GAMM Anual Meeting
Novi Sad, March 20th 2013

Recent advances in low rank ADI for Lyapunov equations

Jens Saak
joint work with
Peter Benner, Patrick Kürschner

Computational Methods in Systems and Control Theory (CSC) Max Planck Institute for Dynamics of Complex Technical Systems

MAX PLANCK INSTITUTE FOR DYNAMICS OF COMPLEX TECHNICAL SYSTEMS MAGDEBURG

Outline

1. Low Rank ADI Basics
2. Low Rank Factors of Residuals
3. Adaptive ADI Shift Computation
4. Numerical Experiments

Low Rank ADI Basics

Low Rank ADI Basics

Low Rank Solution of Lyapunov Equations

Lyapunov equation

Consider

$$FX + XF^T = -GG^T$$

with $F \in \mathbb{R}^{n \times n}$ Hurwitz, $G \in \mathbb{R}^{n \times m}$, $m \ll n$.

Goal: $$X = ZZ^H$$

Low Rank ADI Basics

Low Rank Solution of Lyapunov Equations

Lyapunov equation

Consider

$$FX + XF^T = -GG^T$$

with $F \in \mathbb{R}^{n\times n}$ Hurwitz, $G \in \mathbb{R}^{n\times m}$, $m \ll n$.

Goal: $$X = ZZ^H$$

Low-rank factor ADI iteration (LR-ADI)

[Penzl '99, Benner/Li/Penzl '08]

$$V_1 = (F + p_1 I)^{-1} G, \qquad V_j = \left[I - (p_j + \overline{p_{j-1}})(F + p_j I)^{-1}\right] V_{j-1},$$

$$Z_1 := \sqrt{-2\,\mathrm{Re}\,(p_1)}\, V_1, \qquad Z_j := [Z_{j-1},\ \sqrt{-2\,\mathrm{Re}\,(p_j)}\, V_j].$$

Low Rank ADI Basics

Shift Parameters

Optimal shift parameters

For J ADI iterations, the optimal shift parameters solve the rational minmax problem

$$\min_{p_1,\ldots,p_J \subset \mathbb{C}_-} \left(\max_{1 \leq \ell \leq n} \left| \prod_{j=1}^{J} \frac{\overline{p_j} - \lambda_\ell}{p_j + \lambda_\ell} \right| \right), \qquad \lambda_\ell \in \Lambda(F).$$

(sub-)optimal shifts [Wachspress et al ~'95]

Optimal number J of shifts and their location computable for a given error tolerance for real spectra. $\rightsquigarrow$ suboptimal for $\Lambda(F) \subset \mathbb{C}$.

Heuristic *Penzl* shifts [Penzl '99]

Since λ_ℓ not easily available in large-scale setting, take small numbers of Ritz values of F and F^{-1} (generated with Arnoldi) instead.

Low Rank ADI Basics

Stopping Criteria

Relative Change of Z:

$$\frac{\left|\left|\sqrt{-2\,\mathrm{Re}\,(p_j)}V_j\right|\right|_F}{\left|\left|Z_j\right|\right|_F} \leq tol$$

Low Rank ADI Basics

Stopping Criteria

Relative Change of Z:

$$\frac{\left|\left|\sqrt{-2\,\mathrm{Re}\,(p_j)}V_j\right|\right|_F}{||Z_j||_F} \leq tol$$

$$\mathcal{L}\,(Y) := FY + YF^T + GG^T$$

Residual Based Stopping

- $\left|\left|\mathcal{L}(ZZ^T)\right|\right|_F$ computed via clever QR updates [Penzl'99]
- $\left|\left|\mathcal{L}(ZZ^T)\right|\right|_2 = \lambda_{\max}(\mathcal{L}(ZZ^T))$ apply Arnoldi/Lanczos [S. et al '09-]

Low Rank Factors of Residuals

Low Rank Factors of Residuals

Equivalence of Lyapunov and Stein Equations

e.g.[Hylla '10]

$$FX + XF^T = -GG^T \quad \overset{p\in\mathbb{C}_{<0}}{\Longleftrightarrow} \quad X = F_p X F_p^H - 2\,\mathrm{Re}\,(p) G_p G_p^H$$

Here: $F_p := (F - \overline{p}I)(F + pI)^{-1}$ and $G_p := (F + pI)^{-1} G$

Low Rank Factors of Residuals

Equivalence of Lyapunov and Stein Equations e.g.[Hylla '10]

$$FX + XF^T = -GG^T \quad \overset{p\in\mathbb{C}_{<0}}{\Longleftrightarrow} \quad X = F_p X F_p^H - 2\,\mathrm{Re}(p) G_p G_p^H$$

Here: $F_p := (F - \overline{p}I)(F + pI)^{-1}$ and $G_p := (F + pI)^{-1} G$

And for the ADI iterates : $X_j = F_{p_j} X_{j-1} F_{p_j}^H - 2\,\mathrm{Re}(p_j) G_{p_j} G_{p_j}^H$

Low Rank Factors of Residuals

Equivalence of Lyapunov and Stein Equations

e.g.[HYLLA '10]

$$FX + XF^T = -GG^T \quad \overset{p\in\mathbb{C}_{<0}}{\Longleftrightarrow} \quad X = F_p X F_p^H - 2\,\mathrm{Re}\,(p) G_p G_p^H$$

Here: $F_p := (F - \overline{p}I)(F + pI)^{-1}$ and $G_p := (F + pI)^{-1} G$

And for the ADI iterates : $X_j = F_{p_j} X_{j-1} F_{p_j}^H - 2\,\mathrm{Re}\,(p_j) G_{p_j} G_{p_j}^H$

Then:

$$\begin{aligned}
\mathcal{L}(X_j) &= FX_j + X_j F^T + GG^T \\
&= F\,(X_j - X)+(X_j - X)\,F^T \\
&= \cdots = \left(\prod_{i=1}^{j} F_{p_i}\right) GG^T \left(\prod_{i=1}^{j} F_{p_i}\right)^H \\
&= W_j W_j^H.
\end{aligned}$$

Low Rank Factors of Residuals

Rank and Computability of the Residual Factor

[Benner/Kürschner/S.'13]

$$\mathcal{L}(X_j) = W_j W_j^H \quad \text{for} \quad W_j = \left(\prod_{i=1}^{j} F_{p_i} \right) G$$

Low Rank Factors of Residuals

Rank and Computability of the Residual Factor

[Benner/Kürschner/S.'13]

$$\mathcal{L}(X_j) = W_j W_j^H \quad \text{for} \quad W_j = \left(\prod_{i=1}^{j} F_{p_i} \right) G$$

$$\begin{aligned}
V_j &= \left[I - (\overline{p_{j-1}} + p_j)(F + p_j I)^{-1} \right] V_{j-1} \\
&= (F - \overline{p_{j-1}} I)(F + p_j I)^{-1} V_{j-1} \\
&= (F - \overline{p_{j-2}} I)\, (F + p_{j-1} I)^{-1} (F - \overline{p_{j-1}} I)\, (F + p_j I)^{-1} V_{j-2} \\
&= (F - \overline{p_{j-2}} I)\, F_{p_{j-1}}\, (F + p_j I)^{-1} V_{j-2} \\
&= \cdots = (F + p_j I)^{-1} \left(\prod_{i=1}^{j-1} F_{p_i} \right) G.
\end{aligned}$$

Low Rank Factors of Residuals

Rank and Computability of the Residual Factor

[Benner/Kürschner/S.'13]

$$\mathcal{L}(X_j) = W_j W_j^H \quad \text{for} \quad W_j = \left(\prod_{i=1}^{j} F_{p_i}\right) G$$

$$V_j = \left[I - (\overline{p_{j-1}} + p_j)\,(F + p_j I)^{-1}\right] V_{j-1}$$
$$= \cdots = (F + p_j I)^{-1} \left(\prod_{i=1}^{j-1} F_{p_i}\right) G.$$

$$W_j = \left(\prod_{i=1}^{j} F_{p_i}\right) G = (F - \overline{p_j} I)\,(F + p_j I)^{-1} \left(\prod_{i=1}^{j-1} F_{p_i}\right) G$$

Low Rank Factors of Residuals

Rank and Computability of the Residual Factor

[Benner/Kürschner/S.'13]

$$\mathcal{L}(X_j) = W_j W_j^H \quad \text{for} \quad W_j = \left(\prod_{i=1}^{j} F_{p_i} \right) G$$

$$\begin{aligned} V_j &= \left[I - \left(\overline{p_{j-1}} + p_j \right) (F + p_j I)^{-1} \right] V_{j-1} \\ &= \cdots = (F + p_j I)^{-1} \left(\prod_{i=1}^{j-1} F_{p_i} \right) G. \end{aligned}$$

$$W_j = (F - \overline{p_j} I) V_j$$

Low Rank Factors of Residuals

Rank and Computability of the Residual Factor

[Benner/Kürschner/S.'13]

$$W_j = (F - \overline{p_j} I)\, V_j$$

Moreover:

$$\begin{aligned} V_j &= (F + p_j I)^{-1} (F - \overline{p_{j-1}} I)\, V_{j-1} \\ &= (F + p_j I)^{-1}\, W_{j-1} \end{aligned}$$

Low Rank Factors of Residuals

Rank and Computability of the Residual Factor

[Benner/Kürschner/S.'13]

$$W_j = (F - \overline{p_j} I)\, V_j$$

Moreover:

$$\begin{aligned} V_j &= (F + p_j I)^{-1} (F - \overline{p_{j-1}} I)\, V_{j-1} \\ &= (F + p_j I)^{-1}\, W_{j-1} \end{aligned}$$

$$\begin{aligned} W_j &= (F - \overline{p_j} I)\, V_j = (F - \overline{p_j} I)(F + p_j I)^{-1}\, W_{j-1} \\ &= \left(I - (p_j + \overline{p_j})(F + p_j I)^{-1}\right) W_{j-1} \\ &= W_{j-1} - 2\,\mathrm{Re}\,(p_j) V_j \end{aligned}$$

The latter result was independently derived from a Krylov subspace recurrence perspective in

[Wolf/Panzer/Lohmann '13]

Low Rank Factors of Residuals

A New LR-ADI Formulation [Benner/Kürschner/S.'13]

Algorithm 1 LR-ADI Re-Formulation

Input: F,G in $FX + XF^T = -GG^T$, and proper shifts p_j, $(j = 1, \dots, J)$

Output: Z with $X \approx ZZ^H$

1: $Z = []$
2: $W_0 = G$
3: $\ell = 1$
4: **while** $(\left|\left|W_{\ell-1}^H W_{\ell-1}\right|\right| > tol$ **and** $\ell \leq J)$ **do**
5: $\quad V_\ell = (F + p_\ell I)^{-1} W_{\ell-1}$
6: $\quad W_\ell = W_{\ell-1} - 2\,\mathrm{Re}\,(p_\ell) V_\ell$
7: $\quad Z = \left[Z,\ \sqrt{-2\,\mathrm{Re}\,(p_\ell)} V_\ell\right]$
8: $\quad \ell++$
9: **end while**

Low Rank Factors of Residuals

A New LR-ADI Formulation [Benner/Kürschner/S.'13]

Algorithm 1 LR-ADI Re-Formulation

Input: F, G in $FX + XF^T = -GG^T$, and proper shifts p_j, $(j = 1, \ldots, J)$
Output: Z with $X \approx ZZ^H$

1: $Z = []$
2: $W_0 = G$
3: $\ell = 1$
4: **while** $(\left\|W_{\ell-1}^H W_{\ell-1}\right\| > tol$ **and** $\ell \leq J)$ **do**
5: $\quad V_\ell = (F + p_\ell I)^{-1} W_{\ell-1}$
6: $\quad W_\ell = W_{\ell-1} - 2\,\mathrm{Re}\,(p_\ell) V_\ell$
7: $\quad Z = \left[Z,\ \sqrt{-2\,\mathrm{Re}\,(p_\ell)} V_\ell\right]$
8: $\quad \ell{+}{+}$
9: **end while**

Note:
$\left\|W_{\ell-1} W_{\ell-1}^H\right\|_{2/F} = \left\|W_{\ell-1}^H W_{\ell-1}\right\|_{2/F}$
and $W_{\ell-1}^H W_{\ell-1}$ inner product
$\rightsquigarrow m \times m \equiv$ small

Adaptive ADI Shift Computation

Adaptive ADI Shift Computation

Basic Idea

Optimal shift parameters

For J ADI iterations, the optimal shift parameters solve the rational minmax problem

$$\min_{p_1,\ldots,p_J \subset \mathbb{C}_-} \left(\max_{1 \leq \ell \leq n} \left| \prod_{j=1}^{J} \frac{\overline{p_j} - \lambda_\ell}{p_j + \lambda_\ell} \right| \right), \qquad \lambda_\ell \in \Lambda(F).$$

- If $\Lambda(F)$ is complex: Good shifts require expert knowledge.
- The min-max-problem is solved with respect to $\Lambda(F) \rightsquigarrow \mathbb{R}^n$.
- Should we not restrict to $\mathrm{colspan}\,(Z)$? How?

Adaptive ADI Shift Computation

The New Shift Strategy

[Hund '12]

Additional assumption: $F + F^T < 0$

Then: $\Lambda(U^T F U) \subset \mathbb{C}_{<0}$ for any $U \in \mathbb{R}^{n \times k}$ orthogonal and $k \leq n$.

Bendixsons Theorem

Idea: LR-ADI converges for all proper shifts in $\mathbb{C}_{<0}$.

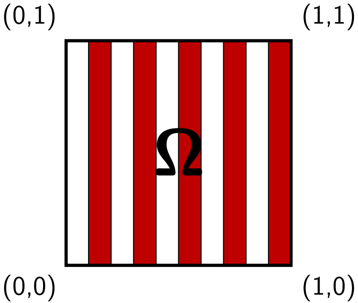
Adaptive Shift Strategy

1. $U = \mathtt{orth}(G)$
2. $p_i = \lambda_i(U^T F U)$ for all $i = 1, \ldots, m$
3. after m LR-ADI steps:
4. $U = \mathtt{orth}(V_m)$ goto 2

Numerical Experiments

Numerical Experiments

FDM discretized convection diffusion equation $n = 10\,000$, $m = 5$ [HUND '12]



- Dirichlet boundary conditions on $\Gamma = \partial\Omega$.
- Centered finite differences,
- 100 equidistant nodes in each direction.
- 5 input parameters (1 per red stripe)

$$\dot{x}(t) = -\Delta x(t) + v.\nabla x(t) + f(\xi, u(t))$$

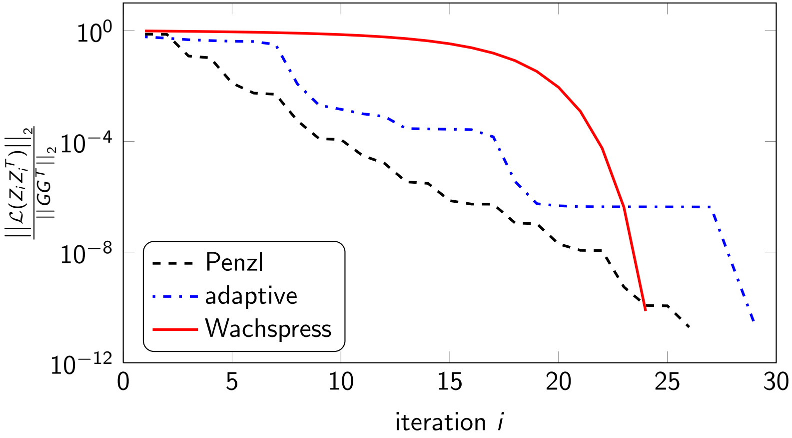
ADI Settings:

- normalized residual tolerance 10^{-10},
- 15 Penzl shifts (from 25 large/small Ritz values)
- Wachspress (10 Arnoldi steps for F/F^{-1})

Numerical Experiments

FDM discretized convection diffusion equation $n = 10\,000$, $m = 5$

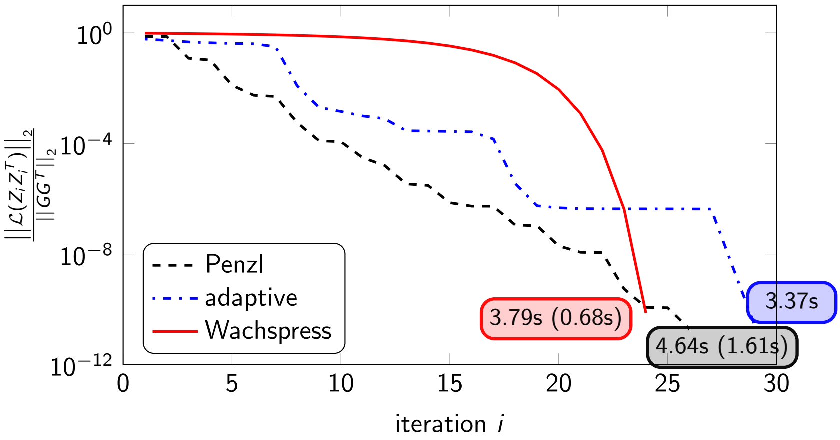
$$\dot{x}(t) = -\Delta x(t) + f(u(t))$$



Numerical Experiments

FDM discretized convection diffusion equation $n = 10\,000$, $m = 5$

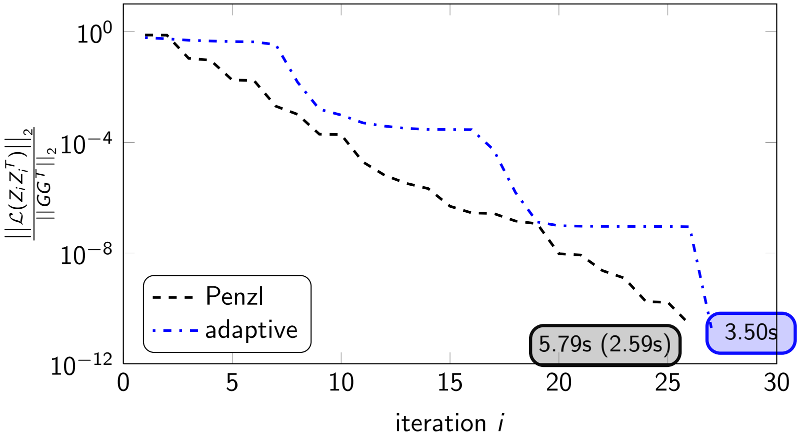
$$\dot{x}(t) = -\Delta x(t) + f(u(t))$$



Numerical Experiments

FDM discretized convection diffusion equation $n = 10\,000$, $m = 5$

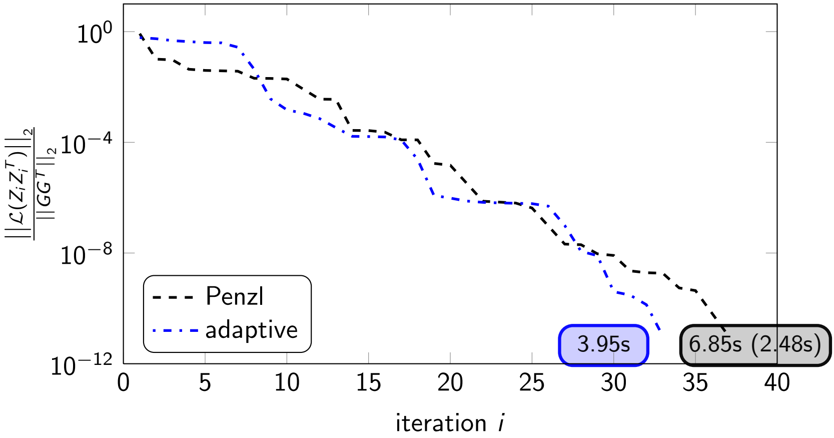
$$\dot{x}(t) = -\Delta x(t) + 10\xi_2\, \partial_{\xi_2} x(t) + f(u(t))$$



Numerical Experiments

FDM discretized convection diffusion equation $n = 10\,000$, $m = 5$

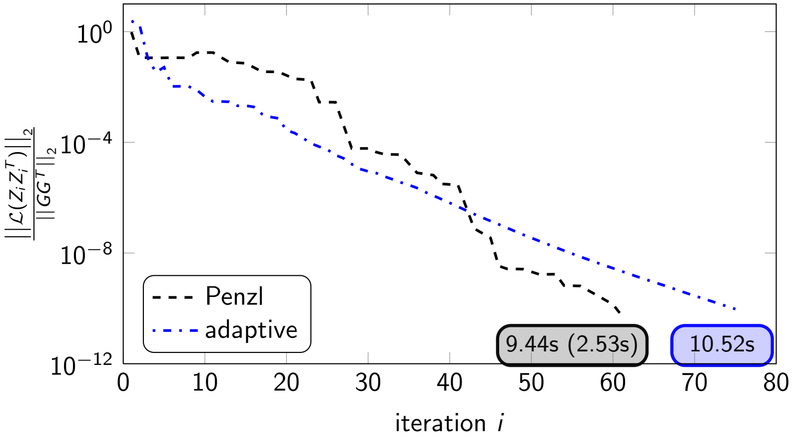
$$\dot{x}(t) = -\Delta x(t) + 100\xi_2\, \partial_{\xi_2} x(t) + f(u(t))$$



Numerical Experiments

FDM discretized convection diffusion equation $n = 10\,000$, $m = 5$

$$\dot{x}(t) = -\Delta x(t) + 1000\xi_2\,\partial_{\xi_2} x(t) + f(u(t))$$



Numerical Experiments

Index-1 Power Systems Models

e.g.[FREITAS/ROMMES/MARTINS '08]

Now: Index-1 differential algebraic systems ⇝

$$FXE^T + EXF^T = -GG^T,$$

where:

$$E = \begin{bmatrix} I_s & 0 \\ 0 & 0 \end{bmatrix}, \qquad F = \begin{bmatrix} F_{11} & F_{12} \\ F_{21} & F_{22} \end{bmatrix}, \qquad G = \begin{bmatrix} G_1 \\ G_2 \end{bmatrix}$$

Idea: Implicitly Apply LR-ADI to the Schur complement form, i.e.,

$$\tilde{F}X + X\tilde{F}^T = -\tilde{G}\tilde{G}^T,$$

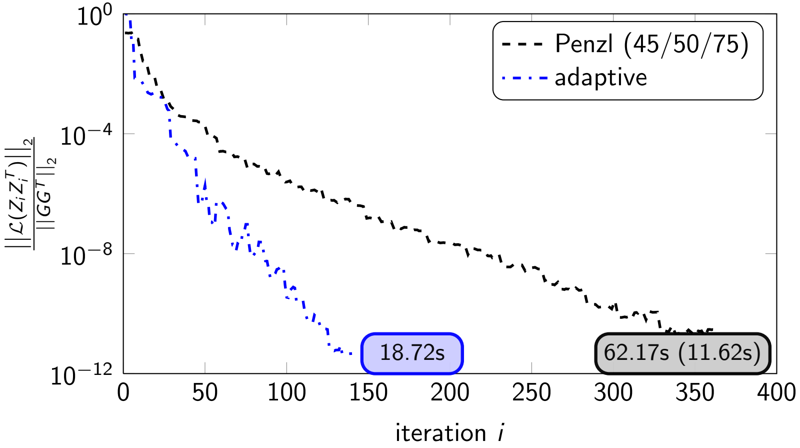
where

$$\tilde{F} = F_{11} - F_{12}F_{22}^{-1}F_{21}, \qquad \tilde{G} = G_1 - F_{12}F_{22}^{-1}G_2.$$

Numerical Experiments

Index-1 Power Systems Models e.g.[Freitas/Rommes/Martins '08]

BIPS_07[1] with $n = 21\,128$, $s = 3084$, $m = 4$



[1] https://sites.google.com/site/rommes/software

Conclusions

We have removed LR-ADIs two major disadvantages:

- Low rank residuals allow much faster evaluation of stopping criteria.

Conclusions

We have removed LR-ADIs two major disadvantages:

- Low rank residuals allow much faster evaluation of stopping criteria.
- Residual factors are now integral part of the algorithm $\rightsquigarrow$ almost no extra cost.

Conclusions

We have removed LR-ADIs two major disadvantages:

- Low rank residuals allow much faster evaluation of stopping criteria.
- Residual factors are now integral part of the algorithm $\rightsquigarrow$ almost no extra cost.
- Adaptive shifts automatize the algorithm and remove required expert knowledge in selecting “good” shifts.

Conclusions

We have removed LR-ADIs two major disadvantages:

- Low rank residuals allow much faster evaluation of stopping criteria.
- Residual factors are now integral part of the algorithm $\rightsquigarrow$ almost no extra cost.
- Adaptive shifts automatize the algorithm and remove required expert knowledge in selecting "good" shifts.

Thank you for your attention! Questions?