A)\rho_j(A)r_0 & \text{if } n = 2j, \\
          \Omega_j(A)\tilde{\rho}_{j+1}(A)r_0 & \text{if } n = 2j + 1,
        \end{cases} \\
  v_n &= \Omega_{j-1}(A)w_{n+1} = \\
        &\begin{cases} 
          \Omega_{j-1}(A)\rho_j(A)r_0 & \text{if } n = 2j - 1, \\
          \Omega_{j-1}(A)\tilde{\rho}_{j+1}(A)r_0 & \text{if } n = 2j,
        \end{cases}
\end{align*}
\]
**s = 1: polynomial recursions**

\[ r_n = \Omega_j(A)w_n = \begin{cases} 
\Omega_j(A)\rho_j(A)r_0 & \text{if } n = 2j, \\
\Omega_j(A)\hat{\rho}_{j+1}(A)r_0 & \text{if } n = 2j + 1,
\end{cases} \]

\[ v_n = \Omega_{j-1}(A)w_{n+1} = \begin{cases} 
\Omega_{j-1}(A)\rho_j(A)r_0 & \text{if } n = 2j - 1, \\
\Omega_{j-1}(A)\hat{\rho}_{j+1}(A)r_0 & \text{if } n = 2j,
\end{cases} \]

Inserting these formulas into \( v_n = (1 - \gamma_n)r_n + \gamma_n r_{n-1} \) we get, after a short calculation, for \( n = 2j \) and \( n = 2j + 1 \), respectively,

\[ \hat{\rho}_{j+1}(t) := (1 - \gamma_{2j}) (1 - \omega_j t) \rho_j(t) + \gamma_{2j} \hat{\rho}_j(t), \]
\[ \rho_{j+1}(t) := (1 - \gamma_{2j+1}) \hat{\rho}_{j+1}(t) + \gamma_{2j+1} \rho_j(t). \]  

(8)

Recall: \( w_{2j} = \rho_j(A)r_0 \perp L_j \) and \( w_{2j+1} = \hat{\rho}_j(A)r_0 \perp L_j. \)
\( s = 1: \) FOPs, BICG, BICGSTAB

The Lanczos (residual) polynomials \( \rho_j \) and the BICG search direction polynomials \( \sigma_j \) are formal orthogonal polynomials (FOPs) in the sense that, for \( i \neq j \),

\[
\rho_i \perp \rho_j \iff \langle \rho_i(A^*)\tilde{r}_0, \rho_j(A)r_0 \rangle = 0,
\]

\[
\sigma_i \perp \tau \sigma_j \iff \langle \sigma_i(A^*)\tilde{r}_0, A^* \sigma_j(A)r_0 \rangle = 0,
\]

where \( \tilde{v}_i^{BICG} \) and \( \tilde{v}_i^{BICG} \) are the search directions and the “shadow” search directions, respectively. In particular:

\[
\begin{align*}
\rho_j \perp P_{j-1}, & \quad \sigma_j \perp \tau P_{j-1}, \\
\iff & \quad r_j^{BICG} \perp \mathcal{L}_j, \quad v_j^{BICG} \perp A \mathcal{L}_j.
\end{align*}
\]
BICG uses in its standard version coupled two-term recursions:

\[
\begin{align*}
\mathbf{r}_{j+1} & := \mathbf{r}_j - \alpha_j \mathbf{A} \mathbf{v}_j, \\
\mathbf{v}_{j+1} & := \mathbf{r}_{j+1} + \beta_j \mathbf{v}_j.
\end{align*}
\]

The corresponding recursions for \(\rho_j\) and \(\sigma_j\) are

\[
\begin{align*}
\rho_{j+1}(t) & := \rho_j(t) - \alpha_j t \sigma_j(t), \\
\sigma_{j+1}(t) & := \rho_{j+1}(t) + \beta_j \sigma_j(t).
\end{align*}
\]

In contrast, in IDR(1), by (8),

\[
\begin{align*}
\hat{\rho}_{j+1}(t) & := (1 - \gamma_{2j}) (1 - \omega_j t) \rho_j(t) + \gamma_{2j} \hat{\rho}_j(t), \\
\rho_{j+1}(t) & := (1 - \gamma_{2j+1}) \hat{\rho}_{j+1}(t) + \gamma_{2j+1} \rho_j(t).
\end{align*}
\]
Comparing the recursions for \( (\rho_j, \sigma_j) \) with those for \( (\rho_j, \hat{\rho}_j) \) we easily see:

\[
(1 - \gamma_{2j+1}) (\hat{\rho}_{j+1}(t) - \rho_j(t)) = -\alpha_j t \sigma_j(t),
\]

or,

\[
\hat{\rho}_{j+1}(t) = \rho_j(t) - \frac{\alpha_j}{1 - \gamma_{2j+1}} t \sigma_j(t),
\]

or,

\[
r_{2j+1} = r_{2j} - \frac{\alpha_j}{(1 - \gamma_{2j+1})} A \Omega_j(A)v_j^{BICG},
\]

\[\equiv: s_j^{STAB} \]

This formula expresses the odd indexed IDR(1) residuals in terms of quantities from BICGSTAB and the coefficient \( \gamma_{2j+1} \).
\textbf{$s = 1$: Comments and conclusions}

- Even indexed IDR(1) residuals are B\textsc{icgstabil} residuals.
**s = 1: Comments and conclusions**

- Even indexed IDR(1) residuals are BICGSTAB residuals.
- \( \rho_j \) and \( \sigma_j \) are fully determined by their degree, their orthogonality condition, the consistency condition \( \rho_j(0) = 1 \), and equal leading coefficients of \( \rho_j \) and \( \sigma_j \).
s = 1: Comments and conclusions

- Even indexed IDR(1) residuals are BICGSTAB residuals.
- $\rho_j$ and $\sigma_j$ are fully determined by their degree, their orthogonality condition, the consistency condition $\rho_j(0) = 1$, and equal leading coefficients of $\rho_j$ and $\sigma_j$.
- In IDR, $\rho_j$ is the same as in BICG. But $\hat{\rho}_j$ is not determined uniquely by degree, orthogonality, and $\hat{\rho}_j(0) = 1$. The last degree of freedom is fixed by the form of the recursions.
• IDR(1) can be viewed as a minor variation of BICGSTAB.
• It is not clear, why one or the other should be more stable.
- IDR(1) can be viewed as a minor variation of BICGSTAB.
- It is not clear, why one or the other should be more stable.
- In fact, the existence of all even indexed IDR(1) residuals requires (like the BIORS version of BICG) that no Lanczos breakdowns and no pivot breakdowns occur.
- The smoothing step breakdown ($\omega_j = 0$) is also the same and can be treated easily by choosing a non-optimal $\omega_j$. 
• Variations of the “prototype algorithm” for IDR(1) method allow us to avoid certain breakdowns; but not all details are clear.

• Likewise there are variations of BICGSTAB that avoid breakdowns by using look-ahead [G./Ressel ’00SIMAX]; they may be more complicated, however.

• So far there are no IDR analogs of BICGSTAB2 [G. ’93SISC] or BICGSTAB(ℓ) [Sleijpen/Fokkema ’93ETNA]. In particular, IDR(s) with s > 1 has nothing to do with BICGSTAB(ℓ) with ℓ > 1.
s = 1: How did the original IDR differ from IDR(1)?

In contrast to IDR(1) of [Son/vGi07], where we had

\[
\begin{align*}
  v_n &:= r_n - \gamma_n(r_n - r_{n-1}), & x'_n &:= x_n - \gamma_n(x_n - x_{n-1}), \\
  r_{n+1} &:= (I - \omega_j A) v_n, & x_{n+1} &:= x'_n + \omega_j v_n,
\end{align*}
\]

the original IDR of [Wes/Son80] used for n odd the recursions

\[
\begin{align*}
  v_n &:= r_n - \gamma'_n(r_{n-1} - r_{n-2}), & x'_n &:= x_n - \gamma'_n(x_{n-1} - x_{n-2}), \\
  r_{n+1} &:= (I - \omega_j A) v_n, & x_{n+1} &:= x'_n + \omega_j v_n.
\end{align*}
\]

with \(\gamma'_n \equiv \gamma_1^{(n)} = \langle p, r_n \rangle / \langle p, \Delta r_{n-2} \rangle\).
$s = 1$: How did the original IDR differ from IDR(1)?

In contrast to IDR(1) of [Son/vGi07], where we had

$$v_n := r_n - \gamma_n(r_n - r_{n-1}), \quad x'_n := x_n - \gamma_n(x_n - x_{n-1}),$$
$$r_{n+1} := (I - \omega jA)v_n, \quad x_{n+1} := x'_n + \omega jv_n,$$

the original IDR of [Wes/Son80] used for $n$ odd the recursions

$$v_n := r_n - \gamma'_n(r_{n-1} - r_{n-2}), \quad x'_n := x_n - \gamma'_n(x_{n-1} - x_{n-2}),$$
$$r_{n+1} := (I - \omega jA)v_n, \quad x_{n+1} := x'_n + \omega jv_n.$$

with $\gamma'_n := \gamma_1^{(n)} = \langle p, r_n \rangle / \langle p, \Delta r_{n-2} \rangle$.

Moreover, in discrepancy of what we have stated here, the new IDR(s) computes the residual differences as $\Delta R = -A\Delta X$. This couples the recursions for $x_n$ and $r_n$ more tightly.
s > 1: IDR(s) \sim ML(s)BICGSTAB

- Relation IDR(1) \sim BICGSTAB \leftrightarrow BICG is matched by relation IDR(s) \sim ML(s)BICGSTAB \leftrightarrow ML(s)BICG.

- ML(s)BICG and ML(s)BICGSTAB are due to Man-Chung Yeung and Tony Chan '97/'99 SISC. Connection to nonsym. block Lanczos [Aliaga/Boley/Freund/Hernández '96/'99 MC].

- Oddly, essentially the same methods are also introduced in [Sle/Son/vGi08]. Yeung/Chan are cited in introduction only.
The fundamental discovery that, in the framework of Lanczos-type product methods, multiple left projections can both speed up the convergence and reduce the MV count (per search space dimension) is due to Yeung and Chan. But Sonneveld and van Gijzen rediscovered it 10 years later independently.

The Yeung/Chan paper is well written, but the formulas look horrible. Amazingly, their Matlab program http://www.uwyo.edu/mathmyeung/p12/mlbicgstab.txt is only 187 lines (incl. 30 lines of comments).
Conclusions

- IDR(1) is as good as BICGSTAB or better.
- IDR(s) is as good as ML(s)BICGSTAB or better.
- Typically, IDR(s) outperforms other methods for a nonsymmetric problem.
- What is missing is the IDR-like generalization of BiCGStab2 to cover the case where $\mathbf{A}$ is real, but has (strongly) non-real eigenvalues.