Optimal Control-Based Feedback Stabilization in Multi-Field Flow Problems

Eberhard Bänsch[1] Peter Benner[2,3] Jens Saak[2,3] Martin Stoll[2]
Heiko K. Weichelt[3]

[1] Department of Applied Mathematics III
Friedrich-Alexander-Universität Erlangen-Nürnberg

[2] Computational Methods in Systems and Control Theory
Max Planck Institute for Dynamics of Complex Technical Systems Magdeburg

[3] Department of Mathematics, Research group Mathematics in Industry and Technology
Chemnitz University of Technology

SPP1253 Annual Meeting
Banz, September 25 - 27 2011

Overview

1. Project Summary
   - Project Description
   - Optimal Control-Based Stabilization for NSEs
   - Previous Work
2. Solving Large Scale Saddle Point Systems
   - Key to Numerical Solution: Saddle Point System
   - Analysis of the Saddle Point System
   - Preconditioned GMRES
3. Numerical Examples
   - Problem Setting
   - GMRES

Project Summary

Scientific Goals of the Project

- Derive and investigate numerical algorithms for optimal control-based (normal and tangential) boundary feedback stabilization of multi-field flow problems.
- Explore the potentials and limitations of feedback-based (Riccati) stabilization techniques.
- Extend current methods for flow described by Navier-Stokes equations (NSE) to flow problems coupled with other field equations of increasing complexity.

Major Challenge

Numerical solution of algebraic Riccati equations associated to special LQR problem for linearized Navier-Stokes/Oseen-like equations.

Project Summary

Scientific Goals of the Project

- Derive and investigate numerical algorithms for optimal control-based (normal and tangential) boundary feedback stabilization of multi-field flow problems.
- Explore the potentials and limitations of feedback-based (Riccati) stabilization techniques.
- Extend current methods for flow described by Navier-Stokes equations (NSE) to flow problems coupled with other field equations of increasing complexity.

Major Challenge

Numerical solution of algebraic Riccati equations associated to special LQR problem for linearized Navier-Stokes/Oseen-like equations.

Optimal Control-Based Stabilization for NSEs

Analytical Solution

[Raymond '05–'07]

Linearized Navier-Stokes Control System

$$\begin{aligned} \partial_t \mathbf{z} - \nu \Delta \mathbf{z} + (\mathbf{z} \cdot \nabla)\mathbf{w} + (\mathbf{w} \cdot \nabla)\mathbf{z} - \omega \mathbf{z} + \nabla p = 0 \ \text{ in } Q_\infty, \\ \operatorname{div} \mathbf{z} = 0 \ \text{ in } Q_\infty, \\ \mathbf{z} = \mathbf{b}\mathbf{u} \text{ in } \Sigma_\infty, \\ \mathbf{z}(0) = \mathbf{z}_0 \ \text{ in } \Omega, \end{aligned}$$

$$J(\mathbf{z}, \mathbf{u}) = \frac{1}{2} \int_0^\infty \langle \mathbf{P}\mathbf{z}, \mathbf{P}\mathbf{z} \rangle_{L_2(\Omega)} + \rho\, \mathbf{u}^T \mathbf{u}\, dt.$$

Proposition [Raymond '05, Bahdra '09]

The solution to the instationary NSEs with perturbed initial data is exponentially controlled to the steady-state solution $\mathbf{w}$ by the feedback law

$$\mathbf{u} = -\rho^{-1} \mathbf{B}^* \mathbf{X} z_H.$$

- $z_H := \mathbf{P}\mathbf{z}$, with $\mathbf{P} : L_2(\Omega) \to V_n^0(\Omega)$ (Helmholtz projector) ($\rightsquigarrow \operatorname{div} \mathbf{z}_H \equiv 0$).
- $\mathbf{X} = \mathbf{X}^* \in \mathcal{L}(V_n^0(\Omega))$: unique nonnegative semidefinite weak solution of

$$0 = \mathbf{I} + (\mathbf{A} + \omega \mathbf{I})^* \mathbf{X} + \mathbf{X}(\mathbf{A} + \omega \mathbf{I}) - \mathbf{X}(\mathbf{B}_\tau \mathbf{B}_\tau^* + \rho^{-1} \mathbf{B}_n \mathbf{B}_n^*)\mathbf{X}.$$

Optimal Control-Based Stabilization for NSEs

Analytical Solution

[RAYMOND '05–'07]

Linearized Navier-Stokes Control System

$$\begin{aligned} \partial_t \mathbf{z} - \nu\Delta\mathbf{z} + (\mathbf{z}\cdot\nabla)\mathbf{w} + (\mathbf{w}\cdot\nabla)\mathbf{z} - \omega\mathbf{z} + \nabla p &= 0 \;\text{ in } Q_\infty,\\ \operatorname{div}\mathbf{z} &= 0 \;\text{ in } Q_\infty,\\ \mathbf{z} &= \mathbf{b}\mathbf{u} \text{ in } \Sigma_\infty,\\ \mathbf{z}(0) &= \mathbf{z}_0 \;\text{ in } \Omega, \end{aligned}$$

$$J(\mathbf{z},\mathbf{u}) = \frac{1}{2}\int_0^\infty \langle \mathbf{P}\mathbf{z}, \mathbf{P}\mathbf{z}\rangle_{L_2(\Omega)} + \rho\,\mathbf{u}^T\mathbf{u}\;dt.$$

Proposition [RAYMOND '05, BAHDRA '09]

The solution to the instationary NSEs with perturbed initial data is exponentially controlled to the steady-state solution $\mathbf{w}$ by the feedback law

$$\mathbf{u} = -\rho^{-1}\mathbf{B}^*\mathbf{X}z_H.$$

- $z_H := \mathbf{P}\mathbf{z}$, with $\mathbf{P} : L_2(\Omega) \to V_n^0(\Omega)$ (Helmholtz projector) ($\rightsquigarrow \operatorname{div} \mathbf{z}_H \equiv 0$).
- $\mathbf{X} = \mathbf{X}^* \in \mathcal{L}(V_n^0(\Omega))$: unique nonnegative semidefinite weak solution of

$$0 = \mathbf{I} + (\mathbf{A}+\omega\mathbf{I})^*\mathbf{X} + \mathbf{X}(\mathbf{A}+\omega\mathbf{I}) - \mathbf{X}(\mathbf{B}_\tau\mathbf{B}_\tau^* + \rho^{-1}\mathbf{B}_n\mathbf{B}_n^*)\mathbf{X}.$$

Optimal Control-Based Stabilization for NSEs

Analytical Solution

[Raymond '05–'07]

Linearized Navier-Stokes Control System

$$\partial_t \mathbf{z} - \nu\Delta\mathbf{z} + (\mathbf{z}\cdot\nabla)\mathbf{w} + (\mathbf{w}\cdot\nabla)\mathbf{z} - \omega\mathbf{z} + \nabla p = 0 \quad \text{in } Q_\infty,$$
$$\operatorname{div}\mathbf{z} = 0 \quad \text{in } Q_\infty,$$
$$\mathbf{z} = \mathbf{bu} \text{ in } \Sigma_\infty,$$
$$\mathbf{z}(0) = \mathbf{z}_0 \quad \text{in } \Omega,$$

$$J(\mathbf{z},\mathbf{u}) = \frac{1}{2}\int_0^\infty \langle \mathbf{Pz},\mathbf{Pz}\rangle_{L_2(\Omega)} + \rho\,\mathbf{u}^T\mathbf{u}\;dt.$$

Proposition [Raymond '05, Bahdra '09]

The solution to the instationary NSEs with perturbed initial data is exponentially controlled to the steady-state solution $\mathbf{w}$ by the feedback law

$$\mathbf{u} = -\rho^{-1}\mathbf{B}^*\mathbf{X}z_H.$$

- $\mathbf{z}_H := \mathbf{Pz}$, with $\mathbf{P} : L_2(\Omega) \to V_n^0(\Omega)$ (Helmholtz projector) ($\rightsquigarrow \operatorname{div}\mathbf{z}_H \equiv 0$).
- $\mathbf{X} = \mathbf{X}^* \in \mathcal{L}(V_n^0(\Omega))$: unique nonnegative semidefinite weak solution of

$$0 = \mathbf{I} + (\mathbf{A}+\omega\mathbf{I})^*\mathbf{X} + \mathbf{X}(\mathbf{A}+\omega\mathbf{I}) - \mathbf{X}(\mathbf{B}_\tau\mathbf{B}_\tau^* + \rho^{-1}\mathbf{B}_n\mathbf{B}_n^*)\mathbf{X}.$$

Optimal Control-Based Stabilization for NSEs

Analytical Solution

[Raymond '05–'07]

Linearized Navier-Stokes Control System

$$
\begin{aligned}
\partial_t \mathbf{z} - \nu\Delta\mathbf{z} + (\mathbf{z}\cdot\nabla)\mathbf{w} + (\mathbf{w}\cdot\nabla)\mathbf{z} - \omega\mathbf{z} + \nabla p &= 0 \;\text{ in } Q_\infty,\\
\operatorname{div}\mathbf{z} &= 0 \;\text{ in } Q_\infty,\\
\mathbf{z} &= \mathbf{b}\mathbf{u} \text{ in } \Sigma_\infty,\\
\mathbf{z}(0) &= \mathbf{z}_0 \;\text{ in } \Omega,
\end{aligned}
$$

$$
J(\mathbf{z},\mathbf{u}) = \frac{1}{2}\int_0^\infty \langle \mathbf{P}\mathbf{z}, \mathbf{P}\mathbf{z}\rangle_{L_2(\Omega)} + \rho\,\mathbf{u}^T\mathbf{u}\,dt.
$$

Proposition [Raymond '05, Bahdra '09]

The solution to the instationary NSEs with perturbed initial data is exponentially controlled to the steady-state solution $\mathbf{w}$ by the feedback law

$$
\mathbf{u} = -\rho^{-1}\mathbf{B}^*\mathbf{X}z_H.
$$

- $\mathbf{z}_H := \mathbf{P}\mathbf{z}$, with $\mathbf{P} : L_2(\Omega) \to V_n^0(\Omega)$ (Helmholtz projector) ($\rightsquigarrow \operatorname{div}\mathbf{z}_H \equiv 0$).
- $\mathbf{X} = \mathbf{X}^* \in \mathcal{L}(V_n^0(\Omega))$: unique nonnegative semidefinite weak solution of

$$
0 = \mathbf{I} + (\mathbf{A} + \omega\mathbf{I})^*\mathbf{X} + \mathbf{X}(\mathbf{A} + \omega\mathbf{I}) - \mathbf{X}(\mathbf{B}_\tau\mathbf{B}_\tau^* + \rho^{-1}\mathbf{B}_n\mathbf{B}_n^*)\mathbf{X}.
$$

Previous Work

1. Discretization of Helmholtz-projected linearized NSEs would need divergence-free finite elements.

   Here, we want to use standard discretization.

   (Taylor-Hood elements available in flow solver NAVIER)

   Explicit projection of ansatz functions possible using application of Helmholtz projection, but too expensive in general.

2. Each step of Newton-Kleinman iteration: Solve

$$A_j^T Z_{j+1} Z_{j+1}^T + Z_{j+1} Z_{j+1}^T A_j = -M - K_j^T K_j,$$

   $n_v := \operatorname{rank}(M) =$ dim of ansatz space for velocities.

   $\rightsquigarrow$ need to solve $n_v + m$ linear systems of equations in each step of Newton-ADI iteration!

3. Linearized system (i.e., $A + \omega M$) is unstable in general.

   But to start Newton iteration, a stabilizing initial guess is needed!

Previous Work

1. Discretization of Helmholtz-projected linearized NSEs would need divergence-free finite elements.

   Here, we want to use standard discretization.

   (Taylor-Hood elements available in flow solver NAVIER)

   Explicit projection of ansatz functions possible using application of Helmholtz projection, but too expensive in general.

2. Each step of Newton-Kleinman iteration: Solve

$$A_j^T Z_{j+1} Z_{j+1}^T + Z_{j+1} Z_{j+1}^T A_j = -M - K_j^T K_j,$$

   $n_v := \operatorname{rank}(M) =$ dim of ansatz space for velocities.

   $\rightsquigarrow$ need to solve $n_v + m$ linear systems of equations in each step of Newton-ADI iteration!

3. Linearized system (i.e., $A + \omega M$) is unstable in general.

   But to start Newton iteration, a stabilizing initial guess is needed!

Previous Work

1. Discretization of Helmholtz-projected linearized NSEs would need divergence-free finite elements.

   Here, we want to use standard discretization.

   (Taylor-Hood elements available in flow solver NAVIER)

   Explicit projection of ansatz functions possible using application of Helmholtz projection, but too expensive in general.

2. Each step of Newton-Kleinman iteration: Solve

$$A_j^T Z_{j+1} Z_{j+1}^T + Z_{j+1} Z_{j+1}^T A_j = -M - K_j^T K_j,$$

   $n_v := \operatorname{rank}(M) =$ dim of ansatz space for velocities.

   $\rightsquigarrow$ need to solve $n_v + m$ linear systems of equations in each step of Newton-ADI iteration!

3. Linearized system (i.e., $A + \omega M$) is unstable in general.

   But to start Newton iteration, a stabilizing initial guess is needed!

Previous Work

annual meeting Freising 2010

1. Discretization of Helmholtz-projected linearized NSEs would need divergence-free finite elements.

   Here, we want to use standard discretization.

   (Taylor-Hood elements available in flow solver NAVIER)

   Explicit projection of ansatz functions possible using application of Helmholtz projection, but too expensive in general.

2. Each step of Newton-Kleinman iteration: Solve

$$A_j^T Z_{j+1} Z_{j+1}^T + Z_{j+1} Z_{j+1}^T A_j = -M - K_j^T K_j,$$

   $n_v := \operatorname{rank}(M) =$ dim of ansatz space for velocities.

   $\rightsquigarrow$ need to solve $n_v + m$ linear systems of equations in each step of Newton-ADI iteration!

3. Linearized system (i.e., $A + \omega M$) is unstable in general.

   But to start Newton iteration, a stabilizing initial guess is needed!

Previous Work

1. Discretization of Helmholtz-projected linearized NSEs would need divergence-free finite elements.

   Here, we want to use standard discretization.

   (Taylor-Hood elements available in flow solver NAVIER)

   E[illegible] functions possible using application of H[illegible] projection, but too expensive in general.

   annual meeting Banz 2009

2. Each step of Newton-Kleinman iteration: Solve

$$A_j^T Z_{j+1} Z_{j+1}^T + Z_{j+1} Z_{j+1}^T A_j = -M - K_j^T K_j,$$

   $n_v := \mathrm{rank}\,(M) =$ dim of ansatz space for velocities.

   $\rightsquigarrow$ need to solve $n_v + m$ linear systems of equations in each step of Newton-ADI iteration!

3. Linearized system (i.e., $A + \omega M$) is unstable in general.

   But to start Newton iteration, a stabilizing initial guess is needed!

Previous Work

1. Discretization of Helmholtz-projected linearized NSEs would need divergence-free finite elements.

   Here, we want to use standard discretization.

   (Taylor-Hood elements available in flow solver NAVIER)

   Explicit projection of ansatz functions possible using application of Helmholtz projection, but too expensive in general.

2. Each step of Newton-Kleinman iteration: Solve

   [Benner '08-'10] *Partial Stabilization of Descriptor Systems Using Spectral Projectors*; to appear in V. Olshevsky et al (eds.), Numerical Linear Algebra in Signals, Systems and Control, Lecture Notes in Electrical Engineering, Springer-Verlag.

   [Hein '10] *MPC/LQG-Based Optimal Control of Nonlinear Parabolic PDEs*; PhD thesis Chemnitz UT.

3. Linearized system (i.e., $A + \omega M$) is unstable in general.

   But to start Newton iteration, a stabilizing initial guess is needed!

Key to Numerical Solution: Saddle Point System

Linearized NSE:

$$\partial_t \mathbf{z} - \nu \Delta \mathbf{z} + (\mathbf{z} \cdot \nabla)\mathbf{w} + (\mathbf{w} \cdot \nabla)\mathbf{z} + \nabla p = 0$$
$$\operatorname{div} \mathbf{z} = 0$$

Key to Numerical Solution: Saddle Point System

Linearized NSE:

$$\partial_t \mathbf{z} - \nu \Delta \mathbf{z} + (\mathbf{z} \cdot \nabla)\mathbf{w} + (\mathbf{w} \cdot \nabla)\mathbf{z} + \nabla p = 0$$
$$\operatorname{div} \mathbf{z} = 0$$

→

DAE:

$$M\dot{\underline{\tilde{z}}} = A\underline{\tilde{z}} + G\underline{\chi}$$
$$0 = G^T \underline{\tilde{z}}$$

Key to Numerical Solution: Saddle Point System

Linearized NSE:

$$\partial_t \mathbf{z} - \nu \Delta \mathbf{z} + (\mathbf{z} \cdot \nabla)\mathbf{w} + (\mathbf{w} \cdot \nabla)\mathbf{z} + \nabla p = 0$$
$$\operatorname{div} \mathbf{z} = 0$$

→

DAE:

$$M\dot{\underline{\tilde{z}}} = A\underline{\tilde{z}} + G\underline{\chi}$$
$$0 = G^T \underline{\tilde{z}}$$

→

State space system:

$$\mathcal{M}\dot{\underline{z}}(t) = \mathcal{A}\underline{z}(t) + \mathcal{B}\mathbf{u}(t)$$

with $\mathcal{M} = \mathcal{M}^T \succ 0$

Key to Numerical Solution: Saddle Point System

Linearized NSE:

$$\partial_t \mathbf{z} - \nu \Delta \mathbf{z} + (\mathbf{z} \cdot \nabla)\mathbf{w} + (\mathbf{w} \cdot \nabla)\mathbf{z} + \nabla p = 0$$
$$\operatorname{div} \mathbf{z} = 0$$

→

DAE:

$$M\dot{\underline{\tilde{z}}} = A\underline{\tilde{z}} + G\underline{\chi}$$
$$0 = G^T \underline{\tilde{z}}$$

→

State space system:

$$\mathcal{M}\dot{\underline{z}}(t) = \mathcal{A}\underline{z}(t) + \mathcal{B}\mathbf{u}(t)$$

with $\mathcal{M} = \mathcal{M}^T \succ 0$

→

Generalized algebraic Riccati equation:

$$\mathcal{R}(X) = \mathcal{M} + \mathcal{A}^T X \mathcal{M} + \mathcal{M} X \mathcal{A} - \mathcal{M} X \mathcal{B} \mathcal{B}^T X \mathcal{M} = 0$$

Key to Numerical Solution: Saddle Point System

Linearized NSE:
$$\partial_t \mathbf{z} - \nu \Delta \mathbf{z} + (\mathbf{z} \cdot \nabla)\mathbf{w} + (\mathbf{w} \cdot \nabla)\mathbf{z} + \nabla p = 0$$
$$\operatorname{div} \mathbf{z} = 0$$

→

DAE:
$$M\dot{\underline{\tilde{z}}} = A\underline{\tilde{z}} + G\underline{\chi}$$
$$0 = G^T \underline{\tilde{z}}$$

→

State space system:
$$\mathcal{M}\dot{\underline{z}}(t) = \mathcal{A}\underline{z}(t) + \mathcal{B}\mathbf{u}(t)$$
with $\mathcal{M} = \mathcal{M}^T \succ 0$

→

Generalized algebraic Riccati equation:
$$\mathcal{R}(X) = \mathcal{M} + \mathcal{A}^T X \mathcal{M} + \mathcal{M} X \mathcal{A} - \mathcal{M} X \mathcal{B}\mathcal{B}^T X \mathcal{M} = 0$$

↓

Newton iteration: $X^{k+1} = X^k + \tilde{N}^k$, where $\tilde{N}^k$ is solution of
$$(\mathcal{A} - \mathcal{B}\mathcal{B}^T X^k \mathcal{M})^T \tilde{N}^k \mathcal{M} + \mathcal{M}\tilde{N}^k (\mathcal{A} - \mathcal{B}\mathcal{B}^T X^k \mathcal{M}) = -\mathcal{R}(X^K)$$

Key to Numerical Solution: Saddle Point System

Linearized NSE:
$$\partial_t \mathbf{z} - \nu \Delta \mathbf{z} + (\mathbf{z} \cdot \nabla)\mathbf{w} + (\mathbf{w} \cdot \nabla)\mathbf{z} + \nabla p = 0$$
$$\operatorname{div} \mathbf{z} = 0$$

→

DAE:
$$M\dot{\underline{\tilde{z}}} = A\underline{\tilde{z}} + G\underline{\chi}$$
$$0 = G^T \underline{\tilde{z}}$$

→

State space system:
$$\mathcal{M}\dot{\underline{z}}(t) = \mathcal{A}\underline{z}(t) + \mathcal{B}\mathbf{u}(t)$$
with $\mathcal{M} = \mathcal{M}^T \succ 0$

↓

Generalized algebraic Riccati equation:
$$\mathcal{R}(X) = \mathcal{M} + \mathcal{A}^T X \mathcal{M} + \mathcal{M} X \mathcal{A} - \mathcal{M} X \mathcal{B}\mathcal{B}^T X \mathcal{M} = 0$$

↓

Newton iteration: $X^{k+1} = X^k + \tilde{N}^k$, where $\tilde{N}^k$ is solution of
$$(\mathcal{A} - \mathcal{B}\mathcal{B}^T X^k \mathcal{M})^T \tilde{N}^k \mathcal{M} + \mathcal{M}\tilde{N}^k (\mathcal{A} - \mathcal{B}\mathcal{B}^T X^k \mathcal{M}) = -\mathcal{R}(X^K)$$

↓

Lyapunov equation $\Rightarrow$ ADI-Method:
$$\mathcal{A}_k^T \tilde{N}^k \mathcal{M} + \mathcal{M}\tilde{N}^k \mathcal{A}_k = -\mathcal{W}_k^T \mathcal{W}_k$$

Key to Numerical Solution: Saddle Point System

Linearized NSE:

$$\partial_t \mathbf{z} - \nu \Delta \mathbf{z} + (\mathbf{z} \cdot \nabla)\mathbf{w} + (\mathbf{w} \cdot \nabla)\mathbf{z} + \nabla p = 0$$

$$\operatorname{div} \mathbf{z} = 0$$

→

DAE:

$$M\dot{\underline{\tilde{z}}} = A\underline{\tilde{z}} + G\underline{\chi}$$

$$0 = G^T \underline{\tilde{z}}$$

→

State space system:

$$\mathcal{M}\dot{\underline{z}}(t) = \mathcal{A}\underline{z}(t) + \mathcal{B}\mathbf{u}(t)$$

with $\mathcal{M} = \mathcal{M}^T \succ 0$

↓

Generalized algebraic Riccati equation:

$$\mathcal{R}(X) = \mathcal{M} + \mathcal{A}^T X \mathcal{M} + \mathcal{M} X \mathcal{A} - \mathcal{M} X \mathcal{B}\mathcal{B}^T X \mathcal{M} = 0$$

↓

Newton iteration: $X^{k+1} = X^k + \tilde{N}^k$, where $\tilde{N}^k$ is solution of

$$(\mathcal{A} - \mathcal{B}\mathcal{B}^T X^k \mathcal{M})^T \tilde{N}^k \mathcal{M} + \mathcal{M}\tilde{N}^k(\mathcal{A} - \mathcal{B}\mathcal{B}^T X^k \mathcal{M}) = -\mathcal{R}(X^K)$$

↓

Lyapunov equation $\Rightarrow$ ADI-Method:

$$\mathcal{A}_k^T \tilde{N}^k \mathcal{M} + \mathcal{M}\tilde{N}^k \mathcal{A}_k = -\mathcal{W}_k^T \mathcal{W}_k$$

↓

Saddle Point System: [Heinkenschloss, Sorensen, Sun '08]

$$\begin{bmatrix} A^T + p_j M - K^k B^T & G \\ G^T & 0 \end{bmatrix} \begin{bmatrix} \Lambda \\ * \end{bmatrix} = \begin{bmatrix} W \\ 0 \end{bmatrix}$$

p_j ... ADI shift parameter

$K^k := MX^k B$ ... Feedback operator

Key to Numerical Solution: Saddle Point System

Linearized NSE:

$$\partial_t \mathbf{z} - \nu \Delta \mathbf{z} + (\mathbf{z} \cdot \nabla)\mathbf{w} + (\mathbf{w} \cdot \nabla)\mathbf{z} + \nabla p = 0$$
$$\operatorname{div} \mathbf{z} = 0$$

→

DAE:

$$M\dot{\underline{\tilde{z}}} = A\underline{\tilde{z}} + G\underline{\chi}$$
$$0 = G^T \underline{\tilde{z}}$$

→

State space system:

$$\mathcal{M}\dot{\underline{z}}(t) = \mathcal{A}\underline{z}(t) + \mathcal{B}\mathbf{u}(t)$$

with $\mathcal{M} = \mathcal{M}^T \succ 0$

↓

Generalized algebraic Riccati equation:

$$\mathcal{R}(X) = \mathcal{M} + \mathcal{A}^T X \mathcal{M} + \mathcal{M} X \mathcal{A} - \mathcal{M} X \mathcal{B} \mathcal{B}^T X \mathcal{M} = 0$$

↓

Newton iteration: $X^{k+1} = X^k + \tilde{N}^k$, where $\tilde{N}^k$ is solution of

$$(\mathcal{A} - \mathcal{B}\mathcal{B}^T X^k \mathcal{M})^T \tilde{N}^k \mathcal{M} + \mathcal{M} \tilde{N}^k (\mathcal{A} - \mathcal{B}\mathcal{B}^T X^k \mathcal{M}) = -\mathcal{R}(X^K)$$

↓

Lyapunov equation ⇒ ADI-Method:

$$\mathcal{A}_k^T \tilde{N}^k \mathcal{M} + \mathcal{M} \tilde{N}^k \mathcal{A}_k = -\mathcal{W}_k^T \mathcal{W}_k$$

↓

Saddle Point System: [Heinkenschloss, Sorensen, Sun '08]

$$\begin{bmatrix} A^T + p_j M - K^k B^T & G \\ G^T & 0 \end{bmatrix} \begin{bmatrix} \Lambda \\ * \end{bmatrix} = \begin{bmatrix} W \\ 0 \end{bmatrix}$$

p_j ... ADI shift parameter

$K^k := M X^k B$ ... Feedback operator

Analysis of the Saddle Point System

Properties

$$\begin{bmatrix} A^T + p_j M - K^k B^T & G \\ G^T & 0 \end{bmatrix} \begin{bmatrix} \Lambda \\ * \end{bmatrix} = \begin{bmatrix} W \\ 0 \end{bmatrix}$$

- $A^T \neq A \in \mathbb{R}^{n_v \times n_v}$, sparse, discretized lin. Nav.-St. operator, constant
- $p_j \in \mathbb{R}^-$, changes in every ADI step
- $M \in \mathbb{R}^{n_v \times n_v}$, spd, sparse, mass matrix, constant
- $K^k \in \mathbb{R}^{n_v \times n_r}$, dense, feedback operator, changes in every Newton step
- $B \in \mathbb{R}^{n_v \times n_r}$, highly sparse, boundary operator, constant
- $G \in \mathbb{R}^{n_v \times n_p}$, sparse, full column rank, discretized gradient, constant

Solver

- Try to avoid direct solvers because of high dimensions.
- Use Krylov subspace method for non symmetric problems. $\Rightarrow$ GMRES

Analysis of the Saddle Point System

Properties

$$\begin{bmatrix} A^T + p_j M - K^k B^T & G \\ G^T & 0 \end{bmatrix} \begin{bmatrix} \Lambda \\ * \end{bmatrix} = \begin{bmatrix} W \\ 0 \end{bmatrix}$$

- $A^T \neq A \in \mathbb{R}^{n_v \times n_v}$, sparse, discretized lin. Nav.-St. operator, constant
- $p_j \in \mathbb{R}^-$, changes in every ADI step
- $M \in \mathbb{R}^{n_v \times n_v}$, spd, sparse, mass matrix, constant
- $K^k \in \mathbb{R}^{n_v \times n_r}$, dense, feedback operator, changes in every Newton step
- $B \in \mathbb{R}^{n_v \times n_r}$, highly sparse, boundary operator, constant
- $G \in \mathbb{R}^{n_v \times n_p}$, sparse, full column rank, discretized gradient, constant

Solver

- Try to avoid direct solvers because of high dimensions.
- Use Krylov subspace method for non symmetric problems. $\Rightarrow$ GMRES

Preconditioned GMRES

Left Preconditioner

[Elman, Silvester, Wathen '06]

Derivation

$$\begin{bmatrix} A^T + p_j M - K^k B^T & G \\ G^T & 0 \end{bmatrix}$$

Preconditioned GMRES

Left Preconditioner

[Elman, Silvester, Wathen '06]

Derivation

$$\begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} \qquad \mathbf{F} \neq \mathbf{F}^T$$

Preconditioned GMRES

Left Preconditioner

[Elman, Silvester, Wathen '06]

Derivation

$$\begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} \qquad \mathbf{F} \neq \mathbf{F}^T$$

- No need for symmetric preconditioner:

Preconditioned GMRES

Left Preconditioner

[ELMAN, SILVESTER, WATHEN '06]

Derivation

$$\begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} \qquad \mathbf{F} \neq \mathbf{F}^T$$

- No need for symmetric preconditioner:

$$P_l = \begin{bmatrix} \mathbf{P_F} & 0 \\ G^T & -P_{SC} \end{bmatrix} \Rightarrow P_l^{-1} = \begin{bmatrix} \mathbf{P_F}^{-1} & 0 \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1} & -P_{SC}^{-1} \end{bmatrix}$$

Preconditioned GMRES

Left Preconditioner

[Elman, Silvester, Wathen '06]

Derivation

$$\begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} \qquad \mathbf{F} \neq \mathbf{F}^T$$

- No need for symmetric preconditioner:

$$P_l = \begin{bmatrix} \mathbf{P_F} & 0 \\ G^T & -P_{SC} \end{bmatrix} \Rightarrow P_l^{-1} = \begin{bmatrix} \mathbf{P_F}^{-1} & 0 \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1} & -P_{SC}^{-1} \end{bmatrix}$$

- Choice of $\mathbf{P_F}, P_{SC}$:

Preconditioned GMRES

Left Preconditioner

[Elman, Silvester, Wathen '06]

Derivation

$$\begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} \qquad \mathbf{F} \neq \mathbf{F}^T$$

- No need for symmetric preconditioner:

$$P_l = \begin{bmatrix} \mathbf{P_F} & 0 \\ G^T & -P_{SC} \end{bmatrix} \Rightarrow P_l^{-1} = \begin{bmatrix} \mathbf{P_F}^{-1} & 0 \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1} & -P_{SC}^{-1} \end{bmatrix}$$

- Choice of $\mathbf{P_F}, P_{SC}$:

$$\begin{bmatrix} \mathbf{P_F}^{-1} & 0 \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1} & -P_{SC}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{P_F}^{-1}\mathbf{F} & * \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1}\mathbf{F} - P_{SC}^{-1} G^T & P_{SC}^{-1} G^T \mathbf{P_F}^{-1} G \end{bmatrix}$$

Preconditioned GMRES

Left Preconditioner

[Elman, Silvester, Wathen '06]

Derivation

$$\begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} \qquad \mathbf{F} \neq \mathbf{F}^T$$

- No need for symmetric preconditioner:

$$P_l = \begin{bmatrix} \mathbf{P_F} & 0 \\ G^T & -P_{SC} \end{bmatrix} \Rightarrow P_l^{-1} = \begin{bmatrix} \mathbf{P_F}^{-1} & 0 \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1} & -P_{SC}^{-1} \end{bmatrix}$$

- Choice of $\mathbf{P_F}, P_{SC}$:

$$\begin{bmatrix} \mathbf{P_F}^{-1} & 0 \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1} & -P_{SC}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} = \begin{bmatrix} I & * \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1} \mathbf{F} - P_{SC}^{-1} G^T & P_{SC}^{-1} G^T \mathbf{P_F}^{-1} G \end{bmatrix}$$

Preconditioned GMRES

Left Preconditioner

[Elman, Silvester, Wathen '06]

Derivation

$$\begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} \qquad \mathbf{F} \neq \mathbf{F}^T$$

- No need for symmetric preconditioner:

$$P_l = \begin{bmatrix} \mathbf{P_F} & 0 \\ G^T & -P_{SC} \end{bmatrix} \Rightarrow P_l^{-1} = \begin{bmatrix} \mathbf{P_F}^{-1} & 0 \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1} & -P_{SC}^{-1} \end{bmatrix}$$

- Choice of $\mathbf{P_F}, P_{SC}$:

$$\begin{bmatrix} \mathbf{P_F}^{-1} & 0 \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1} & -P_{SC}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} = \begin{bmatrix} I & * \\ 0 & P_{SC}^{-1} G^T \mathbf{P_F}^{-1} G \end{bmatrix}$$

Preconditioned GMRES

Left Preconditioner

[Elman, Silvester, Wathen '06]

Derivation

$$\begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} \qquad \mathbf{F} \neq \mathbf{F}^T$$

- No need for symmetric preconditioner:

$$P_l = \begin{bmatrix} \mathbf{P_F} & 0 \\ G^T & -P_{SC} \end{bmatrix} \Rightarrow P_l^{-1} = \begin{bmatrix} \mathbf{P_F}^{-1} & 0 \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1} & -P_{SC}^{-1} \end{bmatrix}$$

- Choice of $\mathbf{P_F}, P_{SC}$:

$$\begin{bmatrix} \mathbf{P_F}^{-1} & 0 \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1} & -P_{SC}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} = \begin{bmatrix} I & * \\ 0 & P_{SC}^{-1} G^T \mathbf{F}^{-1} G \end{bmatrix}$$

Preconditioned GMRES

Left Preconditioner

[Elman, Silvester, Wathen '06]

Derivation

$$\begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} \qquad \mathbf{F} \neq \mathbf{F}^T$$

- No need for symmetric preconditioner:

$$P_l = \begin{bmatrix} \mathbf{P_F} & 0 \\ G^T & -P_{SC} \end{bmatrix} \Rightarrow P_l^{-1} = \begin{bmatrix} \mathbf{P_F}^{-1} & 0 \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1} & -P_{SC}^{-1} \end{bmatrix}$$

- Choice of $\mathbf{P_F}, P_{SC}$:

$$\begin{bmatrix} \mathbf{P_F}^{-1} & 0 \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1} & -P_{SC}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} = \begin{bmatrix} I & * \\ 0 & P_{SC}^{-1} G^T \mathbf{F}^{-1} G \end{bmatrix}$$

Preconditioned GMRES

Left Preconditioner

[Elman, Silvester, Wathen '06]

Derivation

$$\begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} \qquad \mathbf{F} \neq \mathbf{F}^T$$

- No need for symmetric preconditioner:

$$P_l = \begin{bmatrix} \mathbf{P_F} & 0 \\ G^T & -P_{SC} \end{bmatrix} \Rightarrow P_l^{-1} = \begin{bmatrix} \mathbf{P_F}^{-1} & 0 \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1} & -P_{SC}^{-1} \end{bmatrix}$$

- Choice of $\mathbf{P_F}, P_{SC}$:

$$\begin{bmatrix} \mathbf{P_F}^{-1} & 0 \\ P_{SC}^{-1} G^T \mathbf{P_F}^{-1} & -P_{SC}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{F} & G \\ G^T & 0 \end{bmatrix} = \begin{bmatrix} I & * \\ 0 & P_{SC}^{-1} G^T \mathbf{F}^{-1} G \end{bmatrix}$$

Schur Complement

- Choose *Schur complement* for lower right block: $P_{SC} = G^T \mathbf{F}^{-1} G$
- Can not build this matrix (dense, high dimension).
- Have to find a good approximation for P_{SC}.

Preconditioned GMRES

Schur Complement

Section 8.2 [Elman, Silvester, Wathen '06]

Approximation to the Schur Complement Operator

Consider the discrete version of a shifted Navier-Stokes operator:

$$\mathcal{L} = -\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I$$

Preconditioned GMRES

Schur Complement

Section 8.2 [Elman, Silvester, Wathen '06]

Approximation to the Schur Complement Operator

Consider the discrete version of a shifted Navier-Stokes operator:

$$\mathcal{L} = -\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I$$

Suppose analogous operator on pressure space:

$$\mathcal{L}_p = (-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)_p$$

Preconditioned GMRES

Schur Complement

Section 8.2 [Elman, Silvester, Wathen '06]

Approximation to the Schur Complement Operator

Consider the discrete version of a shifted Navier-Stokes operator:

$$\mathcal{L} = -\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I$$

Suppose analogous operator on pressure space:

$$\mathcal{L}_p = (-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)_p$$

Consider a small commutator of $\mathcal{L}$ with the gradient:

$$\mathcal{E} = (-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)\nabla - \nabla(-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)_p$$

Preconditioned GMRES

Schur Complement

Section 8.2 [Elman, Silvester, Wathen '06]

Approximation to the Schur Complement Operator

Consider the discrete version of a shifted Navier-Stokes operator:

$$\mathcal{L} = -\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I$$

Suppose analogous operator on pressure space:

$$\mathcal{L}_p = (-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)_p$$

Consider a small commutator of $\mathcal{L}$ with the gradient:

$$\mathcal{E} = (-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)\nabla - \nabla(-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)_p$$

Discrete version of commutator in terms of finite element matrices:

$$\mathcal{E}_h = (M^{-1}\mathbf{F})(M^{-1}G) - (M^{-1}G)(M_p^{-1}F_p)$$

Preconditioned GMRES

Schur Complement

Section 8.2 [Elman, Silvester, Wathen '06]

Approximation to the Schur Complement Operator

Consider the discrete version of a shifted Navier-Stokes operator:

$$\mathcal{L} = -\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I$$

Suppose analogous operator on pressure space:

$$\mathcal{L}_p = (-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)_p$$

Consider a small commutator of $\mathcal{L}$ with the gradient:

$$\mathcal{E} = (-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)\nabla - \nabla(-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)_p$$

Discrete version of commutator in terms of finite element matrices:

$$\mathcal{E}_h = (M^{-1}\mathbf{F})(M^{-1}G) - (M^{-1}G)(M_p^{-1}F_p)$$

Premultiplying with $G^T\mathbf{F}^{-1}M$, postmultiplying with $F_p^{-1}M_p$:

$$G^T\mathbf{F}^{-1}G \approx G^T M^{-1} G F_p^{-1} M_p$$

Preconditioned GMRES

Schur Complement

Section 8.2 [Elman, Silvester, Wathen '06]

Approximation to the Schur Complement Operator

Consider the discrete version of a shifted Navier-Stokes operator:

$$\mathcal{L} = -\nu\nabla^2 + \vec{w_h}\cdot\nabla + \nabla\cdot\vec{w_h} + p_j I$$

Suppose analogous operator on pressure space:

$$\mathcal{L}_p = (-\nu\nabla^2 + \vec{w_h}\cdot\nabla + \nabla\cdot\vec{w_h} + p_j I)_p$$

Consider a small commutator of $\mathcal{L}$ with the gradient:

$$\mathcal{E} = (-\nu\nabla^2 + \vec{w_h}\cdot\nabla + \nabla\cdot\vec{w_h} + p_j I)\nabla - \nabla(-\nu\nabla^2 + \vec{w_h}\cdot\nabla + \nabla\cdot\vec{w_h} + p_j I)_p$$

Discrete version of commutator in terms of finite element matrices:

$$\mathcal{E}_h = (M^{-1}\mathbf{F})(M^{-1}G) - (M^{-1}G)(M_p^{-1}F_p)$$

Premultiplying with $G^T\mathbf{F}^{-1}M$, postmultiplying with $F_p^{-1}M_p$:

$$G^T\mathbf{F}^{-1}G \approx G^T M^{-1} G F_p^{-1} M_p$$

Preconditioned GMRES

Schur Complement

Section 8.2 [Elman, Silvester, Wathen '06]

Approximation to the Schur Complement Operator

Consider the discrete version of a shifted Navier-Stokes operator:

$$\mathcal{L} = -\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I$$

Suppose analogous operator on pressure space:

$$\mathcal{L}_p = (-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)_p$$

Consider a small commutator of $\mathcal{L}$ with the gradient:

$$\mathcal{E} = (-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)\nabla - \nabla(-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)_p$$

Discrete version of commutator in terms of finite element matrices:

$$\mathcal{E}_h = (M^{-1}\mathbf{F})(M^{-1}G) - (M^{-1}G)(M_p^{-1}F_p)$$

Premultiplying with $G^T\mathbf{F}^{-1}M$, postmultiplying with $F_p^{-1}M_p$:

$$G^T\mathbf{F}^{-1}G \approx S_p F_p^{-1} M_p$$

Preconditioned GMRES

Schur Complement

Section 8.2 [Elman, Silvester, Wathen '06]

Approximation to the Schur Complement Operator

Consider the discrete version of a shifted Navier-Stokes operator:

$$\mathcal{L} = -\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I$$

Suppose analogous operator on pressure space:

$$\mathcal{L}_p = (-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)_p$$

Consider a small commutator of $\mathcal{L}$ with the gradient:

$$\mathcal{E} = (-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)\nabla - \nabla(-\nu\nabla^2 + \vec{w}_h \cdot \nabla + \nabla \cdot \vec{w}_h + p_j I)_p$$

Discrete version of commutator in terms of finite element matrices:

$$\mathcal{E}_h = (M^{-1}\mathbf{F})(M^{-1}G) - (M^{-1}G)(M_p^{-1}F_p)$$

Premultiplying with $G^T\mathbf{F}^{-1}M$, postmultiplying with $F_p^{-1}M_p$:

$$G^T\mathbf{F}^{-1}G \approx S_p F_p^{-1} M_p$$

Schur complement approximation:

$$P_{SC} := S_p F_p^{-1} M_p$$

$$\Rightarrow P_{SC}^{-1} = M_p^{-1} F_p S_p^{-1}$$

Preconditioned GMRES

Preconditioner

[Elman, Silvester, Wathen '06]

Approximation of $\mathbf{P_F}$

- Need approximation for $\mathbf{P_F}$.
- For high Reynolds numbers the approximation has to get more accurate.
- In the future we want to use some V-cycle `Multigrid` steps.

Preconditioned GMRES

Preconditioner

[Elman, Silvester, Wathen '06]

Approximation of $\mathbf{P_F}$

- Need approximation for $\mathbf{P_F}$.
- For high Reynolds numbers the approximation has to get more accurate.
- In the future we want to use some V-cycle `Multigrid` steps.
- Current proof of concept uses direct solvers in MATLAB ("\").

Preconditioned GMRES

Preconditioner

[Elman, Silvester, Wathen '06]

Approximation of $\mathbf{P_F}$

- Need approximation for $\mathbf{P_F}$.
- For high Reynolds numbers the approximation has to get more accurate.
- In the future we want to use some V-cycle `Multigrid` steps.
- Current proof of concept uses direct solvers in MATLAB ("\").

Simplification

We considered $\mathbf{F}$ instead of $A^T + p_j M - KB^T$.
Why can we do this?
$\Rightarrow$ *Feedback-invariance of Krylov Subspaces*

Preconditioned GMRES

Feedback-invariance of the Krylov Subspaces

Saddle Point System

$$\begin{bmatrix} A^T + p_j M - K^k B^T & G \\ G^T & 0 \end{bmatrix} \begin{bmatrix} \Lambda \\ * \end{bmatrix} = \begin{bmatrix} W \\ 0 \end{bmatrix}$$

Preconditioned GMRES

Feedback-invariance of the Krylov Subspaces

Saddle Point System

$$\begin{bmatrix} A^T + p_j M - K^k B^T & G \\ G^T & 0 \end{bmatrix} \begin{bmatrix} \Lambda \\ * \end{bmatrix} = \begin{bmatrix} [C^T K^k] \\ 0 \end{bmatrix}$$

Preconditioned GMRES

Feedback-invariance of the Krylov Subspaces

Saddle Point System

$$\left(\begin{bmatrix} A^T + p_j M & G \\ G^T & 0 \end{bmatrix} - \begin{bmatrix} K^k \\ 0 \end{bmatrix} \cdot \begin{bmatrix} B \\ 0 \end{bmatrix}^T \right) \begin{bmatrix} \Lambda \\ * \end{bmatrix} = \left[\begin{bmatrix} C^T \\ 0 \end{bmatrix} \begin{bmatrix} K^k \\ 0 \end{bmatrix} \right]$$

Preconditioned GMRES

Feedback-invariance of the Krylov Subspaces

Saddle Point System

$$\left(\mathbf{F_S} - \begin{bmatrix} K^k \\ 0 \end{bmatrix} \cdot \begin{bmatrix} B \\ 0 \end{bmatrix}^T \right) \begin{bmatrix} \Lambda \\ * \end{bmatrix} = \left[\begin{bmatrix} C^T \\ 0 \end{bmatrix} \begin{bmatrix} K^k \\ 0 \end{bmatrix} \right]$$

Preconditioned GMRES

Feedback-invariance of the Krylov Subspaces

Saddle Point System

$$\left(\mathbf{F_S} - \mathbf{K}^k \cdot \begin{bmatrix} B \\ 0 \end{bmatrix}^T\right) \begin{bmatrix} \Lambda \\ * \end{bmatrix} = \left[\begin{bmatrix} C^T \\ 0 \end{bmatrix} \mathbf{K}^k \right]$$

Preconditioned GMRES

Feedback-invariance of the Krylov Subspaces

Saddle Point System

$$\left(\mathbf{F_S} - \mathbf{K}^k \cdot \mathbf{B}^T\right) \begin{bmatrix} \Lambda \\ * \end{bmatrix} = \left[\begin{bmatrix} C^T \\ 0 \end{bmatrix} \mathbf{K}^k \right]$$

Preconditioned GMRES

Feedback-invariance of the Krylov Subspaces

Saddle Point System

$$\left(\mathbf{F_S} - \mathbf{K}^k \cdot \mathbf{B}^T\right) \mathbf{\Lambda} = \left[\begin{bmatrix} C^T \\ 0 \end{bmatrix} \mathbf{K}^k \right]$$

Preconditioned GMRES

Feedback-invariance of the Krylov Subspaces

Saddle Point System

$$\left(\mathbf{F_S} - \mathbf{K}^k \cdot \mathbf{B}^T\right) \mathbf{\Lambda} = \left[\mathbf{C}^T \mathbf{K}^k\right]$$

Preconditioned GMRES

Feedback-invariance of the Krylov Subspaces

Saddle Point System

$$\left(\mathbf{F_S} - \mathbf{K}^k \cdot \mathbf{B}^T\right)\mathbf{\Lambda} = \left[\mathbf{C}^T \mathbf{K}^k\right]$$

$$\mathbf{A}^k \mathbf{\Lambda} = \mathbf{W}^k$$

Preconditioned GMRES

Feedback-invariance of the Krylov Subspaces

Saddle Point System

$$\left(\mathbf{F_S} - \mathbf{K}^k \cdot \mathbf{B}^T\right) \mathbf{\Lambda} = \left[\mathbf{C}^T \mathbf{K}^k\right]$$

$$\mathbf{A}^k \mathbf{\Lambda} = \mathbf{W}^k$$

Feedback Invariance of Subspaces [Benner/Beckermann '11]

$$\mathcal{K}_m(\mathbf{A}^k, \mathbf{W}^k) = \mathcal{K}_m(\mathbf{F_S}, \mathbf{W}^k)$$

Preconditioned GMRES

Feedback-invariance of the Krylov Subspaces

Saddle Point System

$$\left(\mathbf{F_S} - \mathbf{K}^k \cdot \mathbf{B}^T\right)\boldsymbol{\Lambda} = \left[\mathbf{C}^T\mathbf{K}^k\right]$$
$$\mathbf{A}^k\boldsymbol{\Lambda} = \mathbf{W}^k$$

Feedback Invariance of Subspaces [Benner/Beckermann '11]

$$\mathcal{K}_m(\mathbf{A}^k, \mathbf{W}^k) = \mathcal{K}_m(\mathbf{F_S}, \mathbf{W}^k)$$

$$\begin{aligned}
\mathcal{K}_m(\mathbf{A}^k, \mathbf{W}^k) &= \mathcal{K}_m(\mathbf{F_S} - \mathbf{K}^k\mathbf{B}^T), [\mathbf{C}^T\,\mathbf{K}^k])\\
&= \mathrm{sp}[\mathbf{C}^T, \mathbf{K}^k, \mathbf{F_S}\mathbf{C}^T - \mathbf{K}^k\mathbf{B}^T\mathbf{C}^T, \mathbf{F_S}\mathbf{K}^k - \mathbf{K}^k\mathbf{B}^T\mathbf{K}^k, \ldots]\\
&= \mathrm{sp}[\mathbf{C}^T, \mathbf{K}^k, \mathbf{F_S}\mathbf{C}^T, \mathbf{F_S}\mathbf{K}^k, \ldots]\\
&= \mathcal{K}_m(\mathbf{F_S}, [\mathbf{C}^T\,\mathbf{K}^k]) = \mathcal{K}_m(\mathbf{F_S}, \mathbf{W}^k)
\end{aligned}$$

Preconditioned GMRES

Feedback-invariance of the Krylov Subspaces

Saddle Point System

$$\left(\mathbf{F_S} - \mathbf{K}^k \cdot \mathbf{B}^T\right) \mathbf{\Lambda} = \left[\mathbf{C}^T \mathbf{K}^k\right]$$
$$\mathbf{A}^k \mathbf{\Lambda} = \mathbf{W}^k$$

Feedback Invariance of Subspaces [Benner/Beckermann '11]

$$\mathcal{K}_m(\mathbf{A}^k, \mathbf{W}^k) = \mathcal{K}_m(\mathbf{F_S}, \mathbf{W}^k)$$

$$\begin{aligned}
\mathcal{K}_m(\mathbf{A}^k, \mathbf{W}^k) &= \mathcal{K}_m(\mathbf{F_S} - \mathbf{K}^k\mathbf{B}^T), [\mathbf{C}^T\,\mathbf{K}^k]) \\
&= \mathrm{sp}[\mathbf{C}^T, \mathbf{K}^k, \mathbf{F_S}\mathbf{C}^T - \mathbf{K}^k\mathbf{B}^T\mathbf{C}^T, \mathbf{F_S}\mathbf{K}^k - \mathbf{K}^k\mathbf{B}^T\mathbf{K}^k, \ldots] \\
&= \mathrm{sp}[\mathbf{C}^T, \mathbf{K}^k, \mathbf{F_S}\mathbf{C}^T, \mathbf{F_S}\mathbf{K}^k, \ldots] \\
&= \mathcal{K}_m(\mathbf{F_S}, [\mathbf{C}^T\,\mathbf{K}^k]) = \mathcal{K}_m(\mathbf{F_S}, \mathbf{W}^k)
\end{aligned}$$

Preconditioned GMRES

Feedback-invariance of the Krylov Subspaces

Saddle Point System

$$\left(\mathbf{F_S} - \mathbf{K}^k \cdot \mathbf{B}^T\right)\mathbf{\Lambda} = \left[\mathbf{C}^T \mathbf{K}^k\right]$$

$$\mathbf{A}^k \mathbf{\Lambda} = \mathbf{W}^k$$

$$\mathrm{sp}[-\mathbf{K}^k\mathbf{B}^T\mathbf{C}^T] \subseteq \mathrm{sp}[\mathbf{K}^k]$$

$$\mathrm{sp}[-\mathbf{K}^k\mathbf{B}^T\mathbf{K}^k] \subseteq \mathrm{sp}[\mathbf{K}^k]$$

Feedback Invariance of Subspaces [Benner/Beckermann '11]

$$\mathcal{K}_m(\mathbf{A}^k, \mathbf{W}^k) = \mathcal{K}_m(\mathbf{F_S}, \mathbf{W}^k)$$

$$\begin{aligned}
\mathcal{K}_m(\mathbf{A}^k, \mathbf{W}^k) &= \mathcal{K}_m(\mathbf{F_S} - \mathbf{K}^k\mathbf{B}^T), [\mathbf{C}^T\ \mathbf{K}^k]) \\
&= \mathrm{sp}[\mathbf{C}^T, \mathbf{K}^k, \mathbf{F_S}\mathbf{C}^T - \mathbf{K}^k\mathbf{B}^T\mathbf{C}^T, \mathbf{F_S}\mathbf{K}^k - \mathbf{K}^k\mathbf{B}^T\mathbf{K}^k, \ldots] \\
&= \mathrm{sp}[\mathbf{C}^T, \mathbf{K}^k, \mathbf{F_S}\mathbf{C}^T, \mathbf{F_S}\mathbf{K}^k, \ldots] \\
&= \mathcal{K}_m(\mathbf{F_S}, [\mathbf{C}^T\ \mathbf{K}^k]) = \mathcal{K}_m(\mathbf{F_S}, \mathbf{W}^k)
\end{aligned}$$

Preconditioned GMRES

Feedback-invariance of the Krylov Subspaces

Saddle Point System

$$\left(\mathbf{F_S} - \mathbf{K}^k \cdot \mathbf{B}^T\right)\mathbf{\Lambda} = \left[\mathbf{C}^T\mathbf{K}^k\right]$$

$$\mathbf{A}^k\mathbf{\Lambda} = \mathbf{W}^k$$

Feedback Invariance of Subspaces [Benner/Beckermann '11]

$$\mathcal{K}_m(\mathbf{A}^k, \mathbf{W}^k) = \mathcal{K}_m(\mathbf{F_S}, \mathbf{W}^k)$$

$$\begin{aligned}
\mathcal{K}_m(\mathbf{A}^k, \mathbf{W}^k) &= \mathcal{K}_m(\mathbf{F_S} - \mathbf{K}^k\mathbf{B}^T), [\mathbf{C}^T\,\mathbf{K}^k])\\
&= \mathrm{sp}[\mathbf{C}^T, \mathbf{K}^k, \mathbf{F_S}\mathbf{C}^T - \mathbf{K}^k\mathbf{B}^T\mathbf{C}^T, \mathbf{F_S}\mathbf{K}^k - \mathbf{K}^k\mathbf{B}^T\mathbf{K}^k, \ldots]\\
&= \mathrm{sp}[\mathbf{C}^T, \mathbf{K}^k, \mathbf{F_S}\mathbf{C}^T, \mathbf{F_S}\mathbf{K}^k, \ldots]\\
&= \mathcal{K}_m(\mathbf{F_S}, [\mathbf{C}^T\,\mathbf{K}^k]) = \mathcal{K}_m(\mathbf{F_S}, \mathbf{W}^k)
\end{aligned}$$

Problem Setting

Example: KÃ¡rmÃ¡n vortex street

- Create matrices with FEM software NAVIER.
- Discretization of the domain with conformal *Taylor-Hood elements*.
- P2-P1 elements fulfill the LBB condition.
  $\Rightarrow$ initial stable discretization

Level	n_v	n_p
1	3 452	453
2	8 726	1 123
3	20 512	2 615
4	45 718	5 783
5	99 652	12 566
6	211 452	26 572

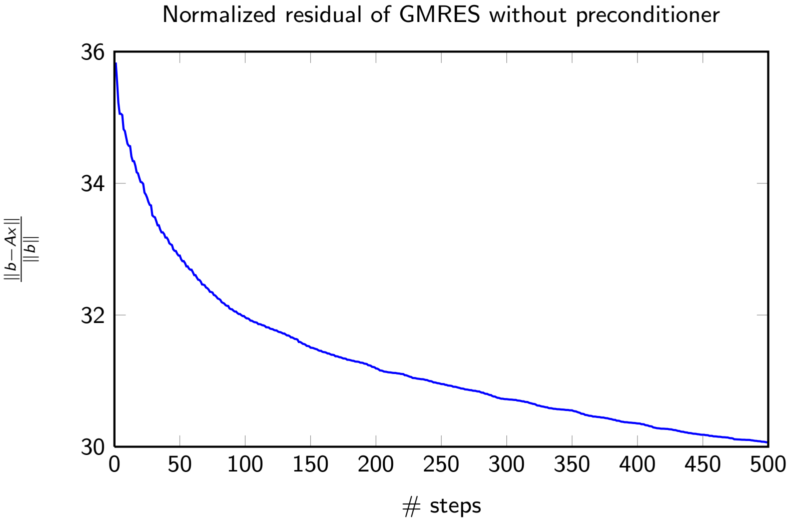
Tabelle: Levels of refinement

GMRES

Computation Comparison

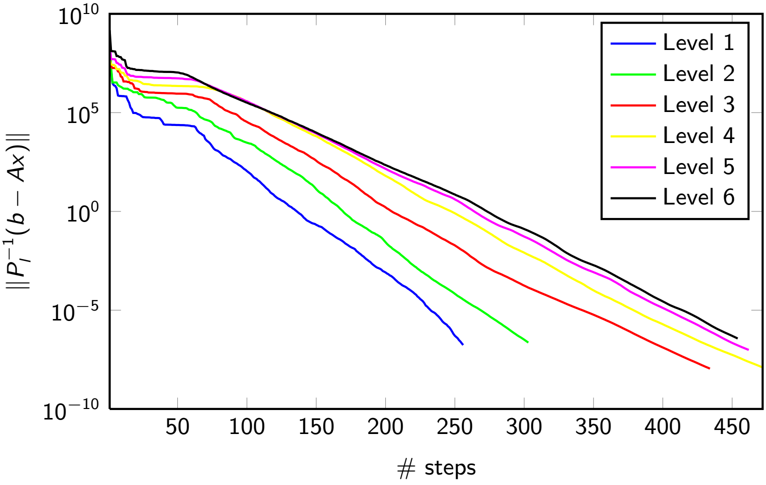
- GMRES with P_l as left preconditioner.
- Initial data: $b = rand(n_v + n_p, 1), x^{(0)} = zeros(n_v + n_p, 1)$
- Computations using MATLAB 2010b on `editha`:
  CPU type: Intel®Xeon®@2.67GHz,
  #CPUs: 4, #Cores: 32 (8 per CPU), RAM: 1 TB

GMRES


Normalized residual of GMRES without preconditioner
36
34
32
30
0
50
100
150
200
250
300
350
400
450
500
$\frac{\|b-Ax\|}{\|b\|}$
steps

GMRES

Preconditioned residuals in GMRES



GMRES

Computation Comparison

- GMRES with P_l as left preconditioner.
- Initial data: $b = rand(n_v + n_p, 1), x^{(0)} = zeros(n_v + n_p, 1)$
- Computations using MATLAB 2010b on `editha`:
  CPU type: Intel®Xeon®@2.67GHz,
  #CPUs: 4, #Cores: 32 (8 per CPU), RAM: 1 TB

Level	# it	$\frac{\|\|P_l^{-1}(b-Ax^{end})\|\|}{1}$	$\frac{\|\|b-Ax^{end}\|\|}{\|\|b\|\|}$	$\frac{res_{end}}{res_0}$
1	256	$8.6 \cdot 10^{-9}$	$7.5 \cdot 10^{-6}$	$2.8 \cdot 10^{-16}$
2	303	$2.4 \cdot 10^{-11}$	$9.5 \cdot 10^{-6}$	$2.7 \cdot 10^{-16}$
3	434	$5.3 \cdot 10^{-10}$	$4.3 \cdot 10^{-6}$	$2.4 \cdot 10^{-16}$
4	472	$9.9 \cdot 10^{-10}$	$8.8 \cdot 10^{-6}$	$2.5 \cdot 10^{-16}$
5	462	$4.5 \cdot 10^{-10}$	$1.5 \cdot 10^{-5}$	$2.4 \cdot 10^{-16}$
6	454	$3.5 \cdot 10^{-10}$	$1.6 \cdot 10^{-5}$	$2.5 \cdot 10^{-16}$
no P_l	500	$8.1 \cdot 10^{-1}$	$8.1 \cdot 10^{-1}$	$8.4 \cdot 10^{-1}$

GMRES

Computation Comparison

- GMRES with P_l as left preconditioner.
- Initial data: $b = rand(n_v + n_p, 1), x^{(0)} = zeros(n_v + n_p, 1)$
- Computations using MATLAB 2010b on `editha`:
  CPU type: Intel®Xeon®@2.67GHz,
  #CPUs: 4, #Cores: 32 (8 per CPU), RAM: 1 TB

Level	# it	$\frac{\|\|P_l^{-1}(b-Ax^{end})\|\|}{1}$	$\frac{\|\|b-Ax^{end}\|\|}{\|\|b\|\|}$	$\frac{res_{end}}{res_0}$
1	256	$8.6 \cdot 10^{-9}$	$7.5 \cdot 10^{-6}$	$2.8 \cdot 10^{-16}$
2	303	$2.4 \cdot 10^{-11}$	$9.5 \cdot 10^{-6}$	$2.7 \cdot 10^{-16}$
3	434	$5.3 \cdot 10^{-10}$	$4.3 \cdot 10^{-6}$	$2.4 \cdot 10^{-16}$
4	472	$9.9 \cdot 10^{-10}$	$8.8 \cdot 10^{-6}$	$2.5 \cdot 10^{-16}$
5	462	$4.5 \cdot 10^{-10}$	$1.5 \cdot 10^{-5}$	$2.4 \cdot 10^{-16}$
6	454	$3.5 \cdot 10^{-10}$	$1.6 \cdot 10^{-5}$	$2.5 \cdot 10^{-16}$
no P_l	500	$8.1 \cdot 10^{-1}$	$8.1 \cdot 10^{-1}$	eps $= 2.22 \cdot 10^{-16}$

GMRES

Computation Comparison

- GMRES with P_l as left preconditioner.
- Initial data: $b = rand(n_v + n_p, 1), x^{(0)} = zeros(n_v + n_p, 1)$
- Computations using MATLAB 2010b on `editha`:
  CPU type: Intel®Xeon®@2.67GHz,
  #CPUs: 4, #Cores: 32 (8 per CPU), RAM: 1 TB

Many thanks for your attention.

Level	# it	$\lVert P_l^{-1}(b - Ax^{end})\rVert$	$\frac{\lVert b - Ax^{end}\rVert}{\lVert b\rVert}$	$\frac{res_{end}}{res_0}$
1	256	$8.6 \cdot 10^{[illegible]}$	$7.5 \cdot 10^{-6}$	$2.8 \cdot 10^{-16}$
2	303	$2.4 \cdot 10^{-11}$	$9.5 \cdot 10^{-6}$	$2.7 \cdot 10^{-16}$
3	434	$5.3 \cdot 10^{-10}$	$4.3 \cdot 10^{-6}$	$2.4 \cdot 10^{-16}$
4	472	$9.9 \cdot 10^{-10}$	$8.8 \cdot 10^{-6}$	$2.5 \cdot 10^{-16}$
5	462	$4.5 \cdot 10^{-10}$	$1.5 \cdot 10^{-5}$	$2.4 \cdot 10^{-16}$
6	454	$3.5 \cdot 10^{-10}$	$1.6 \cdot 10^{-5}$	$2.5 \cdot 10^{-16}$
no P_l	500	$8.1 \cdot 10^{-1}$	$8.1 \cdot 10^{-1}$	$8.4 \cdot 10^{-1}$

References

1. E. BÃ¤nsch and P. Benner
Stabilization of Incompressible Flow Problems by Riccati-Based Feedback
G. Leugering, et al (eds.), *Constrained Optimization and Optimal Control for Partial Differential Equations*, International Series of Numerical Mathematics, Vol. 160, BirkhÃ¤user, Basel, 2011.

2. P. Benner and J. Saak
A Galerkin-Newton-ADI Method for Solving Large-Scale Algebraic Riccati Equations,
Preprint SPP1253-090 (January 2010)

3. P. Benner, J.-R. Li, and T. Penzl.
Numerical solution of large Lyapunov equations, Riccati equations, and linear-quadratic control problems.
Numer. Lin. Alg. Appl., vol. 15, no. 9, pp. 755–777, 2008.

4. H. Elman, D. Silvester and A. Wathen.
Finite Elements and fast Iterative Solvers with applications in incompressible fluid dynamics.
Oxford Science Publications, 2005, ISBN: 0-19-852868-X.