Tuning IDR to fit your applications

Jens-Peter M. Zemke
zemke@tu-harburg.de

joint work with Olaf Rendel & Anisa Rizvanolli

Institut für Numerische Simulation
Technische Universität Hamburg-Harburg

October 23th, 2011, 14:05 - 14:50

Outline

Krylov subspace methods

Hessenberg decompositions

Polynomial representations

IDR

IDR and IDREig

IDRStab and QMRIDR

Tuning IDR

General comments

Shadow vectors

Stabilizing polynomials

Choosing s

Outline

Introduction

Krylov subspace methods: approximations

$$\left.\begin{array}{l}\mathbf{x}_k,\underline{\mathbf{x}}_k,\\ \mathbf{y}_k,\underline{\mathbf{y}}_k\end{array}\right\} \in \mathcal{K}_k(\mathbf{A},\mathbf{q}) := \text{span}\,\{\mathbf{q},\mathbf{A}\mathbf{q},\ldots,\mathbf{A}^{k-1}\mathbf{q}\} = \{p(\mathbf{A})\mathbf{q} \mid p \in \mathbb{P}_{k-1}\},$$

where

$$\mathbb{P}_{k-1} := \Big\{\sum_{j=0}^{k-1} \alpha_j z^j \mid \alpha_j \in \mathbb{C},\ 0 \leqslant j < k\Big\},$$

Introduction

Krylov subspace methods: approximations

$$\left.\begin{matrix} \mathbf{x}_k, \underline{\mathbf{x}}_k, \\ \mathbf{y}_k, \underline{\mathbf{y}}_k \end{matrix}\right\} \in \mathcal{K}_k(\mathbf{A}, \mathbf{q}) := \text{span}\,\{\mathbf{q}, \mathbf{A}\mathbf{q}, \dots, \mathbf{A}^{k-1}\mathbf{q}\} = \{p(\mathbf{A})\mathbf{q} \mid p \in \mathbb{P}_{k-1}\},$$

where

$$\mathbb{P}_{k-1} := \left\{\sum_{j=0}^{k-1} \alpha_j z^j \mid \alpha_j \in \mathbb{C},\ 0 \leqslant j < k\right\},$$

to solutions of linear systems

$$\mathbf{A}\mathbf{x} = \mathbf{r}_0 \ (= \mathbf{b} - \mathbf{A}\mathbf{x}_0), \qquad \mathbf{A} \in \mathbb{C}^{n \times n}, \quad \mathbf{b}, \mathbf{x}_0 \in \mathbb{C}^n,$$

Introduction

Krylov subspace methods: approximations

$$\left.\begin{matrix} \mathbf{x}_k, \underline{\mathbf{x}}_k, \\ \mathbf{y}_k, \underline{\mathbf{y}}_k \end{matrix}\right\} \in \mathcal{K}_k(\mathbf{A}, \mathbf{q}) := \operatorname{span}\{\mathbf{q}, \mathbf{A}\mathbf{q}, \dots, \mathbf{A}^{k-1}\mathbf{q}\} = \{p(\mathbf{A})\mathbf{q} \mid p \in \mathbb{P}_{k-1}\},$$

where

$$\mathbb{P}_{k-1} := \Big\{\sum_{j=0}^{k-1} \alpha_j z^j \mid \alpha_j \in \mathbb{C},\ 0 \leqslant j < k\Big\},$$

to solutions of linear systems

$$\mathbf{A}\mathbf{x} = \mathbf{r}_0\ (= \mathbf{b} - \mathbf{A}\mathbf{x}_0), \qquad \mathbf{A} \in \mathbb{C}^{n \times n}, \quad \mathbf{b}, \mathbf{x}_0 \in \mathbb{C}^n,$$

and (partial) eigenproblems

$$\mathbf{A}\mathbf{v} = \mathbf{v}\lambda, \qquad \mathbf{A} \in \mathbb{C}^{n \times n}.$$

Hessenberg decompositions

Construction of basis vectors resembled in structure of arising Hessenberg decomposition

$$\mathbf{A}\mathbf{Q}_k = \mathbf{Q}_{k+1}\underline{\mathbf{H}}_k,$$

where

- $\mathbf{Q}_{k+1} = (\mathbf{Q}_k, \mathbf{q}_{k+1}) \in \mathbb{C}^{n\times(k+1)}$ collects basis vectors,
- $\underline{\mathbf{H}}_k \in \mathbb{C}^{(k+1)\times k}$ is unreduced extended Hessenberg.

Hessenberg decompositions

Construction of basis vectors resembled in structure of arising Hessenberg decomposition

$$\mathbf{A}\mathbf{Q}_k = \mathbf{Q}_{k+1}\underline{\mathbf{H}}_k,$$

where

- $\mathbf{Q}_{k+1} = (\mathbf{Q}_k, \mathbf{q}_{k+1}) \in \mathbb{C}^{n\times(k+1)}$ collects basis vectors,
- $\underline{\mathbf{H}}_k \in \mathbb{C}^{(k+1)\times k}$ is unreduced extended Hessenberg.

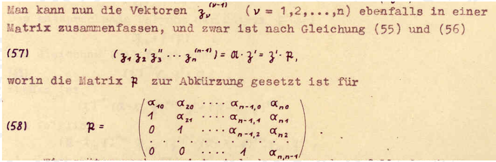
Aspects of perturbed Krylov subspace methods: captured with perturbed Hessenberg decompositions

$$\mathbf{A}\mathbf{Q}_k + \mathbf{F}_k = \mathbf{Q}_{k+1}\underline{\mathbf{H}}_k,$$

$\mathbf{F}_k \in \mathbb{C}^{n\times k}$ accounts for perturbations (finite precision & inexact methods).

Karl Hessenberg & “his” matrix + decomposition

"‘Behandlung linearer Eigenwertaufgaben mit Hilfe der Hamilton-Cayleyschen Gleichung"’, Karl Hessenberg, 1. Bericht der Reihe "‘Numerische Verfahren"’, July, 23rd 1940, page 23:



Man kann nun die Vektoren $\mathfrak{z}_\nu^{(\nu-1)}$ $(\nu = 1,2,\dots,n)$ ebenfalls in einer Matrix zusammenfassen, und zwar ist nach Gleichung (55) und (56)

$$(57) \qquad (\mathfrak{z}_1\,\mathfrak{z}_2'\,\mathfrak{z}_3''\cdots\mathfrak{z}_n^{(n-1)}) = \mathfrak{A}\cdot\mathfrak{Z}' = \mathfrak{Z}'\cdot\mathfrak{P},$$

worin die Matrix $\mathfrak{P}$ zur Abkürzung gesetzt ist für

$$(58) \qquad \mathfrak{P} = \begin{pmatrix} \alpha_{10} & \alpha_{20} & \cdots & \alpha_{n-1,0} & \alpha_{n0} \\ 1 & \alpha_{21} & \cdots & \alpha_{n-1,1} & \alpha_{n1} \\ 0 & 1 & \cdots & \alpha_{n-1,2} & \alpha_{n2} \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & \alpha_{n,n-1} \end{pmatrix}.$$

- Hessenberg decomposition, Eqn. (57),
- Hessenberg matrix, Eqn. (58).

Karl Hessenberg (* September 8th, 1904, † February 22nd, 1959)

Important Polynomials

Residuals of OR and MR approximation

$$\mathbf{x}_k := \mathbf{Q}_k \mathbf{z}_k \quad \text{and} \quad \underline{\mathbf{x}}_k := \mathbf{Q}_k \underline{\mathbf{z}}_k$$

with coefficient vectors

$$\mathbf{z}_k := \mathbf{H}_k^{-1} \mathbf{e}_1 \|\mathbf{r}_0\| \quad \text{and} \quad \underline{\mathbf{z}}_k := \underline{\mathbf{H}}_k^{\dagger} \underline{\mathbf{e}}_1 \|\mathbf{r}_0\|$$

Important Polynomials

Residuals of OR and MR approximation

$$\mathbf{x}_k := \mathbf{Q}_k \mathbf{z}_k \quad \text{and} \quad \underline{\mathbf{x}}_k := \mathbf{Q}_k \underline{\mathbf{z}}_k$$

with coefficient vectors

$$\mathbf{z}_k := \mathbf{H}_k^{-1} \mathbf{e}_1 \|\mathbf{r}_0\| \quad \text{and} \quad \underline{\mathbf{z}}_k := \underline{\mathbf{H}}_k^{\dagger} \underline{\mathbf{e}}_1 \|\mathbf{r}_0\|$$

satisfy

$$\mathbf{r}_k := \mathbf{r}_0 - \mathbf{A}\mathbf{x}_k = \mathcal{R}_k(\mathbf{A})\mathbf{r}_0 \quad \text{and} \quad \underline{\mathbf{r}}_k := \mathbf{r}_0 - \mathbf{A}\underline{\mathbf{x}}_k = \underline{\mathcal{R}}_k(\mathbf{A})\mathbf{r}_0.$$

Important Polynomials

Residuals of OR and MR approximation

$$\mathbf{x}_k := \mathbf{Q}_k \mathbf{z}_k \qquad \text{and} \qquad \underline{\mathbf{x}}_k := \mathbf{Q}_k \underline{\mathbf{z}}_k$$

with coefficient vectors

$$\mathbf{z}_k := \mathbf{H}_k^{-1} \mathbf{e}_1 \|\mathbf{r}_0\| \qquad \text{and} \qquad \underline{\mathbf{z}}_k := \underline{\mathbf{H}}_k^{\dagger} \underline{\mathbf{e}}_1 \|\mathbf{r}_0\|$$

satisfy

$$\mathbf{r}_k := \mathbf{r}_0 - \mathbf{A}\mathbf{x}_k = \mathcal{R}_k(\mathbf{A})\mathbf{r}_0 \qquad \text{and} \qquad \underline{\mathbf{r}}_k := \mathbf{r}_0 - \mathbf{A}\underline{\mathbf{x}}_k = \underline{\mathcal{R}}_k(\mathbf{A})\mathbf{r}_0.$$

Residual polynomials $\mathcal{R}_k$, $\underline{\mathcal{R}}_k$ given by

$$\mathcal{R}_k(z) := \det\left(\mathbf{I}_k - z\mathbf{H}_k^{-1}\right) \qquad \text{and} \qquad \underline{\mathcal{R}}_k(z) := \det\left(\mathbf{I}_k - z\underline{\mathbf{H}}_k^{\dagger}\underline{\mathbf{I}}_k\right).$$

Important Polynomials

Residuals of OR and MR approximation

$$\mathbf{x}_k := \mathbf{Q}_k \mathbf{z}_k \quad \text{and} \quad \underline{\mathbf{x}}_k := \mathbf{Q}_k \underline{\mathbf{z}}_k$$

with coefficient vectors

$$\mathbf{z}_k := \mathbf{H}_k^{-1} \mathbf{e}_1 \|\mathbf{r}_0\| \quad \text{and} \quad \underline{\mathbf{z}}_k := \underline{\mathbf{H}}_k^{\dagger} \underline{\mathbf{e}}_1 \|\mathbf{r}_0\|$$

satisfy

$$\mathbf{r}_k := \mathbf{r}_0 - \mathbf{A}\mathbf{x}_k = \mathcal{R}_k(\mathbf{A})\mathbf{r}_0 \quad \text{and} \quad \underline{\mathbf{r}}_k := \mathbf{r}_0 - \mathbf{A}\underline{\mathbf{x}}_k = \underline{\mathcal{R}}_k(\mathbf{A})\mathbf{r}_0.$$

Residual polynomials $\mathcal{R}_k$, $\underline{\mathcal{R}}_k$ given by

$$\mathcal{R}_k(z) := \det\left(\mathbf{I}_k - z\mathbf{H}_k^{-1}\right) \quad \text{and} \quad \underline{\mathcal{R}}_k(z) := \det\left(\mathbf{I}_k - z\underline{\mathbf{H}}_k^{\dagger}\underline{\mathbf{I}}_k\right).$$

Convergence of OR and MR depends on (harmonic) Ritz values.

Perturbed OR methods

Setting changes when perturbations enter the stage, here, OR method.

Perturbed OR methods

Setting changes when perturbations enter the stage, here, OR method.

In perturbed case

$$\mathbf{A}\mathbf{Q}_k + \mathbf{F}_k = \mathbf{Q}_{k+1}\underline{\mathbf{H}}_k$$

polynomial representation

$$\mathbf{r}_k = \mathcal{R}_k(\mathbf{A})\mathbf{r}_0 - \sum_{\ell=1}^{k} z_{\ell k}\mathcal{R}_{\ell+1:k}(\mathbf{A})\mathbf{f}_\ell + \mathbf{F}_k\mathbf{z}_k$$

(all trailing square Hessenberg matrices are assumed to be regular).

Perturbed OR methods

Setting changes when perturbations enter the stage, here, OR method.

In perturbed case

$$\mathbf{A}\mathbf{Q}_k + \mathbf{F}_k = \mathbf{Q}_{k+1}\underline{\mathbf{H}}_k$$

polynomial representation

$$\mathbf{r}_k = \mathcal{R}_k(\mathbf{A})\mathbf{r}_0 - \sum_{\ell=1}^{k} z_{\ell k}\mathcal{R}_{\ell+1:k}(\mathbf{A})\mathbf{f}_\ell + \mathbf{F}_k\mathbf{z}_k$$

(all trailing square Hessenberg matrices are assumed to be regular).

Here,

$$\mathcal{R}_{\ell+1:k}(z) := \det\left(\mathbf{I}_{k-\ell} - z\mathbf{H}_{\ell+1:k}^{-1}\right).$$

Perturbed OR methods

Setting changes when perturbations enter the stage, here, OR method.

In perturbed case

$$\mathbf{A}\mathbf{Q}_k + \mathbf{F}_k = \mathbf{Q}_{k+1}\underline{\mathbf{H}}_k$$

polynomial representation

$$\mathbf{r}_k = \mathcal{R}_k(\mathbf{A})\mathbf{r}_0 - \sum_{\ell=1}^{k} z_{\ell k}\mathcal{R}_{\ell+1:k}(\mathbf{A})\mathbf{f}_\ell + \mathbf{F}_k\mathbf{z}_k$$

(all trailing square Hessenberg matrices are assumed to be regular).

Here,

$$\mathcal{R}_{\ell+1:k}(z) := \det\left(\mathbf{I}_{k-\ell} - z\mathbf{H}_{\ell+1:k}^{-1}\right).$$

Convergence: $\mathbf{F}_k\mathbf{z}_k$ bounded (inexact methods) & $\mathcal{R}_{\ell+1:k}(\mathbf{A})$ “small”.

Outline

IDR: History repeating

	IDR
1976	Idea by Sonneveld
1979	First talk on IDR
1980	Proceedings
1989	CGS
1992	IDR $\rightsquigarrow$ BiCGStab
1993	BiCGStab2, BiCGStab(ℓ)
later	“acronym explosion” ...

IDR: History repeating

IDR		IDR(s)	
1976	Idea by Sonneveld	2006	Sonneveld & van Gijzen
1979	First talk on IDR	2007	First presentation & report
1980	Proceedings	2008	SIAM paper (SISC)
1989	CGS	2008	IDR(s)BIO
1992	IDR $\rightsquigarrow$ BICGSTAB	2010	IDR(s)STAB(ℓ), IDREIG
1993	BICGSTAB2, BICGSTAB(ℓ)	2011	flexible & multi-shift QMRIDR
later	"acronym explosion" …	later	"acronym explosion"?

IDR: History repeating

IDR		IDR(s)	
1976	Idea by Sonneveld	2006	Sonneveld & van Gijzen
1979	First talk on IDR	2007	First presentation & report
1980	Proceedings	2008	SIAM paper (SISC)
1989	CGS	2008	IDR(s)BIO
1992	IDR $\rightsquigarrow$ BICGSTAB	2010	IDR(s)STAB(ℓ), IDREIG
1993	BICGSTAB2, BICGSTAB(ℓ)	2011	flexible & multi-shift QMRIDR
later	"acronym explosion" . . .	later	"acronym explosion"?

- IDR and IDR based methods are old ($\rightsquigarrow$ my generation),
- IDR(s) is 5 years "old" ($\rightsquigarrow$ my son's generation).

IDR: History repeating

	IDR		IDR(s)
1976	Idea by Sonneveld	2006	Sonneveld & van Gijzen
1979	First talk on IDR	2007	First presentation & report
1980	Proceedings	2008	SIAM paper (SISC)
1989	CGS	2008	IDR(s)BiO
1992	IDR $\rightsquigarrow$ BiCGStab	2010	IDR(s)Stab(ℓ), IDREig
1993	BiCGStab2, BiCGStab(ℓ)	2011	flexible & multi-shift QMRIDR
later	"acronym explosion" ...	later	"acronym explosion"?

- IDR and IDR based methods are old ($\rightsquigarrow$ my generation),
- IDR(s) is 5 years "old" ($\rightsquigarrow$ my son's generation).

IDR is based on Lanczos's method; IDR(s) is based on Lanczos$(s, 1)$.

IDR: History repeating

	IDR		IDR(s)
1976	Idea by Sonneveld	2006	Sonneveld & van Gijzen
1979	First talk on IDR	2007	First presentation & report
1980	Proceedings	2008	SIAM paper (SISC)
1989	CGS	2008	IDR(s)BIO
1992	IDR $\rightsquigarrow$ BICGSTAB	2010	IDR(s)STAB(ℓ), IDREIG
1993	BICGSTAB2, BICGSTAB(ℓ)	2011	flexible & multi-shift QMRIDR
later	"acronym explosion" . . .	later	"acronym explosion"?

- IDR and IDR based methods are old ($\rightsquigarrow$ my generation),
- IDR(s) is 5 years "old" ($\rightsquigarrow$ my son's generation).

IDR is based on Lanczos's method; IDR(s) is based on Lanczos$(s, 1)$.

IDR(s) is a Krylov subspace method $\rightsquigarrow$ all techniques from 90's applicable!

IDR(s)

IDR spaces:

$$\mathcal{G}_0 := \mathcal{K}(\mathbf{A}, \mathbf{q}), \qquad \text{(full Krylov subspace)}$$
$$\mathcal{G}_j := (\alpha_j \mathbf{A} + \beta_j \mathbf{I})(\mathcal{G}_{j-1} \cap \mathcal{S}), \quad j \geqslant 1, \quad \alpha_j, \beta_j \in \mathbb{C}, \quad \alpha_j \neq 0,$$

where

$$\operatorname{codim}(\mathcal{S}) = s, \quad \text{e.g.,} \quad \mathcal{S} = \operatorname{span}\{\widetilde{\mathbf{R}}_0\}^\perp, \quad \widetilde{\mathbf{R}}_0 \in \mathbb{C}^{n \times s}.$$

IDR(s)

IDR spaces:

$$\mathcal{G}_0 := \mathcal{K}(\mathbf{A}, \mathbf{q}), \qquad \text{(full Krylov subspace)}$$
$$\mathcal{G}_j := (\alpha_j \mathbf{A} + \beta_j \mathbf{I})(\mathcal{G}_{j-1} \cap \mathcal{S}), \quad j \geqslant 1, \quad \alpha_j, \beta_j \in \mathbb{C}, \quad \alpha_j \neq 0,$$

where

$$\operatorname{codim}(\mathcal{S}) = s, \quad \text{e.g.,} \quad \mathcal{S} = \operatorname{span}\{\widetilde{\mathbf{R}}_0\}^\perp, \quad \widetilde{\mathbf{R}}_0 \in \mathbb{C}^{n \times s}.$$

Interpreted as Sonneveld spaces (Sleijpen, Sonneveld, van Gijzen 2010):

$$\mathcal{G}_j = \mathcal{S}_j(P_j, \mathbf{A}, \widetilde{\mathbf{R}}_0) := \left\{ P_j(\mathbf{A})\mathbf{v} \mid \mathbf{v} \perp \mathcal{K}_j(\mathbf{A}^\mathsf{H}, \widetilde{\mathbf{R}}_0) \right\},$$
$$P_j(z) := \prod_{i=1}^{j} (\alpha_i z + \beta_i).$$

IDR(s)

IDR spaces:

$$\mathcal{G}_0 := \mathcal{K}(\mathbf{A}, \mathbf{q}), \qquad \text{(full Krylov subspace)}$$
$$\mathcal{G}_j := (\alpha_j \mathbf{A} + \beta_j \mathbf{I})(\mathcal{G}_{j-1} \cap \mathcal{S}), \quad j \geqslant 1, \quad \alpha_j, \beta_j \in \mathbb{C}, \quad \alpha_j \neq 0,$$

where

$$\operatorname{codim}(\mathcal{S}) = s, \quad \text{e.g.,} \quad \mathcal{S} = \operatorname{span}\{\widetilde{\mathbf{R}}_0\}^{\perp}, \quad \widetilde{\mathbf{R}}_0 \in \mathbb{C}^{n \times s}.$$

Interpreted as Sonneveld spaces (Sleijpen, Sonneveld, van Gijzen 2010):

$$\mathcal{G}_j = \mathcal{S}_j(P_j, \mathbf{A}, \widetilde{\mathbf{R}}_0) := \left\{ P_j(\mathbf{A})\mathbf{v} \mid \mathbf{v} \perp \mathcal{K}_j(\mathbf{A}^{\mathsf{H}}, \widetilde{\mathbf{R}}_0) \right\},$$
$$P_j(z) := \prod_{i=1}^{j} (\alpha_i z + \beta_i).$$

Image of shrinking space: Induced Dimension Reduction.

IDR(s)

IDR spaces nested:

$$\{\mathbf{o}\} = \mathcal{G}_{\text{jmax}} \subsetneq \cdots \subsetneq \mathcal{G}_{j+1} \subsetneq \mathcal{G}_j \subsetneq \mathcal{G}_{j-1} \subsetneq \cdots \subsetneq \mathcal{G}_2 \subsetneq \mathcal{G}_1 \subsetneq \mathcal{G}_0.$$

IDR(s)

IDR spaces nested:

$$\{\mathbf{o}\} = \mathcal{G}_{\text{jmax}} \subsetneq \cdots \subsetneq \mathcal{G}_{j+1} \subsetneq \mathcal{G}_j \subsetneq \mathcal{G}_{j-1} \subsetneq \cdots \subsetneq \mathcal{G}_2 \subsetneq \mathcal{G}_1 \subsetneq \mathcal{G}_0.$$

How many vectors in $\mathcal{G}_j \setminus \mathcal{G}_{j+1}$? In generic case, $s+1$.

IDR(s)

IDR spaces nested:

$$\{\mathbf{o}\} = \mathcal{G}_{\text{jmax}} \subsetneq \cdots \subsetneq \mathcal{G}_{j+1} \subsetneq \mathcal{G}_j \subsetneq \mathcal{G}_{j-1} \subsetneq \cdots \subsetneq \mathcal{G}_2 \subsetneq \mathcal{G}_1 \subsetneq \mathcal{G}_0.$$

How many vectors in $\mathcal{G}_j \setminus \mathcal{G}_{j+1}$? In generic case, $s+1$.

Stable basis: Partially orthonormalize basis vectors $\mathbf{g}_k$, $1 \leqslant k \leqslant n$:

IDR(s)

IDR spaces nested:

$$\{\mathbf{o}\} = \mathcal{G}_{\mathrm{jmax}} \subsetneq \cdots \subsetneq \mathcal{G}_{j+1} \subsetneq \mathcal{G}_j \subsetneq \mathcal{G}_{j-1} \subsetneq \cdots \subsetneq \mathcal{G}_2 \subsetneq \mathcal{G}_1 \subsetneq \mathcal{G}_0.$$

How many vectors in $\mathcal{G}_j \setminus \mathcal{G}_{j+1}$? In generic case, $s+1$.

Stable basis: Partially orthonormalize basis vectors $\mathbf{g}_k$, $1 \leqslant k \leqslant n$:

Arnoldi: compute orthonormal basis $\mathbf{G}_{s+1}$ of $\mathcal{K}_{s+1} \subset \mathcal{G}_0$,

$$\mathbf{A}\mathbf{V}_s = \mathbf{A}\mathbf{G}_s = \mathbf{G}_{s+1}\underline{\mathbf{H}}_s, \quad \mathbf{V}_s := \mathbf{G}_s.$$

IDR(s)

IDR spaces nested:

$$\{\mathbf{o}\} = \mathcal{G}_{\mathrm{jmax}} \subsetneq \cdots \subsetneq \mathcal{G}_{j+1} \subsetneq \mathcal{G}_j \subsetneq \mathcal{G}_{j-1} \subsetneq \cdots \subsetneq \mathcal{G}_2 \subsetneq \mathcal{G}_1 \subsetneq \mathcal{G}_0.$$

How many vectors in $\mathcal{G}_j \setminus \mathcal{G}_{j+1}$? In generic case, $s+1$.

Stable basis: Partially orthonormalize basis vectors $\mathbf{g}_k$, $1 \leqslant k \leqslant n$:

Arnoldi: compute orthonormal basis $\mathbf{G}_{s+1}$ of $\mathcal{K}_{s+1} \subset \mathcal{G}_0$,

$$\mathbf{A}\mathbf{V}_s = \mathbf{A}\mathbf{G}_s = \mathbf{G}_{s+1}\underline{\mathbf{H}}_s, \quad \mathbf{V}_s := \mathbf{G}_s.$$

"Lanczos": perform intersection $\mathcal{G}_j \cap \mathcal{S}$, map, and orthonormalize,

$$\mathbf{v}_k = \sum_{i=k-s}^{k} \mathbf{g}_i \gamma_i, \quad \widetilde{\mathbf{R}}_0^{\mathsf{H}} \mathbf{v}_k = \mathbf{o}_s, \quad k \geqslant s+1,$$

IDR(s)

IDR spaces nested:

$$\{\mathbf{o}\} = \mathcal{G}_{\mathrm{jmax}} \subsetneq \cdots \subsetneq \mathcal{G}_{j+1} \subsetneq \mathcal{G}_j \subsetneq \mathcal{G}_{j-1} \subsetneq \cdots \subsetneq \mathcal{G}_2 \subsetneq \mathcal{G}_1 \subsetneq \mathcal{G}_0.$$

How many vectors in $\mathcal{G}_j \setminus \mathcal{G}_{j+1}$? In generic case, $s+1$.

Stable basis: Partially orthonormalize basis vectors $\mathbf{g}_k$, $1 \leqslant k \leqslant n$:

Arnoldi: compute orthonormal basis $\mathbf{G}_{s+1}$ of $\mathcal{K}_{s+1} \subset \mathcal{G}_0$,

$$\mathbf{A}\mathbf{V}_s = \mathbf{A}\mathbf{G}_s = \mathbf{G}_{s+1}\underline{\mathbf{H}}_s, \quad \mathbf{V}_s := \mathbf{G}_s.$$

"Lanczos": perform intersection $\mathcal{G}_j \cap \mathcal{S}$, map, and orthonormalize,

$$\mathbf{v}_k = \sum_{i=k-s}^{k} \mathbf{g}_i \gamma_i, \quad \widetilde{\mathbf{R}}_0^{\mathsf{H}} \mathbf{v}_k = \mathbf{o}_s, \quad k \geqslant s+1,$$

$$(\alpha_j \mathbf{A} + \beta_j \mathbf{I})\mathbf{v}_k \qquad , \quad j = \left\lfloor \frac{k-1}{s+1} \right\rfloor.$$

IDR(s)

IDR spaces nested:

$$\{\mathbf{o}\} = \mathcal{G}_{\text{jmax}} \subsetneq \cdots \subsetneq \mathcal{G}_{j+1} \subsetneq \mathcal{G}_j \subsetneq \mathcal{G}_{j-1} \subsetneq \cdots \subsetneq \mathcal{G}_2 \subsetneq \mathcal{G}_1 \subsetneq \mathcal{G}_0.$$

How many vectors in $\mathcal{G}_j \setminus \mathcal{G}_{j+1}$? In generic case, $s+1$.

Stable basis: Partially orthonormalize basis vectors $\mathbf{g}_k$, $1 \leqslant k \leqslant n$:

Arnoldi: compute orthonormal basis $\mathbf{G}_{s+1}$ of $\mathcal{K}_{s+1} \subset \mathcal{G}_0$,

$$\mathbf{A}\mathbf{V}_s = \mathbf{A}\mathbf{G}_s = \mathbf{G}_{s+1}\underline{\mathbf{H}}_s, \quad \mathbf{V}_s := \mathbf{G}_s.$$

"Lanczos": perform intersection $\mathcal{G}_j \cap \mathcal{S}$, map, and orthonormalize,

$$\mathbf{v}_k = \sum_{i=k-s}^{k} \mathbf{g}_i \gamma_i, \quad \widetilde{\mathbf{R}}_0^{\mathsf{H}} \mathbf{v}_k = \mathbf{o}_s, \quad k \geqslant s+1,$$

$$\mathbf{g}_{k+1}\nu_{k+1} = (\alpha_j \mathbf{A} + \beta_j \mathbf{I})\mathbf{v}_k - \sum_{i=k-j(s+1)-1}^{k} \mathbf{g}_i \nu_i, \quad j = \left\lfloor \frac{k-1}{s+1} \right\rfloor.$$

IDR(s)

Generalized Hessenberg decomposition:

$$\mathbf{A}\mathbf{V}_k = \mathbf{A}\mathbf{G}_k\mathbf{U}_k = \mathbf{G}_{k+1}\underline{\mathbf{H}}_k,$$

where $\mathbf{U}_k \in \mathbb{C}^{k\times k}$ upper triangular.

IDR(s)

Generalized Hessenberg decomposition:

$$\mathbf{A}\mathbf{V}_k = \mathbf{A}\mathbf{G}_k\mathbf{U}_k = \mathbf{G}_{k+1}\underline{\mathbf{H}}_k,$$

where $\mathbf{U}_k \in \mathbb{C}^{k\times k}$ upper triangular.

Structure of Sonneveld pencils:

$$\mathbf{H}_k = \left(\begin{array}{cccccccccccc}
\times & \times & \times & \times & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ \\
+ & \times & \times & \times & \times & \circ & \circ & \circ & \circ & \circ & \circ & \circ \\
\circ & + & \times & \times & \times & \times & \circ & \circ & \circ & \circ & \circ & \circ \\
\circ & \circ & + & \times & \times & \times & \times & \circ & \circ & \circ & \circ & \circ \\
\circ & \circ & \circ & + & \times & \times & \times & \times & \circ & \circ & \circ & \circ \\
\circ & \circ & \circ & \circ & + & \times & \times & \times & \times & \circ & \circ & \circ \\
\circ & \circ & \circ & \circ & \circ & + & \times & \times & \times & \times & \circ & \circ \\
\circ & \circ & \circ & \circ & \circ & \circ & + & \times & \times & \times & \times & \circ \\
\circ & \circ & \circ & \circ & \circ & \circ & \circ & + & \times & \times & \times & \times \\
\circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & + & \times & \times & \times \\
\circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & + & \times & \times \\
\circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & + & \times
\end{array}\right), \quad
\left(\begin{array}{cccccccccccc}
\times & \times & \times & \times & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ \\
\circ & \times & \times & \times & \times & \circ & \circ & \circ & \circ & \circ & \circ & \circ \\
\circ & \circ & \times & \times & \times & \times & \circ & \circ & \circ & \circ & \circ & \circ \\
\circ & \circ & \circ & \times & \times & \times & \times & \circ & \circ & \circ & \circ & \circ \\
\circ & \circ & \circ & \circ & \times & \times & \times & \times & \circ & \circ & \circ & \circ \\
\circ & \circ & \circ & \circ & \circ & \times & \times & \times & \times & \circ & \circ & \circ \\
\circ & \circ & \circ & \circ & \circ & \circ & \times & \times & \times & \times & \circ & \circ \\
\circ & \circ & \circ & \circ & \circ & \circ & \circ & \times & \times & \times & \times & \circ \\
\circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \times & \times & \times & \times \\
\circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \times & \times & \times \\
\circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \times & \times \\
\circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \times
\end{array}\right) = \mathbf{U}_k$$

IDREIG

Eigenvalues of Sonneveld pencil $(\mathbf{H}_k, \mathbf{U}_k)$ are roots of residual polynomials. Those distinct from roots of

$$P_j(z) = \prod_{i=1}^{j} (\alpha_i z + \beta_i), \quad \text{i.e.,} \quad z_i = -\frac{\beta_i}{\alpha_i}, \quad 1 \leqslant i \leqslant j$$

converge to eigenvalues of $\mathbf{A}$.

IDREIG

Eigenvalues of Sonneveld pencil $(\mathbf{H}_k, \mathbf{U}_k)$ are roots of residual polynomials. Those distinct from roots of

$$P_j(z) = \prod_{i=1}^{j} (\alpha_i z + \beta_i), \quad \text{i.e.,} \quad z_i = -\frac{\beta_i}{\alpha_i}, \quad 1 \leqslant i \leqslant j$$

converge to eigenvalues of $\mathbf{A}$.

Suppose $\mathbf{G}_{k+1}$ of full rank. Sonneveld pencil $(\mathbf{H}_k, \mathbf{U}_k)$ as oblique projection:

$$\begin{aligned} \widehat{\mathbf{G}}_k^{\mathsf{H}}(\mathbf{A}, \mathbf{I}_n)\mathbf{G}_k\mathbf{U}_k &= \widehat{\mathbf{G}}_k^{\mathsf{H}}(\mathbf{A}\mathbf{G}_k\mathbf{U}_k, \mathbf{G}_k\mathbf{U}_k) \\ &= \widehat{\mathbf{G}}_k^{\mathsf{H}}(\mathbf{G}_{k+1}\underline{\mathbf{H}}_k, \mathbf{G}_k\mathbf{U}_k) = (\underline{\mathbf{I}}_k^{\mathsf{T}}\underline{\mathbf{H}}_k, \mathbf{U}_k) = (\mathbf{H}_k, \mathbf{U}_k), \end{aligned} \tag{1}$$

here, $\widehat{\mathbf{G}}_k^{\mathsf{H}} := \underline{\mathbf{I}}_k^{\mathsf{T}}\mathbf{G}_{k+1}^{\dagger}$.

IDREig

Eigenvalues of Sonneveld pencil $(\mathbf{H}_k, \mathbf{U}_k)$ are roots of residual polynomials. Those distinct from roots of

$$P_j(z) = \prod_{i=1}^{j} (\alpha_i z + \beta_i), \quad \text{i.e.,} \quad z_i = -\frac{\beta_i}{\alpha_i}, \quad 1 \leqslant i \leqslant j$$

converge to eigenvalues of $\mathbf{A}$.

Suppose $\mathbf{G}_{k+1}$ of full rank. Sonneveld pencil $(\mathbf{H}_k, \mathbf{U}_k)$ as oblique projection:

$$\begin{aligned} \widehat{\mathbf{G}}_k^{\mathsf{H}}(\mathbf{A}, \mathbf{I}_n)\mathbf{G}_k\mathbf{U}_k &= \widehat{\mathbf{G}}_k^{\mathsf{H}}(\mathbf{A}\mathbf{G}_k\mathbf{U}_k, \mathbf{G}_k\mathbf{U}_k) \\ &= \widehat{\mathbf{G}}_k^{\mathsf{H}}(\mathbf{G}_{k+1}\underline{\mathbf{H}}_k, \mathbf{G}_k\mathbf{U}_k) = (\underline{\mathbf{I}}_k^{\mathsf{T}}\underline{\mathbf{H}}_k, \mathbf{U}_k) = (\mathbf{H}_k, \mathbf{U}_k), \end{aligned} \tag{1}$$

here, $\widehat{\mathbf{G}}_k^{\mathsf{H}} := \underline{\mathbf{I}}_k^{\mathsf{T}}\mathbf{G}_{k+1}^{\dagger}$.

Use deflated pencil for Lanczos Ritz values (Gutknecht, Z. (2010): IDREig).

IDREIG

Eigenvalues of Sonneveld pencil $(\mathbf{H}_k, \mathbf{U}_k)$ are roots of residual polynomials. Those distinct from roots of

$$P_j(z) = \prod_{i=1}^{j} (\alpha_i z + \beta_i), \quad \text{i.e.,} \quad z_i = -\frac{\beta_i}{\alpha_i}, \quad 1 \leqslant i \leqslant j$$

converge to eigenvalues of $\mathbf{A}$.

Suppose $\mathbf{G}_{k+1}$ of full rank. Sonneveld pencil $(\mathbf{H}_k, \mathbf{U}_k)$ as oblique projection:

$$\begin{aligned}\widehat{\mathbf{G}}_k^{\mathsf{H}}(\mathbf{A}, \mathbf{I}_n)\mathbf{G}_k\mathbf{U}_k &= \widehat{\mathbf{G}}_k^{\mathsf{H}}(\mathbf{A}\mathbf{G}_k\mathbf{U}_k, \mathbf{G}_k\mathbf{U}_k) \\ &= \widehat{\mathbf{G}}_k^{\mathsf{H}}(\mathbf{G}_{k+1}\underline{\mathbf{H}}_k, \mathbf{G}_k\mathbf{U}_k) = (\underline{\mathbf{I}}_k^{\mathsf{T}}\underline{\mathbf{H}}_k, \mathbf{U}_k) = (\mathbf{H}_k, \mathbf{U}_k),\end{aligned} \tag{1}$$

here, $\widehat{\mathbf{G}}_k^{\mathsf{H}} := \underline{\mathbf{I}}_k^{\mathsf{T}}\mathbf{G}_{k+1}^{\dagger}$.

Use deflated pencil for Lanczos Ritz values (Gutknecht, Z. (2010): IDREIG). First: IDR(s)ORES, Olaf Rendel: IDR(s)BIO, Anisa Rizvanolli: IDR(s)STAB(ℓ).

IDRSTAB

IDR(s)STAB(ℓ) (Tanio & Sugihara; Sleijpen & van Gijzen): combine ideas of IDR(s) and BiCGSTAB(ℓ).

IDRSTAB

IDR(s)STAB(ℓ) (Tanio & Sugihara; Sleijpen & van Gijzen): combine ideas of IDR(s) and BICGSTAB(ℓ).

IDRSTAB (Sleijpen's implementation) recursively computes "(extended) Hessenberg matrices of basis matrices and residuals" ($k \geqslant 1$):

$$
\begin{array}{llll}
\mathbf{G}_{11}^{(k)}, \mathbf{r}_{11}^{(k)} & \mathbf{G}_{12}^{(k)}, \mathbf{r}_{12}^{(k)} & \cdots & \mathbf{G}_{1,\ell+1}^{(k)}, \quad \mathbf{r}_{1,\ell+1}^{(k)} \\
\mathbf{G}_{21}^{(k)}, \mathbf{r}_{21}^{(k)} & \mathbf{G}_{22}^{(k)}, \mathbf{r}_{22}^{(k)} & \cdots & \mathbf{G}_{2,\ell+1}^{(k)}, \quad \mathbf{r}_{2,\ell+1}^{(k)} \\
 & \mathbf{G}_{32}^{(k)}, \mathbf{r}_{32}^{(k)} & \ddots & \vdots \\
 & & \ddots & \mathbf{G}_{\ell+1,\ell+1}^{(k)}, \mathbf{r}_{\ell+1,\ell+1}^{(k)} \\
 & & & \mathbf{G}_{\ell+2,\ell+1}^{(k)}
\end{array}
$$

$$
\begin{aligned}
\mathbf{G}_{i,j}^{(k)} &\in \mathbb{C}^{n\times s}, & \mathbf{r}_{i,j}^{(k)} &\in \mathbb{C}^{n}, \\
\mathbf{G}_{i+1,j}^{(k)} &= \mathbf{A}\mathbf{G}_{i,j}^{(k)}, & \mathbf{r}_{i+1,j}^{(k)} &= \mathbf{A}\mathbf{r}_{i,j}^{(k)}, \\
\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{G}_{ii}^{(k)} &= \mathbf{O}_s, & \widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{r}_{ii}^{(k)} &= \mathbf{o}_s, \\
(\mathbf{G}_{ii}^{(k)})^{\mathsf{H}}\mathbf{G}_{ii}^{(k)} &= \mathbf{I}_s.
\end{aligned}
$$

IDRSTAB

IDR(s)STAB(ℓ) (Tanio & Sugihara; Sleijpen & van Gijzen): combine ideas of IDR(s) and BICGSTAB(ℓ).

IDRSTAB (Sleijpen's implementation) recursively computes "(extended) Hessenberg matrices of basis matrices and residuals" ($k \geqslant 1$):

$$
\begin{array}{llll}
\mathbf{G}_{11}^{(k)}, \mathbf{r}_{11}^{(k)} & \mathbf{G}_{12}^{(k)}, \mathbf{r}_{12}^{(k)} & \cdots & \mathbf{G}_{1,\ell+1}^{(k)}, \quad \mathbf{r}_{1,\ell+1}^{(k)} \\
\mathbf{G}_{21}^{(k)}, \mathbf{r}_{21}^{(k)} & \mathbf{G}_{22}^{(k)}, \mathbf{r}_{22}^{(k)} & \cdots & \mathbf{G}_{2,\ell+1}^{(k)}, \quad \mathbf{r}_{2,\ell+1}^{(k)} \\
 & \mathbf{G}_{32}^{(k)}, \mathbf{r}_{32}^{(k)} & \ddots & \qquad\qquad \vdots \\
 & & \ddots & \mathbf{G}_{\ell+1,\ell+1}^{(k)}, \mathbf{r}_{\ell+1,\ell+1}^{(k)} \\
 & & & \mathbf{G}_{\ell+2,\ell+1}^{(k)}
\end{array}
$$

$$
\begin{aligned}
\mathbf{G}_{i,j}^{(k)} \in \mathbb{C}^{n\times s}, &\quad \mathbf{r}_{i,j}^{(k)} \in \mathbb{C}^{n}, \\
\mathbf{G}_{i+1,j}^{(k)} = \mathbf{A}\mathbf{G}_{i,j}^{(k)}, &\quad \mathbf{r}_{i+1,j}^{(k)} = \mathbf{A}\mathbf{r}_{i,j}^{(k)}, \\
\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{G}_{ii}^{(k)} = \mathbf{O}_s, &\quad \widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{r}_{ii}^{(k)} = \mathbf{o}_s, \\
(\mathbf{G}_{ii}^{(k)})^{\mathsf{H}}\mathbf{G}_{ii}^{(k)} &= \mathbf{I}_s.
\end{aligned}
$$

Initialization using Arnoldi's method:

$$
\mathbf{G}_{21}^{(1)} = \mathbf{A}\mathbf{G}_{11}^{(1)} = (\mathbf{G}_{11}^{(1)}, \mathbf{g}_{\mathsf{tmp}})\underline{\mathbf{H}}_s^{(0)},
$$

$$
\mathbf{r}_{11}^{(1)} = \mathbf{r}_0 - \mathbf{G}_{21}^{(1)}\boldsymbol{\alpha}^{(1)} = (\mathbf{I} - \mathbf{G}_{21}^{(1)}(\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{G}_{21}^{(1)})^{-1}\widetilde{\mathbf{R}}_0^{\mathsf{H}})\mathbf{r}_0, \quad \mathbf{r}_{21}^{(1)} = \mathbf{A}\mathbf{r}_{11}^{(1)}.
$$

IDRSTAB

Columnwise update (IDR part) such that diagonal blocks

- form basis of $\mathcal{G}_j \setminus \mathcal{G}_{j+1}$ with expansion $\mathcal{G}_j = \mathbf{A}(\mathcal{G}_{j-1} \cap \mathcal{S}) \rightsquigarrow \beta^{(j)} \in \mathbb{C}^{s \times s}$,
- are orthonormalized $\rightsquigarrow \underline{\mathbf{H}}_{s-1}^{(j)} \in \mathbb{C}^{s \times (s-1)}$

IDRSTAB

Columnwise update (IDR part) such that diagonal blocks

- form basis of $\mathcal{G}_j \setminus \mathcal{G}_{j+1}$ with expansion $\mathcal{G}_j = \mathbf{A}(\mathcal{G}_{j-1} \cap \mathcal{S}) \rightsquigarrow \boldsymbol{\beta}^{(j)} \in \mathbb{C}^{s \times s}$,
- are orthonormalized $\rightsquigarrow \underline{\mathbf{H}}_{s-1}^{(j)} \in \mathbb{C}^{s \times (s-1)}$

In particular, with $\widetilde{\mathbf{v}}_i \in \mathcal{G}_{j-1} \cap \mathcal{S}$,

$$
\begin{aligned}
\boldsymbol{\beta}_i^{(j)} &= (\widetilde{\mathbf{R}}_0^{\mathsf{H}} \mathbf{G}_{j,j-1})^{-1} \widetilde{\mathbf{R}}_0^{\mathsf{H}} (\mathbf{A}\widetilde{\mathbf{v}}_i) \\
\Rightarrow \quad (\mathbf{A}\widetilde{\mathbf{v}}_i) - \mathbf{G}_{j,j-1}\boldsymbol{\beta}_i^{(j)} &= \mathbf{A}(\widetilde{\mathbf{v}}_i - \mathbf{G}_{j-1,j-1}\boldsymbol{\beta}_i^{(j)}) \in \mathcal{G}_j \cap \mathcal{S}
\end{aligned}
$$

IDRSTAB

Columnwise update (IDR part) such that diagonal blocks

- form basis of $\mathcal{G}_j \setminus \mathcal{G}_{j+1}$ with expansion $\mathcal{G}_j = \mathbf{A}(\mathcal{G}_{j-1} \cap \mathcal{S}) \rightsquigarrow \beta^{(j)} \in \mathbb{C}^{s \times s}$,
- are orthonormalized $\rightsquigarrow \underline{\mathbf{H}}_{s-1}^{(j)} \in \mathbb{C}^{s \times (s-1)}$

In particular, with $\widetilde{\mathbf{v}}_i \in \mathcal{G}_{j-1} \cap \mathcal{S}$,

$$\begin{aligned} \beta_i^{(j)} &= (\widetilde{\mathbf{R}}_0^{\mathsf{H}} \mathbf{G}_{j,j-1})^{-1} \widetilde{\mathbf{R}}_0^{\mathsf{H}} (\mathbf{A}\widetilde{\mathbf{v}}_i) \\ \Rightarrow \quad (\mathbf{A}\widetilde{\mathbf{v}}_i) - \mathbf{G}_{j,j-1}\beta_i^{(j)} &= \mathbf{A}(\widetilde{\mathbf{v}}_i - \mathbf{G}_{j-1,j-1}\beta_i^{(j)}) \in \mathcal{G}_j \cap \mathcal{S} \end{aligned}$$

Every new vector in $\mathcal{G}_j \cap \mathcal{S}$ is orthonormalized with respect to the others.

IDRSTAB

Columnwise update (IDR part) such that diagonal blocks

- form basis of $\mathcal{G}_j \setminus \mathcal{G}_{j+1}$ with expansion $\mathcal{G}_j = \mathbf{A}(\mathcal{G}_{j-1} \cap \mathcal{S}) \rightsquigarrow \beta^{(j)} \in \mathbb{C}^{s \times s}$,
- are orthonormalized $\rightsquigarrow \underline{\mathbf{H}}_{s-1}^{(j)} \in \mathbb{C}^{s \times (s-1)}$

In particular, with $\widetilde{\mathbf{v}}_i \in \mathcal{G}_{j-1} \cap \mathcal{S}$,

$$\begin{aligned} \beta_i^{(j)} &= (\widetilde{\mathbf{R}}_0^{\mathsf{H}} \mathbf{G}_{j,j-1})^{-1} \widetilde{\mathbf{R}}_0^{\mathsf{H}} (\mathbf{A}\widetilde{\mathbf{v}}_i) \\ \Rightarrow \quad (\mathbf{A}\widetilde{\mathbf{v}}_i) - \mathbf{G}_{j,j-1}\beta_i^{(j)} &= \mathbf{A}(\widetilde{\mathbf{v}}_i - \mathbf{G}_{j-1,j-1}\beta_i^{(j)}) \in \mathcal{G}_j \cap \mathcal{S} \end{aligned}$$

Every new vector in $\mathcal{G}_j \cap \mathcal{S}$ is orthonormalized with respect to the others.

Thus, for the IDR-IDRSTAB pencil relating (STAB-purified) diagonal blocks,

- $\beta^{(j)} \in \mathbb{C}^{s \times s}$ couples $\mathbf{G}_{jj}$ and $\mathbf{G}_{j,j-1} = \mathbf{A}\mathbf{G}_{j-1,j-1} \rightsquigarrow \mathbf{U}_k$,
- $\underline{\mathbf{H}}_{s-1}^{(j)} \in \mathbb{C}^{s \times (s-1)}$ couples result with others in same block $\rightsquigarrow \underline{\mathbf{H}}_k$.

IDRSTAB

Columnwise update (IDR part) such that diagonal blocks

- form basis of $\mathcal{G}_j \setminus \mathcal{G}_{j+1}$ with expansion $\mathcal{G}_j = \mathbf{A}(\mathcal{G}_{j-1} \cap \mathcal{S}) \rightsquigarrow \beta^{(j)} \in \mathbb{C}^{s \times s}$,
- are orthonormalized $\rightsquigarrow \underline{\mathbf{H}}_{s-1}^{(j)} \in \mathbb{C}^{s \times (s-1)}$

In particular, with $\widetilde{\mathbf{v}}_i \in \mathcal{G}_{j-1} \cap \mathcal{S}$,

$$\begin{aligned} \beta_i^{(j)} &= (\widetilde{\mathbf{R}}_0^{\mathsf{H}} \mathbf{G}_{j,j-1})^{-1} \widetilde{\mathbf{R}}_0^{\mathsf{H}} (\mathbf{A}\widetilde{\mathbf{v}}_i) \\ \Rightarrow \quad (\mathbf{A}\widetilde{\mathbf{v}}_i) - \mathbf{G}_{j,j-1}\beta_i^{(j)} &= \mathbf{A}(\widetilde{\mathbf{v}}_i - \mathbf{G}_{j-1,j-1}\beta_i^{(j)}) \in \mathcal{G}_j \cap \mathcal{S} \end{aligned}$$

Every new vector in $\mathcal{G}_j \cap \mathcal{S}$ is orthonormalized with respect to the others.

Thus, for the IDR-IDRSTAB pencil relating (STAB-purified) diagonal blocks,

- $\beta^{(j)} \in \mathbb{C}^{s \times s}$ couples $\mathbf{G}_{jj}$ and $\mathbf{G}_{j,j-1} = \mathbf{A}\mathbf{G}_{j-1,j-1} \rightsquigarrow \mathbf{U}_k$,
- $\underline{\mathbf{H}}_{s-1}^{(j)} \in \mathbb{C}^{s \times (s-1)}$ couples result with others in same block $\rightsquigarrow \underline{\mathbf{H}}_k$.

All other blocks in column treated in same manner.

IDRSTAB

Residual updates en détail ($i \leqslant j$, $\mathbf{r}_{j+1,j}^{(k)} = \mathbf{A}\mathbf{r}_{j,j}^{(k)}$):

$$\mathbf{r}_{i,j}^{(k)} = \mathbf{r}_{i,j-1}^{(k)} - \mathbf{G}_{i+1,j}^{(k)}\alpha^{(j)}, \quad \mathbf{r}_{j,j}^{(k)} = (\mathbf{I} - \mathbf{G}_{j+1,j}^{(k)}(\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{G}_{j+1,j}^{(k)})^{-1}\widetilde{\mathbf{R}}_0^{\mathsf{H}})\mathbf{r}_{j,j-1}^{(k)}.$$

IDRSTAB

Residual updates en détail ($i \leqslant j$, $\mathbf{r}_{j+1,j}^{(k)} = \mathbf{A}\mathbf{r}_{j,j}^{(k)}$):

$$\mathbf{r}_{i,j}^{(k)} = \mathbf{r}_{i,j-1}^{(k)} - \mathbf{G}_{i+1,j}^{(k)}\alpha^{(j)}, \quad \mathbf{r}_{j,j}^{(k)} = (\mathbf{I} - \mathbf{G}_{j+1,j}^{(k)}(\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{G}_{j+1,j}^{(k)})^{-1}\widetilde{\mathbf{R}}_0^{\mathsf{H}})\mathbf{r}_{j,j-1}^{(k)}.$$

Here,

$$\alpha^{(j)} := (\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{G}_{j+1,j}^{(k)})^{-1}\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{r}_{j,j-1}^{(k)},$$

IDRSTAB

Residual updates en détail ($i \leqslant j$, $\mathbf{r}_{j+1,j}^{(k)} = \mathbf{A}\mathbf{r}_{j,j}^{(k)}$):

$$\mathbf{r}_{i,j}^{(k)} = \mathbf{r}_{i,j-1}^{(k)} - \mathbf{G}_{i+1,j}^{(k)}\alpha^{(j)}, \quad \mathbf{r}_{j,j}^{(k)} = (\mathbf{I} - \mathbf{G}_{j+1,j}^{(k)}(\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{G}_{j+1,j}^{(k)})^{-1}\widetilde{\mathbf{R}}_0^{\mathsf{H}})\mathbf{r}_{j,j-1}^{(k)}.$$

Here,

$$\alpha^{(j)} := (\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{G}_{j+1,j}^{(k)})^{-1}\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{r}_{j,j-1}^{(k)},$$

$\alpha^{(j)}$ relating $\mathbf{r}_{j,j-1}^{(k)} = \mathbf{A}\mathbf{r}_{j-1,j-1}^{(k)}$ (old) and $\mathbf{r}_{j,j}^{(k)}$ (new) via $\mathbf{G}_{j+1,j}^{(k)} = \mathbf{A}\mathbf{G}_{j,j}^{(k)} \rightsquigarrow \mathbf{U}_k$.

IDRSTAB

Residual updates en détail ($i \leqslant j$, $\mathbf{r}_{j+1,j}^{(k)} = \mathbf{A}\mathbf{r}_{j,j}^{(k)}$):

$$\mathbf{r}_{i,j}^{(k)} = \mathbf{r}_{i,j-1}^{(k)} - \mathbf{G}_{i+1,j}^{(k)}\alpha^{(j)}, \quad \mathbf{r}_{j,j}^{(k)} = (\mathbf{I} - \mathbf{G}_{j+1,j}^{(k)}(\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{G}_{j+1,j}^{(k)})^{-1}\widetilde{\mathbf{R}}_0^{\mathsf{H}})\mathbf{r}_{j,j-1}^{(k)}.$$

Here,

$$\alpha^{(j)} := (\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{G}_{j+1,j}^{(k)})^{-1}\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{r}_{j,j-1}^{(k)},$$

$\alpha^{(j)}$ relating $\mathbf{r}_{j,j-1}^{(k)} = \mathbf{A}\mathbf{r}_{j-1,j-1}^{(k)}$ (old) and $\mathbf{r}_{j,j}^{(k)}$ (new) via $\mathbf{G}_{j+1,j}^{(k)} = \mathbf{A}\mathbf{G}_{j,j}^{(k)} \rightsquigarrow \mathbf{U}_k$.

New cycle (STAB part, $\mathbf{r}_{21}^{(k+1)} = \mathbf{A}\mathbf{r}_{11}^{(k+1)}$, $\gamma^{(\ell)} \in \mathbb{C}^s$ such that $\|\mathbf{r}_{11}^{(k+1)}\| = \min$):

$$\mathbf{r}_{11}^{(k+1)} = \mathbf{r}_{1,\ell+1}^{(k)} - \sum_{i=1}^{\ell} \mathbf{r}_{i+1,\ell+1}^{(k)}\gamma_i^{(\ell)}, \quad \begin{cases} \mathbf{G}_{11}^{(k+1)} = \mathbf{G}_{1,\ell+1}^{(k)} - \sum_{i=1}^{\ell} \mathbf{G}_{i+1,\ell+1}^{(k)}\gamma_i^{(\ell)}, \\ \mathbf{G}_{21}^{(k+1)} = \mathbf{G}_{2,\ell+1}^{(k)} - \sum_{i=1}^{\ell} \mathbf{G}_{i+2,\ell+1}^{(k)}\gamma_i^{(\ell)}. \end{cases}$$

IDRSTAB

Residual updates en détail ($i \leqslant j$, $\mathbf{r}_{j+1,j}^{(k)} = \mathbf{A}\mathbf{r}_{j,j}^{(k)}$):

$$\mathbf{r}_{i,j}^{(k)} = \mathbf{r}_{i,j-1}^{(k)} - \mathbf{G}_{i+1,j}^{(k)}\boldsymbol{\alpha}^{(j)}, \quad \mathbf{r}_{j,j}^{(k)} = (\mathbf{I} - \mathbf{G}_{j+1,j}^{(k)}(\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{G}_{j+1,j}^{(k)})^{-1}\widetilde{\mathbf{R}}_0^{\mathsf{H}})\mathbf{r}_{j,j-1}^{(k)}.$$

Here,

$$\boldsymbol{\alpha}^{(j)} := (\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{G}_{j+1,j}^{(k)})^{-1}\widetilde{\mathbf{R}}_0^{\mathsf{H}}\mathbf{r}_{j,j-1}^{(k)},$$

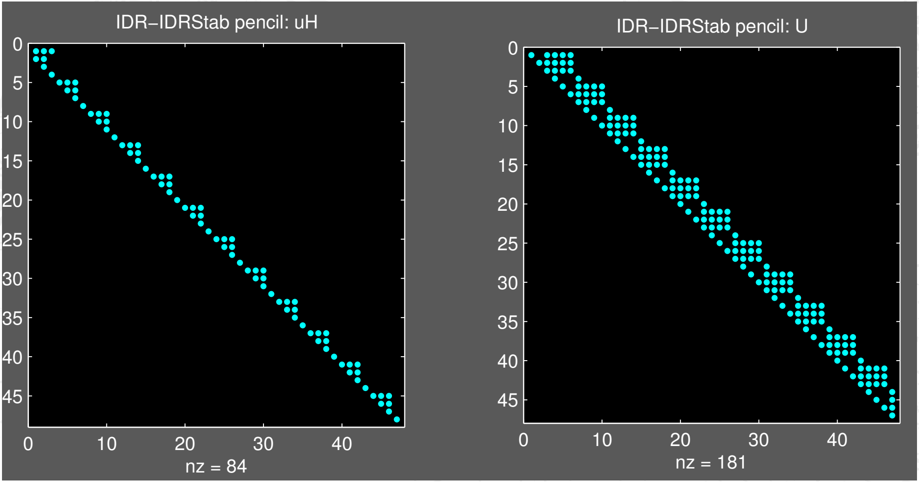
$\boldsymbol{\alpha}^{(j)}$ relating $\mathbf{r}_{j,j-1}^{(k)} = \mathbf{A}\mathbf{r}_{j-1,j-1}^{(k)}$ (old) and $\mathbf{r}_{j,j}^{(k)}$ (new) via $\mathbf{G}_{j+1,j}^{(k)} = \mathbf{A}\mathbf{G}_{j,j}^{(k)} \rightsquigarrow \mathbf{U}_k$.

New cycle (STAB part, $\mathbf{r}_{21}^{(k+1)} = \mathbf{A}\mathbf{r}_{11}^{(k+1)}$, $\gamma^{(\ell)} \in \mathbb{C}^s$ such that $\|\mathbf{r}_{11}^{(k+1)}\| = \min$):

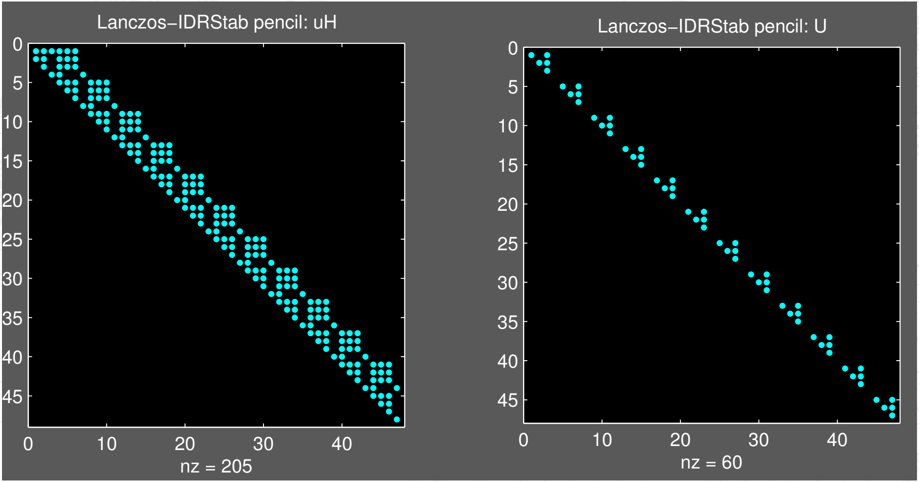
$$\mathbf{r}_{11}^{(k+1)} = \mathbf{r}_{1,\ell+1}^{(k)} - \sum_{i=1}^{\ell} \mathbf{r}_{i+1,\ell+1}^{(k)}\gamma_i^{(\ell)}, \quad \begin{cases} \mathbf{G}_{11}^{(k+1)} = \mathbf{G}_{1,\ell+1}^{(k)} - \sum_{i=1}^{\ell} \mathbf{G}_{i+1,\ell+1}^{(k)}\gamma_i^{(\ell)}, \\ \mathbf{G}_{21}^{(k+1)} = \mathbf{G}_{2,\ell+1}^{(k)} - \sum_{i=1}^{\ell} \mathbf{G}_{i+2,\ell+1}^{(k)}\gamma_i^{(\ell)}. \end{cases}$$

Anisa Rizvanolli: $\rightsquigarrow$ Lanczos-IDRSTAB pencil for eigenvalues, IDRSTABEIG.

Structure of (STAB-purified) IDR-IDRSTAB pencil



Structure of (undeflated) Lanczos-IDRSTAB pencil



QMRIDR

MR methods: use extended Hessenberg matrix

$$\underline{\mathbf{x}}_k := \mathbf{Q}_k \underline{\mathbf{z}}_k, \quad \underline{\mathbf{z}}_k := \underline{\mathbf{H}}_k^{\dagger} \underline{\mathbf{e}}_1 \|\mathbf{r}_0\|.$$

QMRIDR

MR methods: use extended Hessenberg matrix

$$\underline{\mathbf{x}}_k := \mathbf{Q}_k \underline{\mathbf{z}}_k, \quad \underline{\mathbf{z}}_k := \underline{\mathbf{H}}_k^{\dagger} \underline{\mathbf{e}}_1 \|\mathbf{r}_0\|.$$

IDR based: generalized Hessenberg decomposition,

$$\mathbf{A}\mathbf{V}_k = \mathbf{A}\mathbf{G}_k\mathbf{U}_k = \mathbf{G}_{k+1}\underline{\mathbf{H}}_k.$$

Thus,

$$\underline{\mathbf{x}}_k := \mathbf{V}_k \underline{\mathbf{z}}_k = \mathbf{G}_k \mathbf{U}_k \underline{\mathbf{z}}_k, \quad \underline{\mathbf{z}}_k := \underline{\mathbf{H}}_k^{\dagger} \underline{\mathbf{e}}_1 \|\mathbf{r}_0\|.$$

QMRIDR

MR methods: use extended Hessenberg matrix

$$\mathbf{x}_k := \mathbf{Q}_k \mathbf{z}_k, \quad \mathbf{z}_k := \underline{\mathbf{H}}_k^{\dagger} \underline{\mathbf{e}}_1 \|\mathbf{r}_0\|.$$

IDR based: generalized Hessenberg decomposition,

$$\mathbf{A}\mathbf{V}_k = \mathbf{A}\mathbf{G}_k \mathbf{U}_k = \mathbf{G}_{k+1} \underline{\mathbf{H}}_k.$$

Thus,

$$\mathbf{x}_k := \mathbf{V}_k \mathbf{z}_k = \mathbf{G}_k \mathbf{U}_k \mathbf{z}_k, \quad \mathbf{z}_k := \underline{\mathbf{H}}_k^{\dagger} \underline{\mathbf{e}}_1 \|\mathbf{r}_0\|.$$

Simplified residual bound (block-wise orthonormalization):

$$\begin{aligned} \|\mathbf{r}_k\| = \|\mathbf{r}_0 - \mathbf{A}\mathbf{x}_k\| &\leqslant \|\mathbf{G}_{k+1}\| \cdot \left\| \underline{\mathbf{e}}_1 \|\mathbf{r}_0\| - \underline{\mathbf{H}}_k \mathbf{z}_k \right\| \\ &\leqslant \sqrt{\left\lceil \frac{k+1}{s+1} \right\rceil} \cdot \left\| \underline{\mathbf{e}}_1 \|\mathbf{r}_0\| - \underline{\mathbf{H}}_k \mathbf{z}_k \right\|. \end{aligned}$$

QMRIDR

MR methods: use extended Hessenberg matrix

$$\mathbf{x}_k := \mathbf{Q}_k \mathbf{z}_k, \quad \mathbf{z}_k := \underline{\mathbf{H}}_k^{\dagger} \underline{\mathbf{e}}_1 \|\mathbf{r}_0\|.$$

IDR based: generalized Hessenberg decomposition,

$$\mathbf{A}\mathbf{V}_k = \mathbf{A}\mathbf{G}_k \mathbf{U}_k = \mathbf{G}_{k+1} \underline{\mathbf{H}}_k.$$

Thus,

$$\mathbf{x}_k := \mathbf{V}_k \mathbf{z}_k = \mathbf{G}_k \mathbf{U}_k \mathbf{z}_k, \quad \mathbf{z}_k := \underline{\mathbf{H}}_k^{\dagger} \underline{\mathbf{e}}_1 \|\mathbf{r}_0\|.$$

Simplified residual bound (block-wise orthonormalization):

$$\begin{aligned}\|\mathbf{r}_k\| = \|\mathbf{r}_0 - \mathbf{A}\mathbf{x}_k\| &\leqslant \|\mathbf{G}_{k+1}\| \cdot \left\|\underline{\mathbf{e}}_1 \|\mathbf{r}_0\| - \underline{\mathbf{H}}_k \mathbf{z}_k\right\| \\ &\leqslant \sqrt{\left\lceil \frac{k+1}{s+1} \right\rceil} \cdot \left\|\underline{\mathbf{e}}_1 \|\mathbf{r}_0\| - \underline{\mathbf{H}}_k \mathbf{z}_k\right\|.\end{aligned}$$

Implementation based on short recurrences possible.

QMRIDR

Other Krylov-paradigms possible, e.g., flexible QMRIDR:

QMRIDR

Other Krylov-paradigms possible, e.g., flexible QMRIDR:

$$P_j(\mathbf{A})\mathbf{v}_k = (\alpha_j \mathbf{A} + \beta_j \mathbf{I})\mathbf{v}_k \;\rightsquigarrow\; (\alpha_j \mathbf{A}\mathbf{P}_k^{-1} + \beta_j \mathbf{I})\mathbf{v}_k = \mathbf{A}\widetilde{\mathbf{v}}_k + \beta_j \mathbf{v}_k,$$

$$\widetilde{\mathbf{v}}_k := \mathbf{P}_k^{-1}\mathbf{v}_k\alpha_j, \quad \mathbf{A}\widetilde{\mathbf{V}}_k = \mathbf{G}_{k+1}\underline{\mathbf{H}}_k.$$

QMRIDR

Other Krylov-paradigms possible, e.g., flexible QMRIDR:

$$P_j(\mathbf{A})\mathbf{v}_k = (\alpha_j\mathbf{A} + \beta_j\mathbf{I})\mathbf{v}_k \rightsquigarrow (\alpha_j\mathbf{A}\mathbf{P}_k^{-1} + \beta_j\mathbf{I})\mathbf{v}_k = \mathbf{A}\widetilde{\mathbf{v}}_k + \beta_j\mathbf{v}_k,$$

$$\widetilde{\mathbf{v}}_k := \mathbf{P}_k^{-1}\mathbf{v}_k\alpha_j, \quad \mathbf{A}\widetilde{\mathbf{V}}_k = \mathbf{G}_{k+1}\underline{\mathbf{H}}_k.$$

Generalized Hessenberg relation, generically no longer generalized Hessenberg decomposition, as generically

$$\mathbf{A}\widetilde{\mathbf{V}}_k \neq \mathbf{A}\mathbf{G}_k\widetilde{\mathbf{U}}_k$$

for every (upper triangular) $\widetilde{\mathbf{U}}_k$.

QMRIDR

Other Krylov-paradigms possible, e.g., flexible QMRIDR:

$$P_j(\mathbf{A})\mathbf{v}_k = (\alpha_j \mathbf{A} + \beta_j \mathbf{I})\mathbf{v}_k \rightsquigarrow (\alpha_j \mathbf{A}\mathbf{P}_k^{-1} + \beta_j \mathbf{I})\mathbf{v}_k = \mathbf{A}\widetilde{\mathbf{v}}_k + \beta_j \mathbf{v}_k,$$

$$\widetilde{\mathbf{v}}_k := \mathbf{P}_k^{-1}\mathbf{v}_k\alpha_j, \quad \mathbf{A}\widetilde{\mathbf{V}}_k = \mathbf{G}_{k+1}\underline{\mathbf{H}}_k.$$

Generalized Hessenberg relation, generically no longer generalized Hessenberg decomposition, as generically

$$\mathbf{A}\widetilde{\mathbf{V}}_k \neq \mathbf{A}\mathbf{G}_k\widetilde{\mathbf{U}}_k$$

for every (upper triangular) $\widetilde{\mathbf{U}}_k$.

Computation of flexible MR iterate and flexible MR approximation:

$$\underline{\mathbf{z}}_k := \underline{\mathbf{H}}_k^{\dagger}\underline{\mathbf{e}}_1\|\mathbf{r}_0\|, \quad \underline{\mathbf{x}}_k := \widetilde{\mathbf{V}}_k\underline{\mathbf{z}}_k.$$

QMRIDR

Other Krylov-paradigms possible, e.g., flexible QMRIDR:

$$P_j(\mathbf{A})\mathbf{v}_k = (\alpha_j \mathbf{A} + \beta_j \mathbf{I})\mathbf{v}_k \rightsquigarrow (\alpha_j \mathbf{A}\mathbf{P}_k^{-1} + \beta_j \mathbf{I})\mathbf{v}_k = \mathbf{A}\widetilde{\mathbf{v}}_k + \beta_j \mathbf{v}_k,$$
$$\widetilde{\mathbf{v}}_k := \mathbf{P}_k^{-1}\mathbf{v}_k\alpha_j, \quad \mathbf{A}\widetilde{\mathbf{V}}_k = \mathbf{G}_{k+1}\underline{\mathbf{H}}_k.$$

Generalized Hessenberg relation, generically no longer generalized Hessenberg decomposition, as generically

$$\mathbf{A}\widetilde{\mathbf{V}}_k \neq \mathbf{A}\mathbf{G}_k\widetilde{\mathbf{U}}_k$$

for every (upper triangular) $\widetilde{\mathbf{U}}_k$.

Computation of flexible MR iterate and flexible MR approximation:

$$\underline{\mathbf{z}}_k := \underline{\mathbf{H}}_k^{\dagger}\underline{\mathbf{e}}_1\|\mathbf{r}_0\|, \quad \underline{\mathbf{x}}_k := \widetilde{\mathbf{V}}_k\underline{\mathbf{z}}_k.$$

Flexible IDR variants algorithmically very easy to implement.

QMRIDR

Multi-shift is a technique developed for shifted systems

$$(\mathbf{A} - \sigma\mathbf{I})\mathbf{x}^{(\sigma)} = \mathbf{r}_0, \quad \sigma \in \mathbb{C}.$$

QMRIDR

Multi-shift is a technique developed for shifted systems

$$(\mathbf{A} - \sigma\mathbf{I})\mathbf{x}^{(\sigma)} = \mathbf{r}_0, \quad \sigma \in \mathbb{C}.$$

We look for quasi-optimal approximations of the form

$$\mathbf{x}^{(\sigma)} \approx \underline{\mathbf{x}}_k^{(\sigma)} := \mathbf{V}_k \underline{\mathbf{z}}_k^{(\sigma)}.$$

QMRIDR

Multi-shift is a technique developed for shifted systems

$$(\mathbf{A} - \sigma\mathbf{I})\mathbf{x}^{(\sigma)} = \mathbf{r}_0, \quad \sigma \in \mathbb{C}.$$

We look for quasi-optimal approximations of the form

$$\mathbf{x}^{(\sigma)} \approx \underline{\mathbf{x}}_k^{(\sigma)} := \mathbf{V}_k\underline{\mathbf{z}}_k^{(\sigma)}.$$

Since $\mathbf{A}\mathbf{V}_k = \mathbf{A}\mathbf{G}_k\mathbf{U}_k = \mathbf{G}_{k+1}\underline{\mathbf{H}}_k$, and since we use $\mathbf{G}_{k+1}\underline{\mathbf{e}}_1\|\mathbf{r}_0\| = \mathbf{r}_0$,

$$\underline{\mathbf{r}}_k^{(\sigma)} = \mathbf{r}_0 - (\mathbf{A} - \sigma\mathbf{I})\underline{\mathbf{x}}_k^{(\sigma)} = \mathbf{G}_{k+1}\left(\underline{\mathbf{e}}_1\|\mathbf{r}_0\| - (\underline{\mathbf{H}}_k - \sigma\underline{\mathbf{U}}_k)\underline{\mathbf{z}}_k^{(\sigma)}\right).$$

QMRIDR

Multi-shift is a technique developed for shifted systems

$$(\mathbf{A} - \sigma\mathbf{I})\mathbf{x}^{(\sigma)} = \mathbf{r}_0, \quad \sigma \in \mathbb{C}.$$

We look for quasi-optimal approximations of the form

$$\mathbf{x}^{(\sigma)} \approx \underline{\mathbf{x}}_k^{(\sigma)} := \mathbf{V}_k \underline{\mathbf{z}}_k^{(\sigma)}.$$

Since $\mathbf{A}\mathbf{V}_k = \mathbf{A}\mathbf{G}_k\mathbf{U}_k = \mathbf{G}_{k+1}\underline{\mathbf{H}}_k$, and since we use $\mathbf{G}_{k+1}\underline{\mathbf{e}}_1\|\mathbf{r}_0\| = \mathbf{r}_0$,

$$\underline{\mathbf{r}}_k^{(\sigma)} = \mathbf{r}_0 - (\mathbf{A} - \sigma\mathbf{I})\underline{\mathbf{x}}_k^{(\sigma)} = \mathbf{G}_{k+1}\left(\underline{\mathbf{e}}_1\|\mathbf{r}_0\| - (\underline{\mathbf{H}}_k - \sigma\underline{\mathbf{U}}_k)\underline{\mathbf{z}}_k^{(\sigma)}\right).$$

Thus, $\underline{\mathbf{z}}_k^{(\sigma)}$ quasi-optimal:

$$\underline{\mathbf{z}}_k^{(\sigma)} := (\underline{\mathbf{H}}_k - \sigma\underline{\mathbf{U}}_k)^{\dagger}\underline{\mathbf{e}}_1\|\mathbf{r}_0\|.$$

QMRIDR

Multi-shift is a technique developed for shifted systems

$$(\mathbf{A} - \sigma\mathbf{I})\mathbf{x}^{(\sigma)} = \mathbf{r}_0, \quad \sigma \in \mathbb{C}.$$

We look for quasi-optimal approximations of the form

$$\mathbf{x}^{(\sigma)} \approx \underline{\mathbf{x}}_k^{(\sigma)} := \mathbf{V}_k \underline{\mathbf{z}}_k^{(\sigma)}.$$

Since $\mathbf{A}\mathbf{V}_k = \mathbf{A}\mathbf{G}_k\mathbf{U}_k = \mathbf{G}_{k+1}\underline{\mathbf{H}}_k$, and since we use $\mathbf{G}_{k+1}\underline{\mathbf{e}}_1\|\mathbf{r}_0\| = \mathbf{r}_0$,

$$\underline{\mathbf{r}}_k^{(\sigma)} = \mathbf{r}_0 - (\mathbf{A} - \sigma\mathbf{I})\underline{\mathbf{x}}_k^{(\sigma)} = \mathbf{G}_{k+1}\left(\underline{\mathbf{e}}_1\|\mathbf{r}_0\| - (\underline{\mathbf{H}}_k - \sigma\underline{\mathbf{U}}_k)\underline{\mathbf{z}}_k^{(\sigma)}\right).$$

Thus, $\underline{\mathbf{z}}_k^{(\sigma)}$ quasi-optimal:

$$\underline{\mathbf{z}}_k^{(\sigma)} := (\underline{\mathbf{H}}_k - \sigma\underline{\mathbf{U}}_k)^{\dagger}\underline{\mathbf{e}}_1\|\mathbf{r}_0\|.$$

Various extensions for IDRSTAB: Olaf Rendel, Z. $\rightsquigarrow$ QMRIDRSTAB.

Outline

Lanczos$(s, 1) \rightsquigarrow$ the idea behind IDR(s)

Excerpt from (Sleijpen and van der Vorst, 1995, p. 204):

> *"[..], we expect to recover the convergence behavior of the incorporated Bi-CG process (in the BiCGstab methods) if we compute the iteration coefficients as accurately as possible. Therefore, we want to avoid all additional perturbations that might be introduced by an unfortunate choice of the polynomial process that is carried out on top of the Bi-CG process."*

Lanczos$(s, 1) \rightsquigarrow$ the idea behind IDR(s)

Excerpt from (Sleijpen and van der Vorst, 1995, p. 204):

> *"[..], we expect to recover the convergence behavior of the incorporated Bi-CG process (in the BiCGstab methods) if we compute the iteration coefficients as accurately as possible. Therefore, we want to avoid all additional perturbations that might be introduced by an unfortunate choice of the polynomial process that is carried out on top of the Bi-CG process."*

IDR based on Lanczos$(s, 1)$. Properties of IDR inherited from Lanczos$(s, 1)$.

Lanczos$(s, 1) \rightsquigarrow$ the idea behind IDR(s)

Excerpt from (Sleijpen and van der Vorst, 1995, p. 204):

> *"[..], we expect to recover the convergence behavior of the incorporated Bi-CG process (in the BiCGstab methods) if we compute the iteration coefficients as accurately as possible. Therefore, we want to avoid all additional perturbations that might be introduced by an unfortunate choice of the polynomial process that is carried out on top of the Bi-CG process."*

IDR based on Lanczos$(s, 1)$. Properties of IDR inherited from Lanczos$(s, 1)$.

Noted in (van Gijzen et al., 2011):

> *"[..] numerical experiments indicate that the "local closeness" of this Lanczos process to an unperturbed one is the driving force behind IDR based methods."*

Natural & good choices

Variety of approaches to chose the shadow vectors $\widetilde{\mathbf{R}}_0$:

- problem dependent,
- Recycle old information, e.g., use space spanned by previous solutions to similar problems (Newton's method; Optimization; Design Processes),
- Use (previously computed) (left) eigenvector information in IDR eigenvalue solvers,
- In PDE problems adapt shadow space to match geometrical structure (Substructuring; (Non-)Overlapping Schwarz).

Natural & good choices

Variety of approaches to chose the shadow vectors $\widetilde{\mathbf{R}}_0$:

- problem dependent,
- computer dependent,

- In general use orthonormalized basis vectors; this ensures enhanced numerical stability,
- In parallel implementations use shadow vectors adapted to the topology, i.e., non-overlapping shadow vectors,
- For better Lanczos$(s, 1)$ coefficients use higher precision,
- For faster evaluation use sparse and/or integer (e.g., with elements $0, \pm 1$) shadow vectors.

Natural & good choices

Variety of approaches to chose the shadow vectors $\widetilde{\mathbf{R}}_0$:

- problem dependent,
- computer dependent,
- independent.

If nothing is known about the matrix $\mathbf{A}$ and the computer architecture, in some sense the best choice seems to be an orthonormalized set of random vectors, cf. (Sonneveld, 2010).

This is the choice we used in our experiments.

Thinking locally or acting globally

Questions concerning the STAB-part:

- How do we choose the degrees of the polynomials?
- How do we choose the coefficients of the polynomials?

Thinking locally or acting globally

Questions concerning the STAB-part:

- How do we choose the degrees of the polynomials?
- How do we choose the coefficients of the polynomials?

Increasing the degree of the STAB-polynomials enlarges the space in which we can look for solutions.

Thinking locally or acting globally

Questions concerning the STAB-part:

- How do we choose the degrees of the polynomials?
- How do we choose the coefficients of the polynomials?

Increasing the degree of the STAB-polynomials enlarges the space in which we can look for solutions.

To avoid complex arithmetic for real nonsymmetric problems, e.g., stiff problems with large imaginary eigenvalues, and still ensure convergence, the degree should allow for complex roots, i.e., $\ell \geqslant 2$.

Thinking locally or acting globally

Questions concerning the STAB-part:

- How do we choose the degrees of the polynomials?
- How do we choose the coefficients of the polynomials?

Increasing the degree of the STAB-polynomials enlarges the space in which we can look for solutions.

To avoid complex arithmetic for real nonsymmetric problems, e.g., stiff problems with large imaginary eigenvalues, and still ensure convergence, the degree should allow for complex roots, i.e., $\ell \geqslant 2$.

Numerical experiments indicate: increasing degree $\rightsquigarrow$ better approximations to solutions of linear systems.

Thinking locally or acting globally

Questions concerning the STAB-part:

- How do we choose the degrees of the polynomials?
- How do we choose the coefficients of the polynomials?

Increasing the degree of the STAB-polynomials enlarges the space in which we can look for solutions.

To avoid complex arithmetic for real nonsymmetric problems, e.g., stiff problems with large imaginary eigenvalues, and still ensure convergence, the degree should allow for complex roots, i.e., $\ell \geqslant 2$.

Numerical experiments indicate: increasing degree $\rightsquigarrow$ better approximations to solutions of linear systems.

Unfortunately, higher degrees result in worse approximations of eigenvalues.

Thinking locally or acting globally

Questions concerning the STAB-part:

- How do we choose the degrees of the polynomials?
- How do we choose the coefficients of the polynomials?

Increasing the degree of the STAB-polynomials enlarges the space in which we can look for solutions.

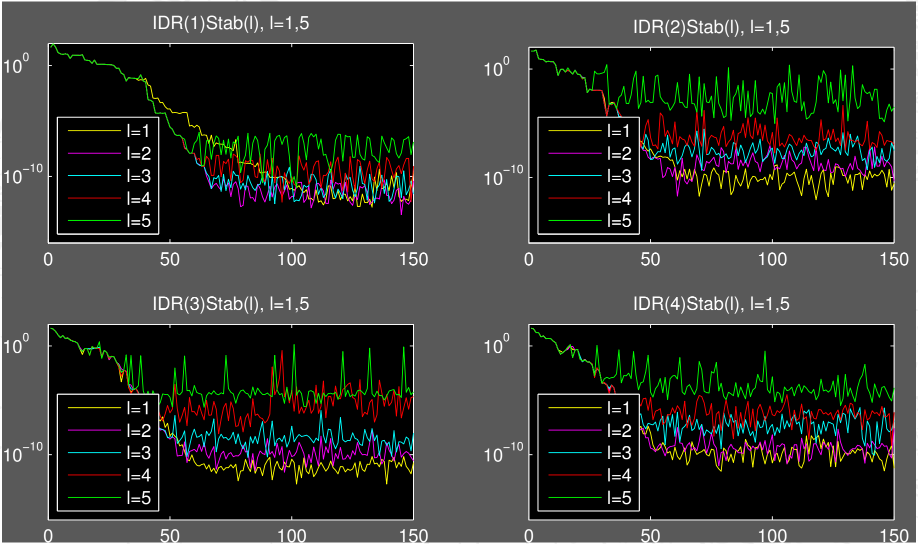
To avoid complex arithmetic for real nonsymmetric problems, e.g., stiff problems with large imaginary eigenvalues, and still ensure convergence, the degree should allow for complex roots, i.e., $\ell \geqslant 2$.

Numerical experiments indicate: increasing degree $\rightsquigarrow$ better approximations to solutions of linear systems.

Unfortunately, higher degrees result in worse approximations of eigenvalues.

We advocate to use a moderate degree ($\ell \in \{1, 2, 3, 4\}$) for eigenvalues.

Dependence of the Ritz value convergence on ℓ



Thinking locally or acting globally

Questions concerning the STAB-part:

- How do we choose the degrees of the polynomials? .
- How do we choose the coefficients of the polynomials?

Thinking locally or acting globally

Questions concerning the STAB-part:

- How do we choose the degrees of the polynomials? .
- How do we choose the coefficients of the polynomials?

In IDR linear system solvers we can minimize the norm of the residual vector over the space.

Thinking locally or acting globally

Questions concerning the STAB-part:

- How do we choose the degrees of the polynomials? .
- How do we choose the coefficients of the polynomials?

In IDR linear system solvers we can minimize the norm of the residual vector over the space.

This may slow down convergence, a cure is to ensure that the coefficients of the Lanczos$(s, 1)$ process are computed more accurately, allowing an increase in norm $\rightsquigarrow$ "vanilla variant" (Sleijpen and van der Vorst, 1995).

Thinking locally or acting globally

Questions concerning the STAB-part:

- How do we choose the degrees of the polynomials? .
- How do we choose the coefficients of the polynomials?

In IDR linear system solvers we can minimize the norm of the residual vector over the space.

This may slow down convergence, a cure is to ensure that the coefficients of the Lanczos$(s,1)$ process are computed more accurately, allowing an increase in norm $\rightsquigarrow$ "vanilla variant" (Sleijpen and van der Vorst, 1995).

Convergence depends on the interpolation of the function $z \mapsto z^{-1}$ on the spectrum using the Ritz values. We investigate various choices for the polynomial roots based on inclusion/exclusion regions for the spectrum and placement of poles.

Thinking locally or acting globally

On the next slides we use for simplicity QMRIDR(s), e.g., $\ell = 1$ and compare the following (mostly theoretical) choices:

Thinking locally or acting globally

On the next slides we use for simplicity QMRIDR(s), e.g., $\ell = 1$ and compare the following (mostly theoretical) choices:

- use some inner eigenvalues as roots, either close to the mean of the eigenvalues or to the harmonic mean,
- use the first approximations computed by (an exact) Arnoldi process,
- use the last approximations computed by (an exact) Arnoldi process.

Thinking locally or acting globally

On the next slides we use for simplicity QMRIDR(s), e.g., $\ell = 1$ and compare the following (mostly theoretical) choices:

- use some inner eigenvalues as roots, either close to the mean of the eigenvalues or to the harmonic mean,
- use the first approximations computed by (an exact) Arnoldi process,
- use the last approximations computed by (an exact) Arnoldi process.

For comparison, we include the convergence curves using the residual minimization and its “vanilla variant”.

Thinking locally or acting globally

On the next slides we use for simplicity QMRIDR(s), e.g., $\ell = 1$ and compare the following (mostly theoretical) choices:

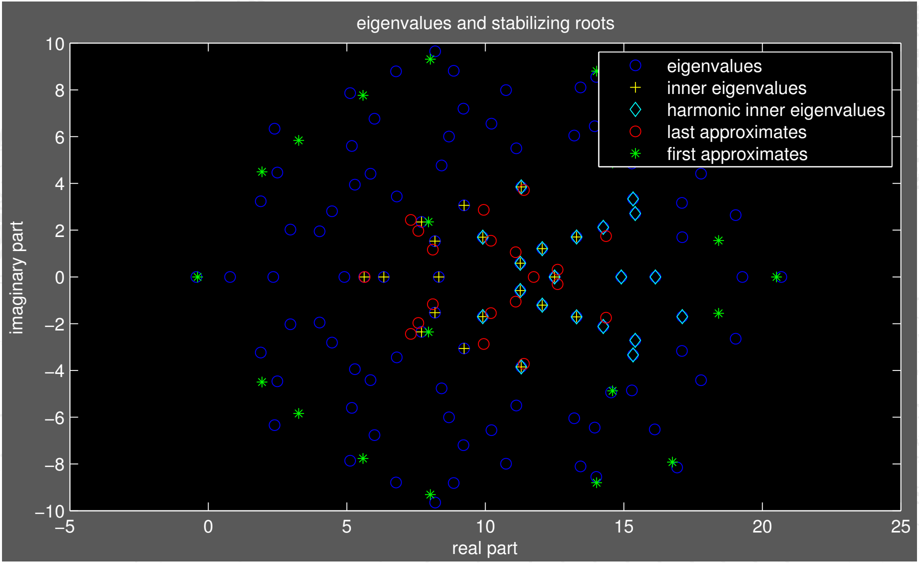
- use some inner eigenvalues as roots, either close to the mean of the eigenvalues or to the harmonic mean,
- use the first approximations computed by (an exact) Arnoldi process,
- use the last approximations computed by (an exact) Arnoldi process.

For comparison, we include the convergence curves using the residual minimization and its "vanilla variant".

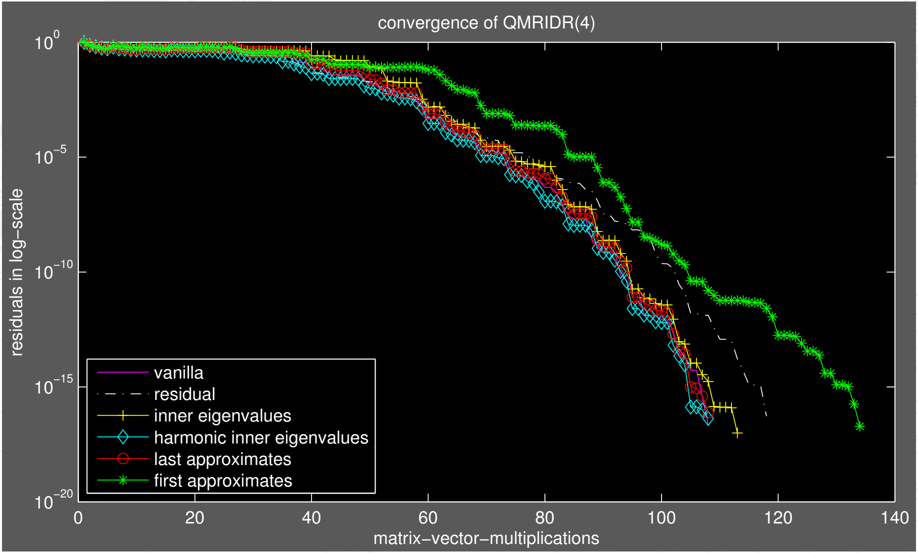
In the experiments we always used matrices $\mathbf{A} \in \mathbb{R}^{100\times 100}$:

- a shifted random matrix,
- a Grcar matrix,
- a Frank matrix,
- a randomly perturbed Poisson matrix, $\tau = \texttt{eps} = 2^{-52} \approx 2.2204 \cdot 10^{-16}$,
- a randomly perturbed Poisson matrix, $\tau = \sqrt[4]{\texttt{eps}} \approx 1.2207 \cdot 10^{-4}$.

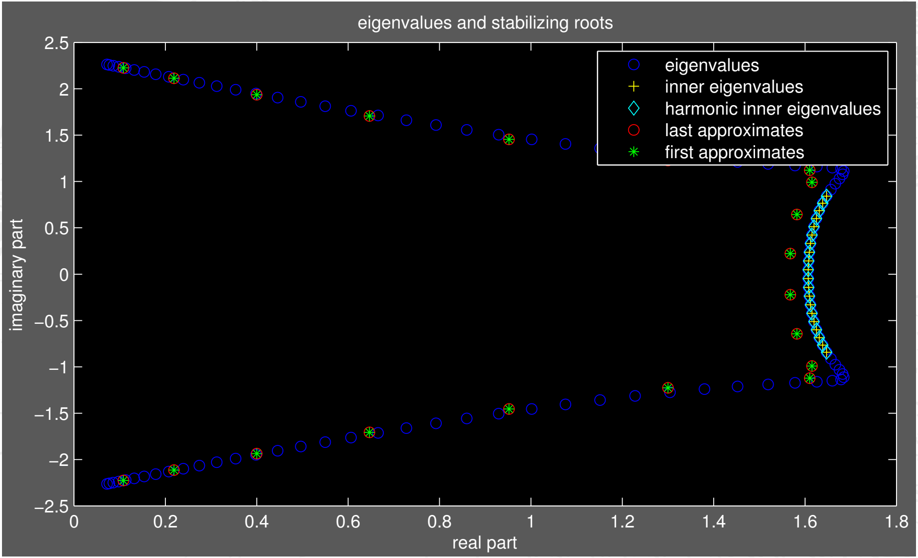
Various choices for stabilizer roots: Example 1



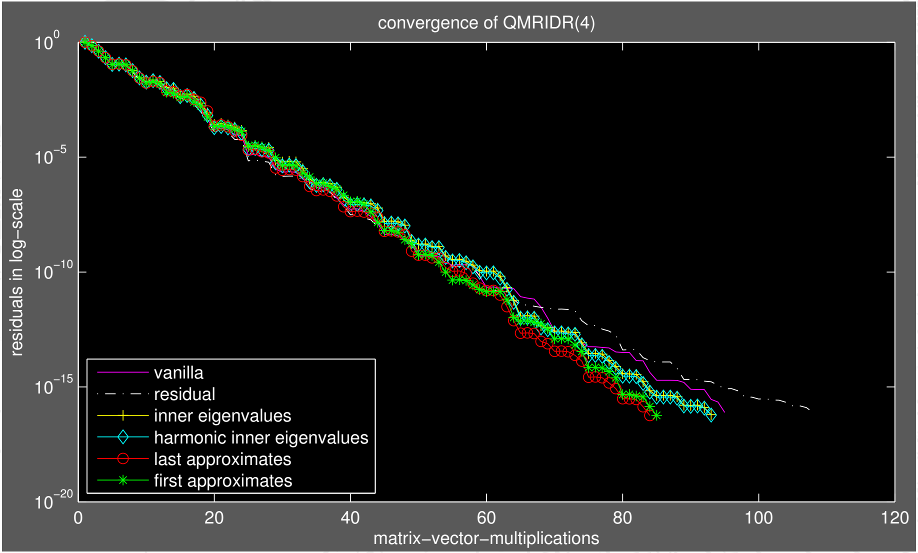
Various choices for stabilizer roots: Example 1



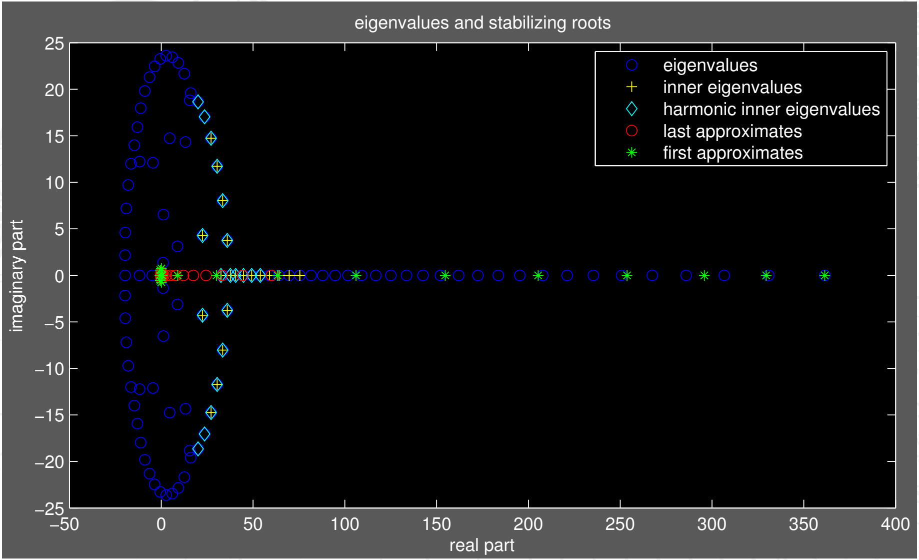
Various choices for stabilizer roots: Example 2



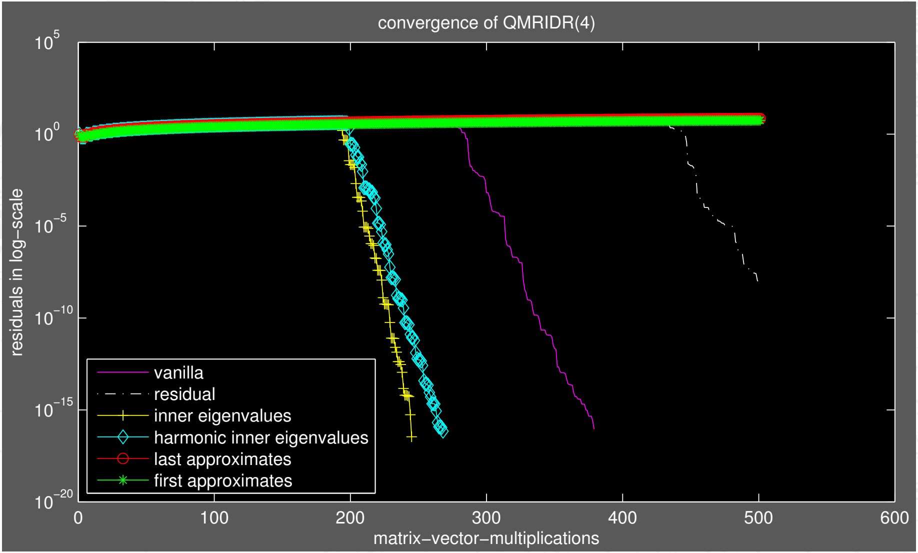
Various choices for stabilizer roots: Example 2



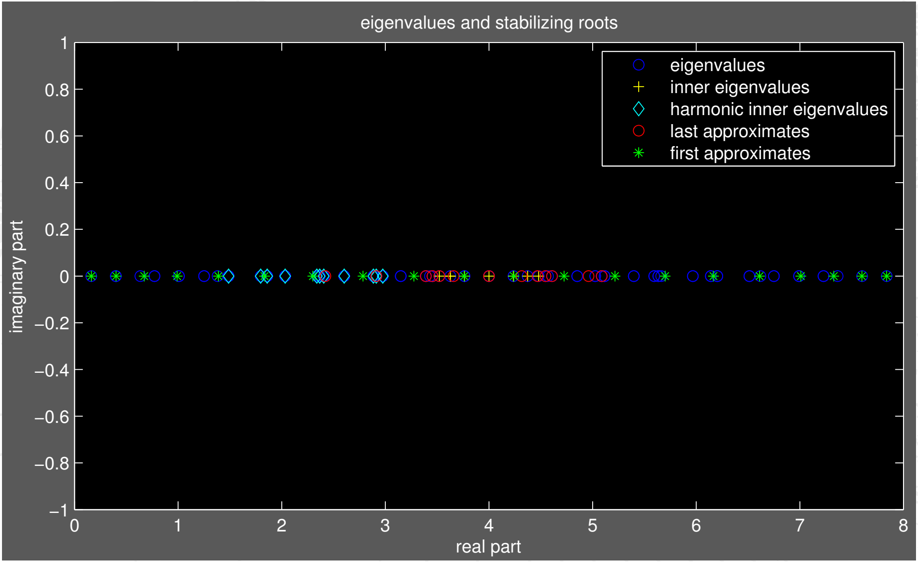
Various choices for stabilizer roots: Example 3



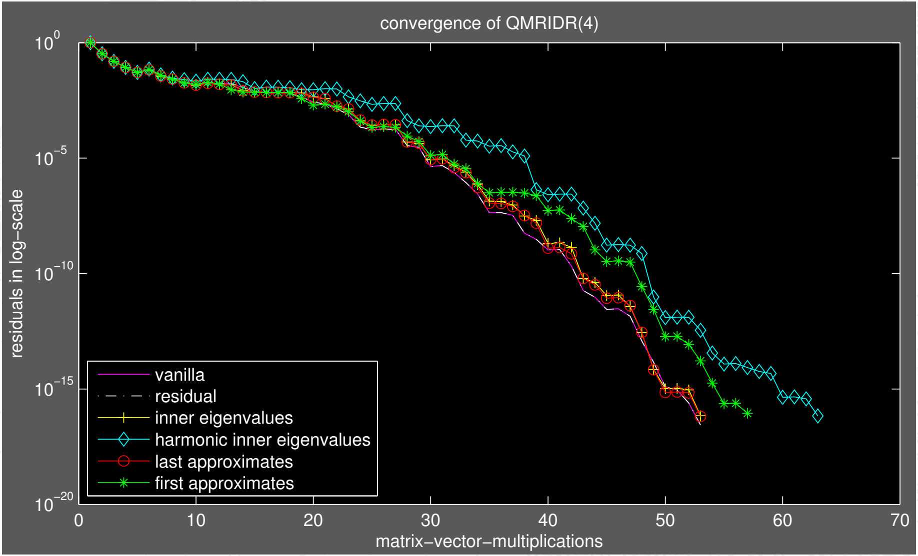
Various choices for stabilizer roots: Example 3



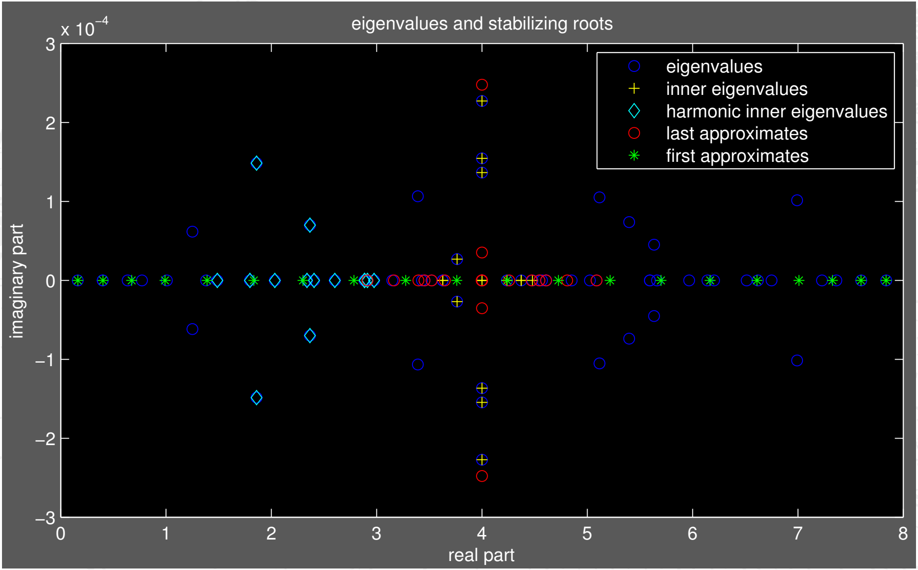
Various choices for stabilizer roots: Example 4



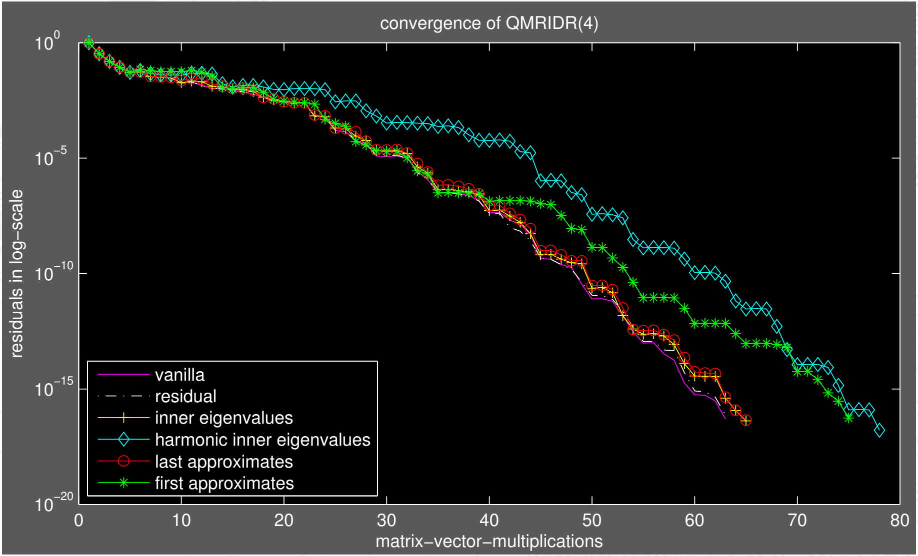
Various choices for stabilizer roots: Example 4



Various choices for stabilizer roots: Example 5



Various choices for stabilizer roots: Example 5



Optimality, cost, and stability

In (Sonneveld, 2010) a relation between IDR and GMRES for the case of random shadow vectors was pointed out.

Optimality, cost, and stability

In (Sonneveld, 2010) a relation between IDR and GMRES for the case of random shadow vectors was pointed out.

Neglecting the influence of the STAB-part, i.e., focusing on Lanczos$(s, 1)$, the deviation of IDR from GMRES is described using stochastic arguments.

Optimality, cost, and stability

In (Sonneveld, 2010) a relation between IDR and GMRES for the case of random shadow vectors was pointed out.

Neglecting the influence of the STAB-part, i.e., focusing on Lanczos$(s, 1)$, the deviation of IDR from GMRES is described using stochastic arguments.

As a rule of thumb:

> *As s tends to infinity, the convergence curves of Lanczos$(s, 1)$ tend to the convergence curve of full* GMRES*.*

Optimality, cost, and stability

In (Sonneveld, 2010) a relation between IDR and GMRES for the case of random shadow vectors was pointed out.

Neglecting the influence of the STAB-part, i.e., focusing on Lanczos$(s,1)$, the deviation of IDR from GMRES is described using stochastic arguments.

As a rule of thumb:

> *As s tends to infinity, the convergence curves of Lanczos$(s,1)$ tend to the convergence curve of full GMRES.*

In practice, the first steps of IDR/QMRIDR and Arnoldi/GMRES coincide, as we ideally start IDR with these methods.

Optimality, cost, and stability

In (Sonneveld, 2010) a relation between IDR and GMRES for the case of random shadow vectors was pointed out.

Neglecting the influence of the STAB-part, i.e., focusing on Lanczos$(s,1)$, the deviation of IDR from GMRES is described using stochastic arguments.

As a rule of thumb:

> *As s tends to infinity, the convergence curves of Lanczos$(s,1)$ tend to the convergence curve of full GMRES.*

In practice, the first steps of IDR/QMRIDR and Arnoldi/GMRES coincide, as we ideally start IDR with these methods.

We present some examples that depict the relations in (Sonneveld, 2010), show additionally the effects of finite precision, and relate GMRES to QMR$(s,1)$ and to QMRIDR(s).

Optimality, cost, and stability

In (Sonneveld, 2010) a relation between IDR and GMRES for the case of random shadow vectors was pointed out.

Neglecting the influence of the STAB-part, i.e., focusing on Lanczos$(s,1)$, the deviation of IDR from GMRES is described using stochastic arguments.

As a rule of thumb:

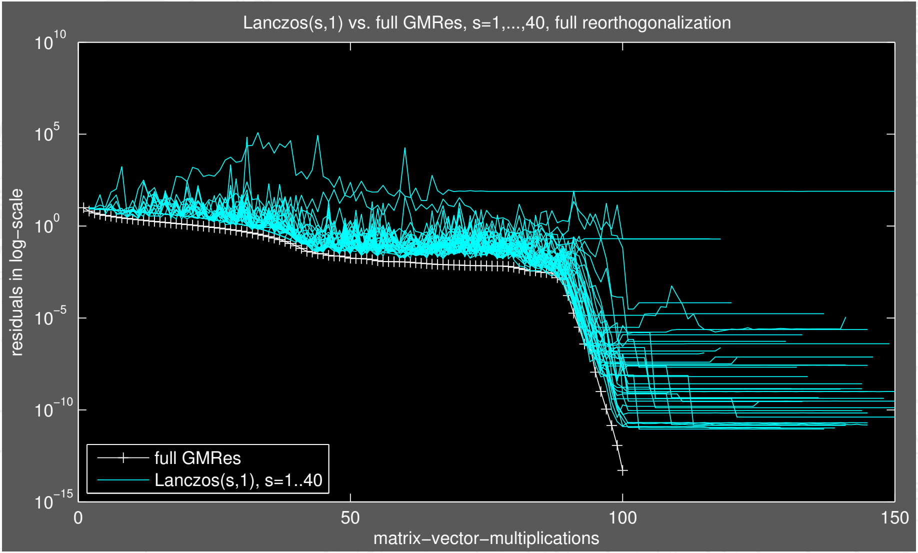
> *As s tends to infinity, the convergence curves of Lanczos$(s,1)$ tend to the convergence curve of full GMRES.*

In practice, the first steps of IDR/QMRIDR and Arnoldi/GMRES coincide, as we ideally start IDR with these methods.

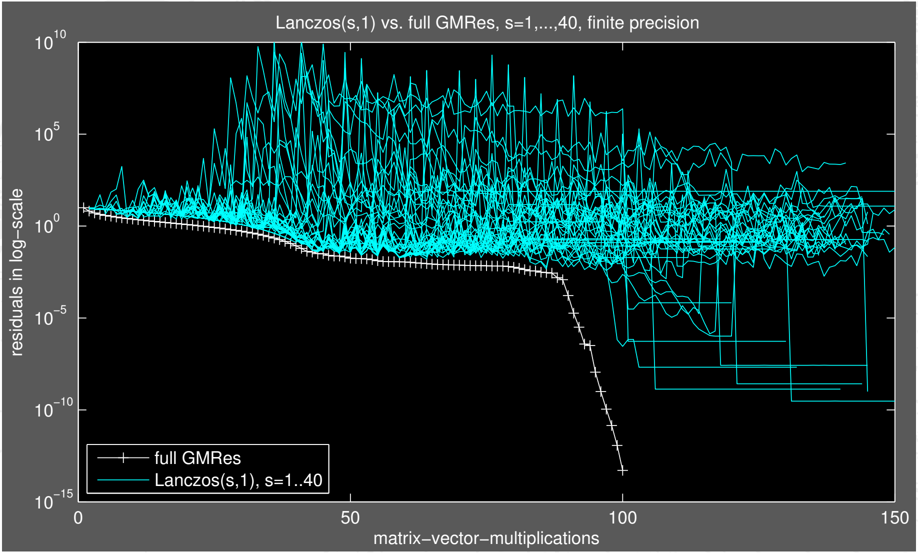
We present some examples that depict the relations in (Sonneveld, 2010), show additionally the effects of finite precision, and relate GMRES to QMR$(s,1)$ and to QMRIDR(s).

We remark that the prototype IDR algorithm suffered from instability for large values of s. We only consider new, stable implementations.

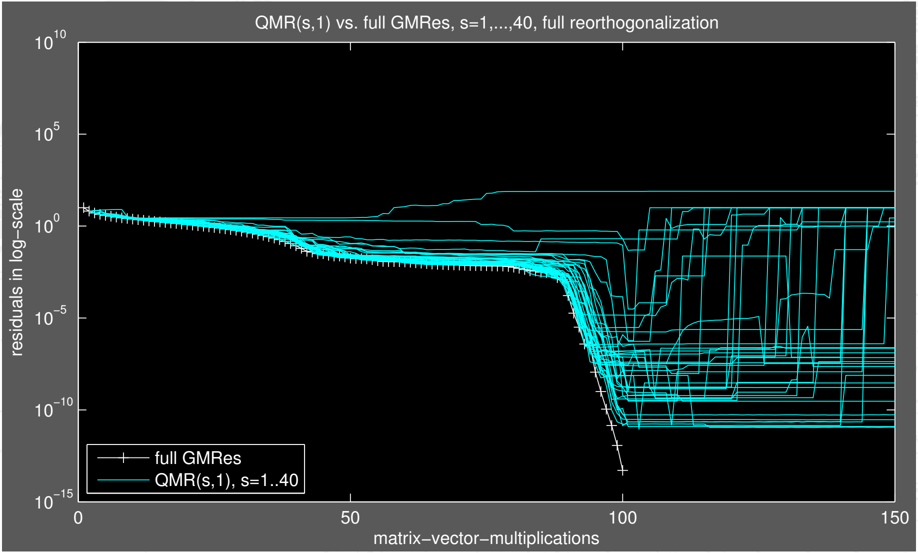
"Exact" Lanczos$(s, 1)$ versus full GMRES



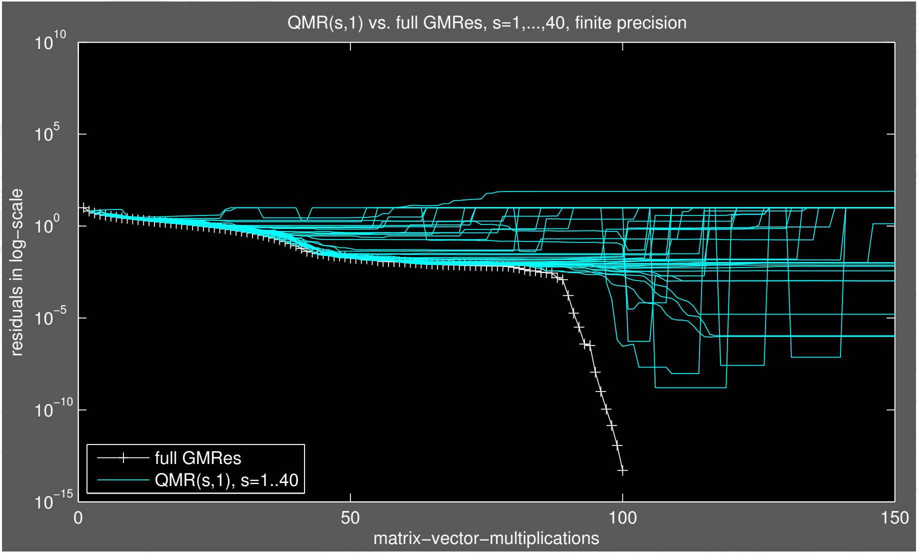
"Finite precision" Lanczos$(s, 1)$ versus full GMRES



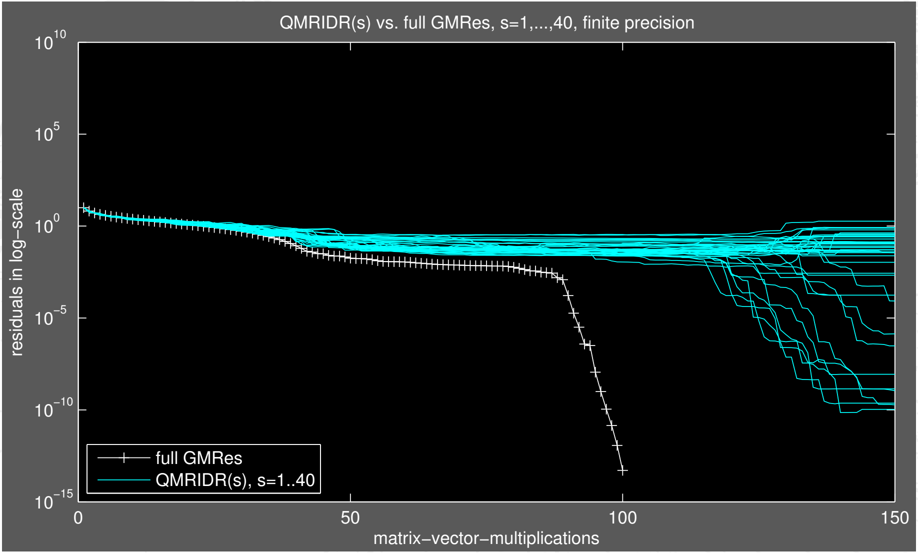
"Exact" QMR$(s, 1)$ versus full GMRES



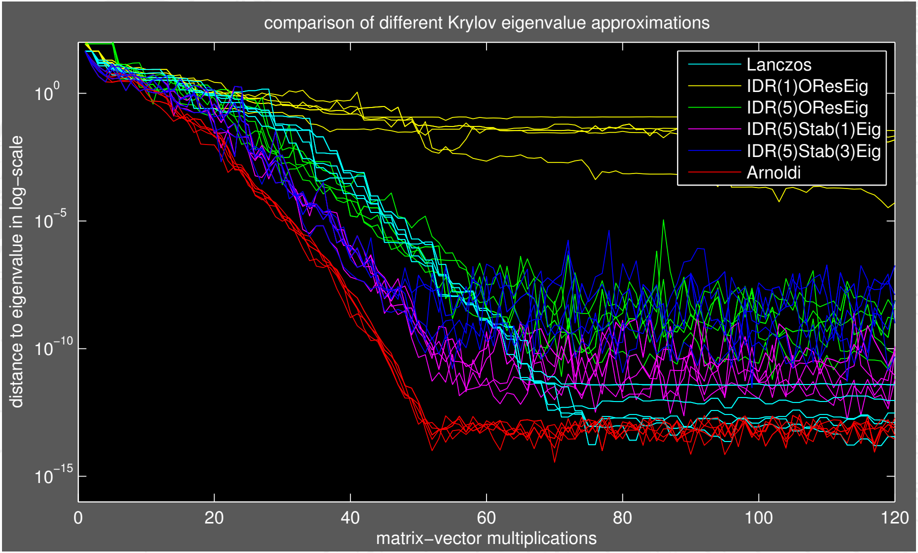
"Finite precision" QMR$(s, 1)$ versus full GMRES



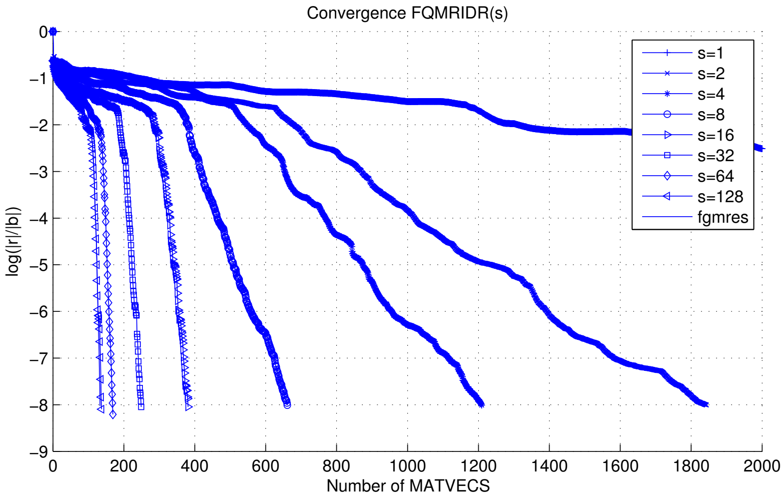
Finite precision QMRIDR(s) versus full GMRES



A comparison: IDR based eigenvalue solvers



Flexible QMRIDR(s)



Perturbations

IDR based on short recurrences, i.e., Lanczos based.

Perturbations

IDR based on short recurrences, i.e., Lanczos based.

$\rightsquigarrow$ Behavior in finite precision? Inexact methods? General perturbations?

Perturbations

IDR based on short recurrences, i.e., Lanczos based.

$\rightsquigarrow$ Behavior in finite precision? Inexact methods? General perturbations?

Lanczos	**IDR**
deviation	deviation
multiple Ritz values	ghost polynomial roots
delay of convergence	delay of convergence
attainable accuracy: condition	attainable accuracy: worse than Lanczos
analysis by Chris Paige	thus far no error analysis available

Perturbations

IDR based on short recurrences, i.e., Lanczos based.

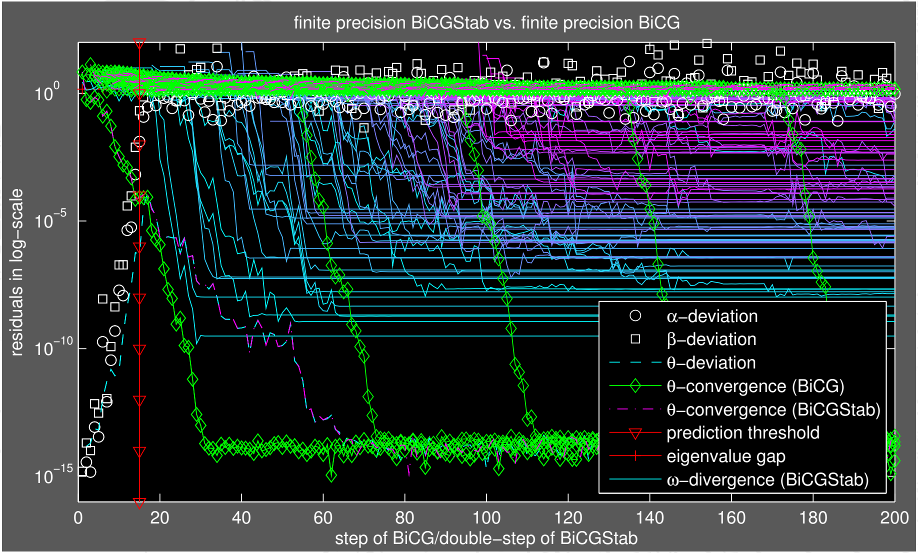
$\rightsquigarrow$ Behavior in finite precision? Inexact methods? General perturbations?

Lanczos	IDR
deviation	deviation
multiple Ritz values	ghost polynomial roots
delay of convergence	delay of convergence
attainable accuracy: condition	attainable accuracy: worse than Lanczos
analysis by Chris Paige	thus far no error analysis available

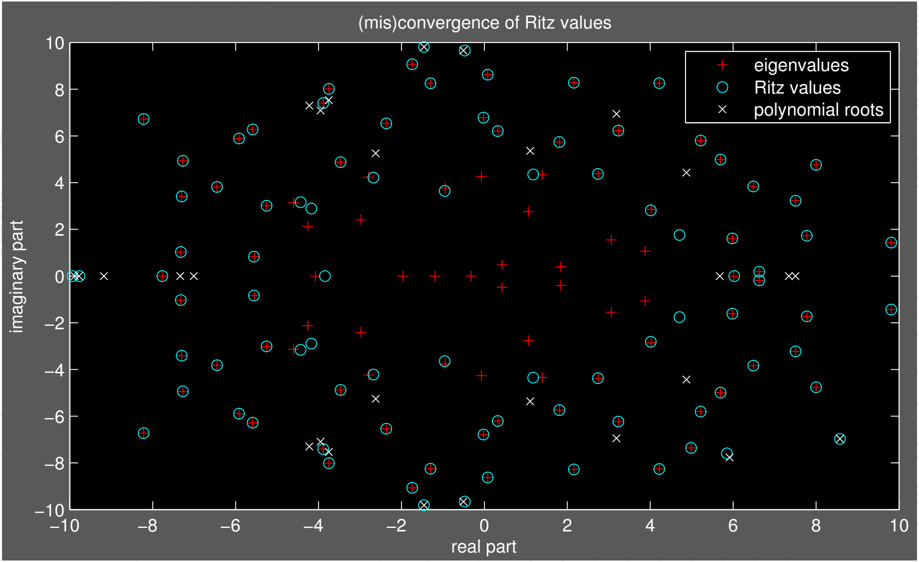
But:

- IDR transpose-free,
- easy to implement,
- more stable (for large values of s),
- often close to “optimal” methods (for large values of s).

BiCGStab vs. BiCG



IDR(3)STAB(3): “Ghost polynomial roots”



Conclusion and Outview

- The new implementations of IDR, i.e., IDRSTAB, QMRIDR, its combinations, and the eigensolver counterparts, are very promising.

Conclusion and Outview

- The new implementations of IDR, i.e., IDRSTAB, QMRIDR, its combinations, and the eigensolver counterparts, are very promising.
- IDR based methods offer a variety of parameters. We presented some ideas and experiments to sketch recent progress.

Conclusion and Outview

- The new implementations of IDR, i.e., IDRSTAB, QMRIDR, its combinations, and the eigensolver counterparts, are very promising.
- IDR based methods offer a variety of parameters. We presented some ideas and experiments to sketch recent progress.
- As a rule of thumb: If nothing about the problem is known in advance,

Conclusion and Outview

- The new implementations of IDR, i.e., IDRSTAB, QMRIDR, its combinations, and the eigensolver counterparts, are very promising.
- IDR based methods offer a variety of parameters. We presented some ideas and experiments to sketch recent progress.
- As a rule of thumb: If nothing about the problem is known in advance,
  - chose s as large as possible,

Conclusion and Outview

- The new implementations of IDR, i.e., IDRSTAB, QMRIDR, its combinations, and the eigensolver counterparts, are very promising.
- IDR based methods offer a variety of parameters. We presented some ideas and experiments to sketch recent progress.
- As a rule of thumb: If nothing about the problem is known in advance,
  - chose s as large as possible,
  - chose a polynomial with moderate degree,

Conclusion and Outview

- The new implementations of IDR, i.e., IDRSTAB, QMRIDR, its combinations, and the eigensolver counterparts, are very promising.
- IDR based methods offer a variety of parameters. We presented some ideas and experiments to sketch recent progress.
- As a rule of thumb: If nothing about the problem is known in advance,
  - chose s as large as possible,
  - chose a polynomial with moderate degree,
  - chose the coefficients using the "vanilla" strategy,

Conclusion and Outview

- The new implementations of IDR, i.e., IDRSTAB, QMRIDR, its combinations, and the eigensolver counterparts, are very promising.
- IDR based methods offer a variety of parameters. We presented some ideas and experiments to sketch recent progress.
- As a rule of thumb: If nothing about the problem is known in advance,
  - chose s as large as possible,
  - chose a polynomial with moderate degree,
  - chose the coefficients using the "vanilla" strategy,
  - use random starting vectors,

Conclusion and Outview

- The new implementations of IDR, i.e., IDRSTAB, QMRIDR, its combinations, and the eigensolver counterparts, are very promising.
- IDR based methods offer a variety of parameters. We presented some ideas and experiments to sketch recent progress.
- As a rule of thumb: If nothing about the problem is known in advance,
  - chose s as large as possible,
  - chose a polynomial with moderate degree,
  - chose the coefficients using the “vanilla” strategy,
  - use random starting vectors,
  - use some QMR variant.

Conclusion and Outview

- The new implementations of IDR, i.e., IDRSTAB, QMRIDR, its combinations, and the eigensolver counterparts, are very promising.
- IDR based methods offer a variety of parameters. We presented some ideas and experiments to sketch recent progress.
- As a rule of thumb: If nothing about the problem is known in advance,
  - chose s as large as possible,
  - chose a polynomial with moderate degree,
  - chose the coefficients using the “vanilla” strategy,
  - use random starting vectors,
  - use some QMR variant.
- Knowledge should be used carefully in the parameter selection process, but accelerating the convergence should definitely be tried.

Conclusion and Outview

- The new implementations of IDR, i.e., IDRSTAB, QMRIDR, its combinations, and the eigensolver counterparts, are very promising.
- IDR based methods offer a variety of parameters. We presented some ideas and experiments to sketch recent progress.
- As a rule of thumb: If nothing about the problem is known in advance,
  - chose s as large as possible,
  - chose a polynomial with moderate degree,
  - chose the coefficients using the “vanilla” strategy,
  - use random starting vectors,
  - use some QMR variant.
- Knowledge should be used carefully in the parameter selection process, but accelerating the convergence should definitely be tried.
- An error analysis and a description of the finite precision behavior is desperately needed.

Conclusion and Outview

- The new implementations of IDR, i.e., IDRSTAB, QMRIDR, its combinations, and the eigensolver counterparts, are very promising.
- IDR based methods offer a variety of parameters. We presented some ideas and experiments to sketch recent progress.
- As a rule of thumb: If nothing about the problem is known in advance,
  - chose s as large as possible,
  - chose a polynomial with moderate degree,
  - chose the coefficients using the “vanilla” strategy,
  - use random starting vectors,
  - use some QMR variant.
- Knowledge should be used carefully in the parameter selection process, but accelerating the convergence should definitely be tried.
- An error analysis and a description of the finite precision behavior is desperately needed.
- The next logical step, the development of IDR algorithms that allow to change the old stabilizing polynomials on the fly, cures some of the peculiarities current implementations suffer from.

どうも有難う御座いました。

Thank you very much for inviting me to 同志社大学.

This talk is partially based on the following technical reports:

Eigenvalue computations based on IDR, Martin H. Gutknecht and Z., Bericht 145, Institut für Numerische Simulation, TUHH, 2010,

Flexible and multi-shift induced dimension reduction algorithms for solving large sparse linear systems, Martin B. van Gijzen, Gerard L.G. Sleijpen, and Z., Bericht 156, Institut für Numerische Simulation, TUHH, 2011.

Additional material can be found in the proceedings:

Tuning IDR to fit your applications, Olaf Rendel and Z., 2011.

Sleijpen, G. L. and van der Vorst, H. A. (1995).
Maintaining convergence properties of BiCGstab methods in finite precision arithmetic.
Numer. Algorithms, 10(3-4):203–223.

Sonneveld, P. (2010).
On the convergence behaviour of IDR(s).
Technical Report 10-08, Department of Applied Mathematical Analysis, Delft University of Technology, Delft.

van Gijzen, M. B., Sleijpen, G. L., and Zemke, J.-P. M. (2011).
Flexible and multi-shift induced dimension reduction algorithms for solving large sparse linear systems.
Bericht 156, TUHH, Institute of Numerical Simulation.
Online available at
`http://doku.b.tu-harburg.de/volltexte/2011/1114/`.