MAX PLANCK INSTITUTE
FOR DYNAMICS OF COMPLEX
TECHNICAL SYSTEMS
MAGDEBURG

CSC
COMPUTATIONAL METHODS IN
SYSTEMS AND CONTROL THEORY

Riccati-Based Feedback Control of Nonlinear Unsteady PDEs

Eberhard Bänsch (FAU Erlangen)
Peter Benner, Jens Saak (MPI/CSC)
Matthias Heinkenschloss (Rice U, Houston)
Heiko Weichelt (The MathWorks, Inc., Cambridge, UK)

International Workshop on Optimal Control of Dynamical Systems and Applications
Osijek, Croatia
June 20–22, 2018

Optimal Control

is used for the optimization of **dynamical processes**,

described by ordinary or partial differential equations.

This is achieved by minimizing a **cost functional**

(penalizing, e.g. energy consumption, deviation from reference trajectory),

such that a prescribed target
is reached in given or **minimal time**

whilst complying with given control and state constraints.

Let (x_*, u_*) solve $\min_{u \in \mathcal{U}_{ad}} J(x,u)$ **s.t.** $\dot{x}(t) = f(x(t), u(t))$.

Let (x_*, u_*) solve $\min_{u \in \mathcal{U}_{ad}} J(x,u)$ **s.t.** $\dot{x}(t) = f(x(t), u(t))$.

Fundamental observation

Optimized trajectory $x_*(t; u_*)$ and precomputed optimal control $u_*(t)$ will not be attainable in practice due to

- modeling errors and/or unmodeled dynamics,
- model uncertainties,
- external perturbations,
- measurement errors.

Let (x_*, u_*) solve $\min_{u \in \mathcal{U}_{ad}} J(x,u)$ **s.t.** $\dot{x}(t) = f(x(t), u(t))$.

Fundamental observation

Optimized trajectory $x_*(t; u_*)$ and precomputed optimal control $u_*(t)$ will not be attainable in practice due to

- modeling errors and/or unmodeled dynamics,
- model uncertainties,
- external perturbations,
- measurement errors.

Consequence: need **feedback control**

$$u(t) = u_*(t) + U(t, x(t) - x_*(t))$$

in order to attenuate perturbations/errors!

Example: Optimal control of a simple transport model

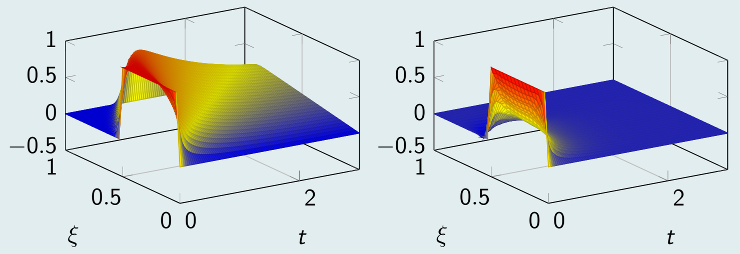
Burgers' equation:

$$\begin{aligned}
\partial_t x(t,\xi) &= \nu\,\partial_{\xi\xi} x(t,\xi) - x(t,\xi)\,\partial_\xi x(t,\xi) + B(\xi)u(t),\\
x(t,0) &= x(t,1) = 0, \quad x(0,\xi) = x_0(\xi), \quad \xi \in (0,1),\\
y(t,\xi) &= C\,x(t,\xi).
\end{aligned}$$

Example: Optimal control of a simple transport model

Burgers' equation:

$$
\begin{aligned}
\partial_t x(t,\xi) &= \nu\,\partial_{\xi\xi} x(t,\xi) - x(t,\xi)\,\partial_\xi x(t,\xi) + B(\xi)u(t),\\
x(t,0) &= x(t,1) = 0, \quad x(0,\xi) = x_0(\xi), \quad \xi \in (0,1),\\
y(t,\xi) &= C\,x(t,\xi).
\end{aligned}
$$

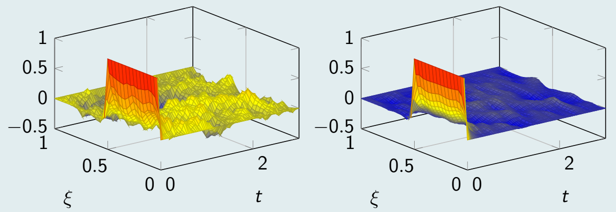


Example: Optimal control of a simple transport model

Burgers' equation:

$$\begin{aligned}\partial_t x(t,\xi) &= \nu\,\partial_{\xi\xi}x(t,\xi) - x(t,\xi)\,\partial_\xi x(t,\xi) + B(\xi)u(t) + F(\xi)v(t),\\ x(t,0) &= x(t,1) = 0, \quad x(0,\xi) = x_0(\xi) + \eta(\xi), \quad \xi \in (0,1),\\ y(t,\xi) &= C\,x(t,\xi) + w(t,\xi).\end{aligned}$$

Nonlinear control (here: MPC-LQG):



Reduction of tracking error $\int_0^T \|x(t) - x_*(t)\|_2^2\,dt$ by factor > 10.

[Benner/Görner, PAMM 2006]; [Benner/Görner/Saak, Springer LNCSE 2006].

The Linear-Quadratic Regulator (LQR) Problem

Minimize $\mathcal{J}(u) = \frac{1}{2}\int\limits_0^\infty \left(y^T Q y + u^T R u\right) dt$ for $u \in \mathcal{L}_2(0,\infty;\mathbb{R}^m)$,

subject to

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \qquad x(0) = x_0, \\ y(t) &= Cx(t), \end{aligned}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$.

The Linear-Quadratic Regulator (LQR) Problem

Minimize $\mathcal{J}(u) = \frac{1}{2}\int_0^\infty \left(y^T Q y + u^T R u\right) dt$ for $u \in \mathcal{L}_2(0,\infty;\mathbb{R}^m)$,

subject to

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \qquad x(0) = x_0, \\ y(t) &= Cx(t), \end{aligned}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{p\times n}$.

Solution of finite-dimensional LQR problem: **feedback control**

$$u_*(t) = -B^T X_* x(t) =: -K_* x(t),$$

where $X_* = X_*^T \geq 0$ is unique **stabilizing**[1] solution of **algebraic Riccati equation (ARE)**

$$0 = \mathcal{R}(X) := C^T QC + A^T X + XA - XBR^{-1}B^T X.$$

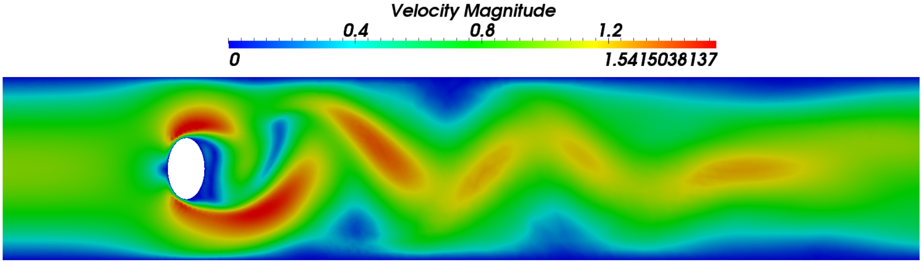
[1] X is stabilizing $\Leftrightarrow \Lambda\left(A - BB^T X\right) \subset \mathbb{C}^-$.

- **Physical transport** is one of the most fundamental dynamical processes in nature.
- **Prediction** and **manipulation** of transport processes are important research topics, e.g., to
  - avoid stall — for stable and safe flight;
  - save energy (or increase attainable speed) by minimizing drag coefficient;
  - use fluid flow for optimal transport (e.g., in blood veins).
- **Open-loop** controllers are widely used in various engineering fields.
  $\rightarrow$ **Not robust** regarding perturbation
- Dynamical systems are often influenced via so called **distributed control**.
  $\rightarrow$ **Unfeasible** in many real-world areas

$\Rightarrow$ **Boundary feedback stabilization (closed-loop)**
should be used to increase robustness and feasibility.

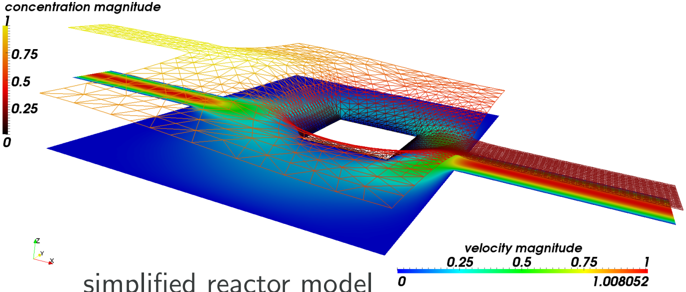
Overview

- Consider 2D flow problems described by **incompressible Navier–Stokes equations**.
- Riccati feedback approach requires the solution of an **algebraic Riccati equation**.
- Conservation of mass introduces a **divergence-freeness** condition $\rightsquigarrow$ problems with mathematical basis of control design schemes.



Kármán vortex street

- Consider 2D flow problems described by **incompressible Navier–Stokes equations**.
- Riccati feedback approach requires the solution of an **algebraic Riccati equation**.
- Conservation of mass introduces a **divergence-freeness** condition $\rightsquigarrow$ problems with mathematical basis of control design schemes.
- **Coupling** flow problems with a **scalar reaction-advection-diffusion equation**.



simplified reactor model

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.

[1] This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.

   Establish a numerical realization for Leray projection.

[1] This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.

   Establish a numerical realization for Leray projection.

2. NAVIER: FE package using $\mathcal{P}_2$-$\mathcal{P}_1$ Taylor–Hood elements.

[1] This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.

   Establish a numerical realization for Leray projection.

2. NAVIER: FE package using $\mathcal{P}_2$-$\mathcal{P}_1$ Taylor–Hood elements.

   Incorporate unsteady boundary conditions and boundary control operator.

[1] This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.

   Establish a numerical realization for Leray projection.

2. NAVIER: FE package using $\mathcal{P}_2$-$\mathcal{P}_1$ Taylor–Hood elements.

   Incorporate unsteady boundary conditions and boundary control operator.

3. LQR theory for generalized state-space systems.

[1] This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.

   Establish a numerical realization for Leray projection.

2. NAVIER: FE package using $\mathcal{P}_2$-$\mathcal{P}_1$ Taylor–Hood elements.

   Incorporate unsteady boundary conditions and boundary control operator.

3. LQR theory for generalized state-space systems.

   Incorporate a DAE structure without using expensive DAE methods.

[1] This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

— Available Tools and Necessary Tasks at Project Start[1] —

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.

   Establish a numerical realization for Leray projection.

2. NAVIER: FE package using $\mathcal{P}_2$-$\mathcal{P}_1$ Taylor–Hood elements.

   Incorporate unsteady boundary conditions and boundary control operator.

3. LQR theory for generalized state-space systems.

   Incorporate a DAE structure without using expensive DAE methods.

4. Kleinman–Newton-ADI framework for solving generalized algebraic Riccati equations.

[1] This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.

   Establish a numerical realization for Leray projection.

2. NAVIER: FE package using $\mathcal{P}_2$-$\mathcal{P}_1$ Taylor–Hood elements.

   Incorporate unsteady boundary conditions and boundary control operator.

3. LQR theory for generalized state-space systems.

   Incorporate a DAE structure without using expensive DAE methods.

4. Kleinman–Newton-ADI framework for solving generalized algebraic Riccati equations.

   Incorporate the divergence-free condition without explicit projection.

[1] This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

— Available Tools and Necessary Tasks at Project Start[1] —

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.

   Establish a numerical realization for Leray projection.

2. NAVIER: FE package using $\mathcal{P}_2$-$\mathcal{P}_1$ Taylor–Hood elements.

   Incorporate unsteady boundary conditions and boundary control operator.

3. LQR theory for generalized state-space systems.

   Incorporate a DAE structure without using expensive DAE methods.

4. Kleinman–Newton-ADI framework for solving generalized algebraic Riccati equations.

   Incorporate the divergence-free condition without explicit projection.

5. Preconditioned iterative methods to solve stationary Navier–Stokes systems.

[1] This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.

   Establish a numerical realization for Leray projection.

2. NAVIER: FE package using $\mathcal{P}_2$-$\mathcal{P}_1$ Taylor–Hood elements.

   Incorporate unsteady boundary conditions and boundary control operator.

3. LQR theory for generalized state-space systems.

   Incorporate a DAE structure without using expensive DAE methods.

4. Kleinman–Newton-ADI framework for solving generalized algebraic Riccati equations.

   Incorporate the divergence-free condition without explicit projection.

5. Preconditioned iterative methods to solve stationary Navier–Stokes systems.

   Develop techniques to deal with complex-shifted multi-field flow systems.

[1] This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

— Available Tools and Necessary Tasks at Project Start[1] —

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.

2. NAVIER: FE package using $\mathcal{P}_2$-$\mathcal{P}_1$ Taylor–Hood elements.

3. LQR theory for generalized state-space systems.

4. Kleinman–Newton-ADI framework for solving generalized algebraic Riccati equations.

5. Preconditioned iterative methods to solve stationary Navier–Stokes systems.

[1]This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

— Available Tools and Necessary Tasks at Project Start[1] —

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.

   - use discrete projector from [Heinkenschloss/Sorensen/Sun '08]
   - implicitly project on "hidden manifold"

   ⇒ Nested iteration: solve large-scale sparse saddle point system

2. NAVIER: FE package using $\mathcal{P}_2$-$\mathcal{P}_1$ Taylor–Hood elements.
3. LQR theory for generalized state-space systems.
4. Kleinman–Newton-ADI framework for solving generalized algebraic Riccati equations.
5. Preconditioned iterative methods to solve stationary Navier–Stokes systems.

[1] This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.
   - use discrete projector from [Heinkenschloss/Sorensen/Sun '08]
   - implicitly project on "hidden manifold"
   - ⇒ Nested iteration: solve large-scale sparse saddle point system
2. NAVIER: FE package using $\mathcal{P}_2$-$\mathcal{P}_1$ Taylor–Hood elements.
   - adapt various ideas from [Elman/Silvester/Wathen '05]
   - ⇒ develop suitable preconditioner to be used with GMRES
   - ⇒ efficient preconditioner uses various approximation methods
3. LQR theory for generalized state-space systems.
4. Kleinman–Newton-ADI framework for solving generalized algebraic Riccati equations.
5. Preconditioned iterative methods to solve stationary Navier–Stokes systems.

[1] This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.
2. NAVIER: FE package using $\mathcal{P}_2$-$\mathcal{P}_1$ Taylor–Hood elements.
3. LQR theory for generalized state-space systems.
4. Kleinman–Newton-ADI framework for solving generalized algebraic Riccati equations.
5. Preconditioned iterative methods to solve stationary Navier-Stokes systems.

- use discrete projector from [Heinkenschloss/Sorensen/Sun '08]
- implicitly project on "hidden manifold"

⇒ Nested iteration: solve large-scale sparse saddle point system

- adapt various ideas from [Elman/Silvester/Wathen '05]

⇒ develop suitable preconditioner to be used with GMRES

⇒ efficient preconditioner uses various approximation methods

- combine [Kürschner '16], [B./Byers '98], and [Feizinger/Hylla/Sachs '09]

⇒ extended ideas in [B./Heinkenschloss/Saak/Weichelt '16/'18]

⇒ develop a highly compatible method to solve Riccati equations

[1] This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.

- use discrete projector from [Heinkenschloss/Sorensen/Sun '08]
- implicitly project on "hidden manifold"

⇒ Nested iteration: solve large-scale sparse saddle point system

2. NAVIER: FE package using $\mathcal{P}_2$-$\mathcal{P}_1$ Taylor–Hood elements

- adapt various ideas from [Elman/Silvester/Wathen '05]

⇒ develop suitable preconditioner to be used with GMRES

⇒ efficient preconditioner uses various approximation methods

3. LQR theory for generalized state-space systems.

4. Kleinman–Newton-ADI framework for solving generalized algebraic Riccati equations.

- combine [Kürschner '16], [B./Byers '98], and [Feizinger/Hylla/Sachs '09]

⇒ extended ideas in [B./Heinkenschloss/Saak/Weichelt '16/'18]

⇒ develop a highly compatible method to solve Riccati equations

5. Preconditioned iterative methods to solve stationary Navier-Stokes systems.

- include feedback into forward simulation within NAVIER

⇒ closed-loop forward flow simulation

[1] This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

1. Functional analytic control approach by Raymond ([Raymond '05–'07]) works in subspace of divergence-free functions.

- use discrete projector from [Heinkenschloss/Sorensen/Sun '08]
- implicitly project on "hidden manifold"

⇒ Nested iteration: solve large-scale sparse saddle point system

2. NAVIER: FE package using $\mathcal{P}_2$-$\mathcal{P}_1$ Taylor–Hood elements.
3. LQR theory for generalized state-space systems.
4. Kleinman–Newton-ADI framework for solving generalized algebraic Riccati equations.

- combine [Kürschner '16], [B./Byers '98], and [Feizinger/Hylla/Sachs '09]

⇒ extended ideas in [B./Heinkenschloss/Saak/Weichelt '16/'18]

⇒ develop a highly compatible method to solve Riccati equations

5. Preconditioned iterative methods to solve stationary Navier-Stokes systems.

[1] This work started as part of the DFG SPP1253 "Optimization with PDEs" (2007–2013).

Navier–Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \frac{1}{\text{Re}} \Delta \vec{v} + (\vec{v} \cdot \nabla) \vec{v} + \nabla p = \vec{f}$$

$$\operatorname{div} \vec{v} = 0$$

- defined for time $t \in (0, \infty)$ and space $\vec{x} \in \Omega \subset \mathbb{R}^2$ bounded with $\Gamma = \partial\Omega$
- $+$ boundary and initial conditions
- initial boundary value problem with additional algebraic constraints

Navier–Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \frac{1}{\mathrm{Re}}\Delta \vec{v} + (\vec{v}\cdot\nabla)\vec{v} + \nabla p = \vec{f}$$

$$\operatorname{div} \vec{v} = 0$$

Linearize + Discretize → Index-2 DAE

$$M\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{v}(t) = A\mathbf{v}(t) + \widehat{G}\mathbf{p}(t) + B\mathbf{u}(t)$$

$$0 = \widehat{G}^T\mathbf{v}(t)$$

$$\mathbf{y}(t) = C\mathbf{v}(t)$$

$$A,\ M \in \mathbb{R}^{n\times n},\ \widehat{G} \in \mathbb{R}^{n\times n_{\mathbf{p}}}$$

$$B \in \mathbb{R}^{n\times n_r},\ C \in \mathbb{R}^{n_a\times n}$$

$$\mathbf{u}(t) \in \mathbb{R}^{n_r},\ \mathbf{y}(t) \in \mathbb{R}^{n_a}$$

$$\operatorname{rank}\left(\widehat{G}\right) = n_{\mathbf{p}}$$

$$M = M^T \succ 0$$

$$\mathbf{v}(t) \in \mathbb{R}^{n},\ \mathbf{p}(t) \in \mathbb{R}^{n_{\mathbf{p}}}$$

$$n = n_{\mathbf{v}},\ N = n + n_{\mathbf{p}}$$

Navier–Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \frac{1}{\mathrm{Re}}\Delta \vec{v} + (\vec{v}\cdot\nabla)\vec{v} + \nabla p = \vec{f}$$

$$\operatorname{div} \vec{v} = 0$$

$A,\ M \in \mathbb{R}^{n\times n},\ \widehat{G} \in \mathbb{R}^{n\times n_{\mathbf{p}}}$

$B \in \mathbb{R}^{n\times n_r},\ C \in \mathbb{R}^{n_a\times n}$

$\mathbf{u}(t) \in \mathbb{R}^{n_r},\ \mathbf{y}(t) \in \mathbb{R}^{n_a}$

$\operatorname{rank}\left(\widehat{G}\right) = n_{\mathbf{p}}$

Linearize + Discretize → Index-2 DAE

$$M\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{v}(t) = A\mathbf{v}(t) + \widehat{G}\mathbf{p}(t) + B\mathbf{u}(t)$$

$$0 = \widehat{G}^T\mathbf{v}(t)$$

$$\mathbf{y}(t) = C\mathbf{v}(t)$$

$M = M^T \succ 0$

$\mathbf{v}(t) \in \mathbb{R}^{n},\ \mathbf{p}(t) \in \mathbb{R}^{n_{\mathbf{p}}}$

$n = n_{\mathbf{v}},\ N = n + n_{\mathbf{p}}$

Show that projection in [Hei/Sor/Sun '08] is discretized version of Leray projector in [Ray '06].

$$M\Pi^T = \Pi M \quad \wedge \quad \Pi^T\mathbf{v} = \mathbf{v}_{\mathrm{div},0}$$

[**Bänsch**/B./Saak/**Stoll**/Weichelt **'13**,**'15**]

Feedback Stabilization for Index-2 DAE Systems
— Physics of Multi-Field Flow —

Navier–Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \frac{1}{\mathrm{Re}}\Delta \vec{v} + (\vec{v}\cdot\nabla)\vec{v} + \nabla p = \vec{f}$$

$$\operatorname{div} \vec{v} = 0$$

Concentration Equation

$$\frac{\partial c^{(\vec{v})}}{\partial t} - \frac{1}{\mathrm{Re}\,\mathrm{Sc}}\Delta c^{(\vec{v})} + (\vec{v}\cdot\nabla)c^{(\vec{v})} = 0$$

$$A,\ M \in \mathbb{R}^{n\times n},\ \widehat{G} \in \mathbb{R}^{n\times n_{\mathbf{p}}},$$
$$B \in \mathbb{R}^{n\times n_r},\ C \in \mathbb{R}^{n_a\times n}$$
$$\mathbf{u}(t) \in \mathbb{R}^{n_r},\ \mathbf{y}(t) \in \mathbb{R}^{n_a}$$
$$\operatorname{rank}\left(\widehat{G}\right) = n_{\mathbf{p}}$$

Linearize + Discretize → Index-2 DAE

$$M\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{v}(t) = A\mathbf{v}(t) + \widehat{G}\mathbf{p}(t) + B\mathbf{u}(t)$$
$$0 = \widehat{G}^T\mathbf{v}(t)$$
$$\mathbf{y}(t) = C\mathbf{v}(t)$$

$$M = M^T \succ 0$$
$$\mathbf{v}(t) \in \mathbb{R}^{n},\ \mathbf{p}(t) \in \mathbb{R}^{n_{\mathbf{p}}}$$
$$n = n_{\mathbf{v}},\ N = n + n_{\mathbf{p}}$$

Show that projection in [Hei/Sor/Sun '08] is discretized version of Leray projector in [Ray '06].

$$M\Pi^T = \Pi M \quad \wedge \quad \Pi^T\mathbf{v} = \mathbf{v}_{\mathrm{div},0}$$

[**Bänsch**/B./Saak/**Stoll**/Weichelt **'13**,**'15**]

Navier–Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \frac{1}{\mathrm{Re}}\Delta\vec{v} + (\vec{v}\cdot\nabla)\vec{v} + \nabla p = \vec{f}$$

$$\operatorname{div}\vec{v} = 0$$

Concentration Equation

$$\frac{\partial c^{(\vec{v})}}{\partial t} - \frac{1}{\mathrm{Re}\,\mathrm{Sc}}\Delta c^{(\vec{v})} + (\vec{v}\cdot\nabla)c^{(\vec{v})} = 0$$

$$A,\ M \in \mathbb{R}^{n\times n},\ \widehat{G} \in \mathbb{R}^{n\times n_{\mathbf{p}}}$$

$$B \in \mathbb{R}^{n\times n_r},\ C \in \mathbb{R}^{n_a\times n}$$

$$\mathbf{u}(t) \in \mathbb{R}^{n_r},\ \mathbf{y}(t) \in \mathbb{R}^{n_a}$$

$$\operatorname{rank}\left(\widehat{G}\right) = n_{\mathbf{p}}$$

Linearize + Discretize → Index-2 DAE

$$M\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = A\mathbf{x}(t) + \widehat{G}\mathbf{p}(t) + B\mathbf{u}(t)$$

$$0 = \widehat{G}^T\mathbf{x}(t)$$

$$\mathbf{y}(t) = C\mathbf{x}(t)$$

$$M = M^T \succ 0$$

$$\mathbf{x}(t) = \begin{bmatrix}\mathbf{v}(t)\\ \mathbf{c}(t)\end{bmatrix} \in \mathbb{R}^n$$

$$n = n_{\mathbf{v}} + n_{\mathbf{c}},\ N = n + n_{\mathbf{p}}$$

Show that projection in [Hei/Sor/Sun '08] is discretized version of Leray projector in [Ray '06].

$$M\Pi^T = \Pi M \quad \wedge \quad \Pi^T\mathbf{v} = \mathbf{v}_{\mathrm{div},0}$$

[**Bänsch**/B./Saak/**Stoll**/Weichelt **'13**,**'15**]

Navier–Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \frac{1}{\mathrm{Re}}\Delta\vec{v} + (\vec{v}\cdot\nabla)\vec{v} + \nabla p = \vec{f}$$

$$\operatorname{div}\vec{v} = 0$$

Concentration Equation

$$\frac{\partial c^{(\vec{v})}}{\partial t} - \frac{1}{\mathrm{Re}\,\mathrm{Sc}}\Delta c^{(\vec{v})} + (\vec{v}\cdot\nabla)c^{(\vec{v})} = 0$$

$$A,\ M \in \mathbb{R}^{n\times n},\ \widehat{G} \in \mathbb{R}^{n\times n_{\mathbf{p}}},$$
$$B \in \mathbb{R}^{n\times n_r},\ C \in \mathbb{R}^{n_a\times n}$$
$$\mathbf{u}(t) \in \mathbb{R}^{n_r},\ \mathbf{y}(t) \in \mathbb{R}^{n_a}$$
$$\operatorname{rank}\left(\widehat{G}\right) = n_{\mathbf{p}}$$

Linearize + Discretize → Index-2 DAE

$$M\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = A\mathbf{x}(t) + \widehat{G}\mathbf{p}(t) + B\mathbf{u}(t)$$
$$0 = \widehat{G}^T\mathbf{x}(t)$$
$$\mathbf{y}(t) = C\mathbf{x}(t)$$

$$M = M^T \succ 0$$
$$\mathbf{x}(t) = \begin{bmatrix}\mathbf{v}(t)\\ \mathbf{c}(t)\end{bmatrix} \in \mathbb{R}^n$$
$$n = n_{\mathbf{v}} + n_{\mathbf{c}},\ N = n + n_{\mathbf{p}}$$

Show that projection in [Hei/Sor/Sun '08] is discretized version of Leray projector in [Ray '06].

$$M\Pi^T = \Pi M \quad \wedge \quad \Pi^T\mathbf{v} = \mathbf{v}_{\mathrm{div},0}$$

[**Bänsch**/B./Saak/**Stoll**/Weichelt **'13**,**'15**]

Extension to coupled flow case, i.e.,

$$\hat{\Pi} := \begin{bmatrix}\Pi & 0\\ 0 & I\end{bmatrix} \quad \wedge \quad \begin{bmatrix}\Pi^T & 0\\ 0 & I\end{bmatrix}\begin{bmatrix}\mathbf{v}\\ \mathbf{c}\end{bmatrix} = \begin{bmatrix}\mathbf{v}_{\mathrm{div},0}\\ \mathbf{c}\end{bmatrix}.$$

[Bänsch/B./Saak/Weichelt '14]

Feedback Stabilization for Index-2 DAE Systems
— Physics of Multi-Field Flow —

Navier–Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \frac{1}{\mathrm{Re}}\Delta \vec{v} + (\vec{v}\cdot\nabla)\vec{v} + \nabla p = \vec{f}$$

$$\operatorname{div} \vec{v} = 0$$

Concentration Equation

$$\frac{\partial c^{(\vec{v})}}{\partial t} - \frac{1}{\mathrm{Re}\,\mathrm{Sc}}\Delta c^{(\vec{v})} + (\vec{v}\cdot\nabla)c^{(\vec{v})} = 0$$

Linearize + Discretize → Index-2 DAE

$$M\frac{\mathrm{d}}{\mathrm{d}t}\mathbf{x}(t) = A\mathbf{x}(t) + \widehat{G}\mathbf{p}(t) + B\mathbf{u}(t)$$
$$0 = \widehat{G}^T\mathbf{x}(t)$$
$$\mathbf{y}(t) = C\mathbf{x}(t)$$

Show that projection in [Hei/Sor/Sun '08] is discretized version of Leray projector in [Ray '06].

$$M\Pi^T = \Pi M \quad \wedge \quad \Pi^T\mathbf{v} = \mathbf{v}_{\mathrm{div},0}$$

[**Bänsch**/B./Saak/**Stoll**/Weichelt **'13**,**'15**]

Extension to coupled flow case, i.e.,

$$\widehat{\Pi} := \begin{bmatrix} \Pi & 0 \\ 0 & I \end{bmatrix} \quad \wedge \quad \begin{bmatrix} \Pi^T & 0 \\ 0 & I \end{bmatrix}\begin{bmatrix} \mathbf{v} \\ \mathbf{c} \end{bmatrix} = \begin{bmatrix} \mathbf{v}_{\mathrm{div},0} \\ \mathbf{c} \end{bmatrix}.$$

[Bänsch/B./Saak/Weichelt '14]

Helmholtz Decomposition [Girault/Raviart '86]

- Splitting:

$$(L^2(\Omega))^2 = \mathbf{H}(\mathrm{div}, 0) \perp \mathbf{H}(\mathrm{div}, 0)^{\perp}$$

Divergence-free: $\mathbf{H}(\mathrm{div}, 0) := \{\vec{v} \in (L^2(\Omega))^2 \mid \mathrm{div}\, \vec{v} = 0, \vec{v} \cdot \vec{n}_{|\Gamma} = 0\}$

Curl-free: $\mathbf{H}(\mathrm{div}, 0)^{\perp} = \{\nabla p \mid p \in H^1(\Omega)\}$

Helmholtz Decomposition [Girault/Raviart '86]

- Splitting:

$$(L^2(\Omega))^2 = \mathbf{H}(\mathrm{div}, 0) \perp \mathbf{H}(\mathrm{div}, 0)^\perp$$

Divergence-free: $\mathbf{H}(\mathrm{div}, 0) := \{\vec{v} \in (L^2(\Omega))^2 \mid \mathrm{div}\, \vec{v} = 0, \vec{v} \cdot \vec{n}_{|\Gamma} = 0\}$

Curl-free: $\mathbf{H}(\mathrm{div}, 0)^\perp = \{\nabla p \mid p \in H^1(\Omega)\}$

Leray Projector P

This splitting is equivalent to $\vec{v} = \vec{v}_{\mathrm{div},0} + \nabla p$, where $\vec{v}_{\mathrm{div},0}$ and p fulfill

$$\begin{aligned} \vec{v}_{\mathrm{div},0} + \nabla p &= \vec{v} \quad \text{in } \Omega, \\ \mathrm{div}\, \vec{v}_{\mathrm{div},0} &= 0 \quad \text{in } \Omega, \\ \vec{v}_{\mathrm{div},0} \cdot \vec{n} &= 0 \quad \text{on } \Gamma. \end{aligned}$$

$P : (L^2(\Omega))^2 \to \mathbf{H}(\mathrm{div}, 0)$ with $P : \vec{v} \mapsto \vec{v}_{\mathrm{div},0}$.

Projection Method

[Heinkenschloss/Sorensen/Sun '08]

- Index reduction for Lyapunov-solver.
- Projector:

$$\Pi^T := I_{n_v} - M_v^{-1} G (G^T M_v^{-1} G)^{-1} G^T.$$

Projection Method [Heinkenschloss/Sorensen/Sun '08]

- Index reduction for Lyapunov-solver.
- Projector:

$$\Pi^T := I_{n_v} - M_v^{-1} G (G^T M_v^{-1} G)^{-1} G^T.$$

Recall $P : \vec{v} \mapsto \vec{w}$:

$$\begin{aligned} \vec{w} + \nabla p &= \vec{v}, \\ \operatorname{div} \vec{w} &= 0 \end{aligned} \quad \Rightarrow \quad \begin{bmatrix} M_v & G \\ G^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} M_v \mathbf{v} \\ 0 \end{bmatrix}$$

$$\mathbf{p} = (G^T M_v^{-1} G)^{-1} G^T \mathbf{v}$$

$$\mathbf{w} = (I_{n_v} - M_v^{-1} G (G^T M_v^{-1} G)^{-1} G^T) \mathbf{v}$$

Projection Method [Heinkenschloss/Sorensen/Sun '08]

- Index reduction for Lyapunov-solver.
- Projector:

$$\Pi^T := I_{n_v} - M_v^{-1} G (G^T M_v^{-1} G)^{-1} G^T.$$

Recall $P : \vec{v} \mapsto \vec{w}$:

$$\begin{aligned} \vec{w} + \nabla p &= \vec{v}, \\ \operatorname{div} \vec{w} &= 0 \end{aligned} \quad \Rightarrow \quad \begin{bmatrix} M_v & G \\ G^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} M_v \mathbf{v} \\ 0 \end{bmatrix}$$

$$\mathbf{p} = (G^T M_v^{-1} G)^{-1} G^T \mathbf{v}$$

$$\mathbf{w} = (I_{n_v} - M_v^{-1} G (G^T M_v^{-1} G)^{-1} G^T) \mathbf{v}$$

Leray vs. Π^T

$$\begin{aligned} \vec{w} &= P(\vec{v}), \\ 0 &= \operatorname{div} \vec{w} \end{aligned} \quad \Rightarrow \quad \begin{aligned} \mathbf{w} &= \Pi^T \mathbf{v}, \\ 0 &= G^T \mathbf{w} \end{aligned}$$

Minimize

$$\mathcal{J}(\mathbf{y},\mathbf{u}) = \frac{1}{2}\int_0^\infty \lambda^2 ||\mathbf{y}||^2 + ||\mathbf{u}||^2 \,\mathrm{d}t$$

subject to

$$\widehat{\Theta}_r^T M \widehat{\Theta}_r \frac{\mathrm{d}}{\mathrm{d}t}\widetilde{\mathbf{x}}(t) = \widehat{\Theta}_r^T A \widehat{\Theta}_r \widetilde{\mathbf{x}}(t) + \widehat{\Theta}_r^T B\mathbf{u}(t)$$
$$\mathbf{y}(t) = C\widehat{\Theta}_r \widetilde{\mathbf{x}}(t)$$

with $\widehat{\Pi} = \widehat{\Theta}_l \widehat{\Theta}_r^T$ such that $\widehat{\Theta}_r^T \widehat{\Theta}_l = I \in \mathbb{R}^{(n-n_\mathbf{p})\times(n-n_\mathbf{p})}$ and $\widetilde{\mathbf{x}} = \widehat{\Theta}_l^T \mathbf{x}$.

Minimize

$$\mathcal{J}(\mathbf{y}, \mathbf{u}) = \frac{1}{2} \int_0^\infty \lambda^2 ||\mathbf{y}||^2 + ||\mathbf{u}||^2 \, \mathrm{d}t$$

subject to

$$\mathcal{M} \frac{\mathrm{d}}{\mathrm{d}t} \widetilde{\mathbf{x}}(t) = \mathcal{A} \widetilde{\mathbf{x}}(t) + \mathcal{B} \mathbf{u}(t)$$
$$\mathbf{y}(t) = \mathcal{C} \widetilde{\mathbf{x}}(t)$$

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

Minimize

$$\mathcal{J}(\mathbf{y},\mathbf{u}) = \frac{1}{2}\int_0^\infty \lambda^2||\mathbf{y}||^2 + ||\mathbf{u}||^2 \, \mathrm{d}t$$

subject to

$$\mathcal{M}\frac{\mathrm{d}}{\mathrm{d}t}\widetilde{\mathbf{x}}(t) = \mathcal{A}\widetilde{\mathbf{x}}(t) + \mathcal{B}\mathbf{u}(t)$$
$$\mathbf{y}(t) = \mathcal{C}\widetilde{\mathbf{x}}(t)$$

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

Riccati Based Feedback Approach

- Optimal control: $\mathbf{u}(t) = -\mathcal{K}\widetilde{\mathbf{x}}(t)$, with feedback: $\mathcal{K} = \mathcal{B}^T\mathcal{X}\mathcal{M}$,

  where $\mathcal{X}$ is the solution of the generalized continuous-time algebraic Riccati equation (GCARE)

$$\mathcal{R}(\mathcal{X}) = \lambda^2\mathcal{C}^T\mathcal{C} + \mathcal{A}^T\mathcal{X}\mathcal{M} + \mathcal{M}\mathcal{X}\mathcal{A} - \mathcal{M}\mathcal{X}\mathcal{B}\mathcal{B}^T\mathcal{X}\mathcal{M} = 0.$$

Determine $\mathcal{X} = \mathcal{X}^T \succeq 0$ such that $\mathcal{R}(\mathcal{X}) = \mathcal{C}^T\mathcal{C} + \mathcal{A}^T\mathcal{X}\mathcal{M} + \mathcal{M}\mathcal{X}\mathcal{A} - \mathcal{M}\mathcal{X}\mathcal{B}\mathcal{B}^T\mathcal{X}\mathcal{M} = 0.$

Determine $\mathcal{X} = \mathcal{X}^T \succeq 0$ such that $\mathcal{R}(\mathcal{X}) = \mathcal{C}^T\mathcal{C} + \mathcal{A}^T\mathcal{X}\mathcal{M} + \mathcal{M}\mathcal{X}\mathcal{A} - \mathcal{M}\mathcal{X}\mathcal{B}\mathcal{B}^T\mathcal{X}\mathcal{M} = 0.$

Kleinman–Newton method

Determine $\mathcal{X} = \mathcal{X}^T \succeq 0$ such that $\mathcal{R}(\mathcal{X}) = \mathcal{C}^T\mathcal{C} + \mathcal{A}^T\mathcal{X}\mathcal{M} + \mathcal{M}\mathcal{X}\mathcal{A} - \mathcal{M}\mathcal{X}\mathcal{B}\mathcal{B}^T\mathcal{X}\mathcal{M} = 0.$

Kleinman–Newton method

Step $\mathbf{m+1}$: Solve the Lyapunov equation

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)})^T\mathcal{X}^{(m+1)}\mathcal{M} + \mathcal{M}\mathcal{X}^{(m+1)}(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)}) = -(\mathcal{W}^{(m)})^T\mathcal{W}^{(m)} \quad (1)$$

Determine $\mathcal{X} = \mathcal{X}^T \succeq 0$ such that $\mathcal{R}(\mathcal{X}) = \mathcal{C}^T\mathcal{C} + \mathcal{A}^T\mathcal{X}\mathcal{M} + \mathcal{M}\mathcal{X}\mathcal{A} - \mathcal{M}\mathcal{X}\mathcal{B}\mathcal{B}^T\mathcal{X}\mathcal{M} = 0.$

Step m + 1: Solve the Lyapunov equation

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)})^T\mathcal{X}^{(m+1)}\mathcal{M} + \mathcal{M}\mathcal{X}^{(m+1)}(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)}) = -(\mathcal{W}^{(m)})^T\mathcal{W}^{(m)} \tag{1}$$

Kleinman–Newton method

low-rank ADI method

Determine $\mathcal{X} = \mathcal{X}^T \succeq 0$ such that $\mathcal{R}(\mathcal{X}) = \mathcal{C}^T\mathcal{C} + \mathcal{A}^T\mathcal{X}\mathcal{M} + \mathcal{M}\mathcal{X}\mathcal{A} - \mathcal{M}\mathcal{X}\mathcal{B}\mathcal{B}^T\mathcal{X}\mathcal{M} = 0.$

Kleinman–Newton method

Step $\mathbf{m+1}$: Solve the Lyapunov equation

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)})^T\mathcal{X}^{(m+1)}\mathcal{M} + \mathcal{M}\mathcal{X}^{(m+1)}(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)}) = -(\mathcal{W}^{(m)})^T\mathcal{W}^{(m)} \tag{1}$$

low-rank ADI method

Step ℓ: Solve the projected and shifted linear system

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)} + q_\ell\mathcal{M})^T\mathcal{V}_\ell = \mathcal{Y} \tag{2}$$

Determine $\mathcal{X} = \mathcal{X}^T \succeq 0$ such that $\mathcal{R}(\mathcal{X}) = \mathcal{C}^T\mathcal{C} + \mathcal{A}^T\mathcal{X}\mathcal{M} + \mathcal{M}\mathcal{X}\mathcal{A} - \mathcal{M}\mathcal{X}\mathcal{B}\mathcal{B}^T\mathcal{X}\mathcal{M} = 0.$

Kleinman–Newton method

Step $\mathbf{m+1}$: Solve the Lyapunov equation

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)})^T\mathcal{X}^{(m+1)}\mathcal{M} + \mathcal{M}\mathcal{X}^{(m+1)}(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)}) = -(\mathcal{W}^{(m)})^T\mathcal{W}^{(m)} \quad (1)$$

low-rank ADI method

Step ℓ: Solve the projected and shifted linear system

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)} + q_\ell\mathcal{M})^T\mathcal{V}_\ell = \mathcal{Y} \quad (2)$$

linear solver

Feedback Stabilization for Index-2 DAE Systems
— Nested Iteration without Projection —

Determine $\mathcal{X} = \mathcal{X}^T \succeq 0$ such that $\mathcal{R}(\mathcal{X}) = \mathcal{C}^T\mathcal{C} + \mathcal{A}^T\mathcal{X}\mathcal{M} + \mathcal{M}\mathcal{X}\mathcal{A} - \mathcal{M}\mathcal{X}\mathcal{B}\mathcal{B}^T\mathcal{X}\mathcal{M} = 0.$

Kleinman–Newton method

Step m + 1: Solve the Lyapunov equation

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)})^T\mathcal{X}^{(m+1)}\mathcal{M} + \mathcal{M}\mathcal{X}^{(m+1)}(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)}) = -(\mathcal{W}^{(m)})^T\mathcal{W}^{(m)} \qquad (1)$$

low-rank ADI method

Step ℓ: Solve the projected and shifted linear system

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)} + q_\ell\mathcal{M})^T\mathcal{V}_\ell = \mathcal{Y} \qquad (2)$$

linear solver

Avoid explicit projection using $\widehat{\Theta}_r\mathcal{V}_\ell = V_\ell$, $\mathcal{Y} = \widehat{\Theta}_r^T Y$, and [Hei/Sor/Sun '08]:

Determine $\mathcal{X} = \mathcal{X}^T \succeq 0$ such that $\mathcal{R}(\mathcal{X}) = \mathcal{C}^T\mathcal{C} + \mathcal{A}^T\mathcal{X}\mathcal{M} + \mathcal{M}\mathcal{X}\mathcal{A} - \mathcal{M}\mathcal{X}\mathcal{B}\mathcal{B}^T\mathcal{X}\mathcal{M} = 0$.

Kleinman–Newton method

Step $\mathbf{m+1}$: Solve the Lyapunov equation

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)})^T\mathcal{X}^{(m+1)}\mathcal{M} + \mathcal{M}\mathcal{X}^{(m+1)}(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)}) = -(\mathcal{W}^{(m)})^T\mathcal{W}^{(m)} \qquad (1)$$

low-rank ADI method

Step ℓ: Solve the projected and shifted linear system

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)} + q_\ell\mathcal{M})^T\mathcal{V}_\ell = \mathcal{Y} \qquad (2)$$

linear solver

Avoid explicit projection using $\widehat{\Theta}_r\mathcal{V}_\ell = V_\ell$, $\mathcal{Y} = \widehat{\Theta}_r^T Y$, and [Hei/Sor/Sun '08]:
Replace (2) and **solve instead** the saddle point system (SPS)

$$\begin{bmatrix} A^T - (K^{(m)})^T B^T + q_\ell M & \widehat{G} \\ \widehat{G}^T & 0 \end{bmatrix} \begin{bmatrix} V_\ell \\ * \end{bmatrix} = \begin{bmatrix} Y \\ 0 \end{bmatrix}$$

for different ADI shifts $q_\ell \in \mathbb{C}^-$ for a couple of rhs Y.

Feedback Stabilization for Index-2 DAE Systems — Nested Iteration without Projection —

Determine $\mathcal{X} = \mathcal{X}^T \succeq 0$ such that $\mathcal{R}(\mathcal{X}) = \mathcal{C}^T\mathcal{C} + \mathcal{A}^T\mathcal{X}\mathcal{M} + \mathcal{M}\mathcal{X}\mathcal{A} - \mathcal{M}\mathcal{X}\mathcal{B}\mathcal{B}^T\mathcal{X}\mathcal{M} = 0.$

Kleinman–Newton method

Step $\mathbf{m+1}$: Solve the Lyapunov equation

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)})^T\mathcal{X}^{(m+1)}\mathcal{M} + \mathcal{M}\mathcal{X}^{(m+1)}(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)}) = -(\mathcal{W}^{(m)})^T\mathcal{W}^{(m)} \qquad (1)$$

low-rank ADI method

Step ℓ: Solve the projected and shifted linear system

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)} + q_\ell\mathcal{M})^T\mathcal{V}_\ell = \mathcal{Y} \qquad (2)$$

linear solver

Avoid explicit projection using $\widehat{\Theta}_r\mathcal{V}_\ell = V_\ell$, $\mathcal{Y} = \widehat{\Theta}_r^T Y$, and [Hei/Sor/Sun '08]:
Replace (2) and **solve instead** the saddle point system (SPS)
(using *Sherman–Morrison–Woodbury* formula)

$$\begin{bmatrix} A^T - (K^{(m)})^T B^T + q_\ell M & \widehat{G} \\ \widehat{G}^T & 0 \end{bmatrix} \begin{bmatrix} V_\ell \\ * \end{bmatrix} = \begin{bmatrix} Y \\ 0 \end{bmatrix}$$

for different ADI shifts $q_\ell \in \mathbb{C}^-$ for a couple of rhs Y.

Determine $\mathcal{X} = \mathcal{X}^T \succeq 0$ such that $\mathcal{R}(\mathcal{X}) = \mathcal{C}^T\mathcal{C} + \mathcal{A}^T\mathcal{X}\mathcal{M} + \mathcal{M}\mathcal{X}\mathcal{A} - \mathcal{M}\mathcal{X}\mathcal{B}\mathcal{B}^T\mathcal{X}\mathcal{M} = 0$.

Kleinman–Newton method

Step $\mathbf{m+1}$: Solve the Lyapunov equation

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)})^T\mathcal{X}^{(m+1)}\mathcal{M} + \mathcal{M}\mathcal{X}^{(m+1)}(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)}) = -(\mathcal{W}^{(m)})^T\mathcal{W}^{(m)} \tag{1}$$

low-rank ADI method

Step ℓ: Solve the projected and shifted linear system

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)} + q_\ell\mathcal{M})^T\mathcal{V}_\ell = \mathcal{Y} \tag{2}$$

linear solver

Avoid explicit projection using $\widehat{\Theta}_r\mathcal{V}_\ell = V_\ell$, $\mathcal{Y} = \widehat{\Theta}_r^T Y$, and [HEI/SOR/SUN '08]:
Replace (2) and **solve instead** the saddle point system (SPS)
(using *Sherman–Morrison–Woodbury* formula)

$$\begin{bmatrix} A^T + q_\ell M & \widehat{G} \\ \widehat{G}^T & 0 \end{bmatrix} \begin{bmatrix} V_\ell \\ * \end{bmatrix} = \begin{bmatrix} \tilde{Y} \\ 0 \end{bmatrix}$$

for different ADI shifts $q_\ell \in \mathbb{C}^-$ for a couple of rhs $\tilde{Y}$.

Theorem [B./Heinkenschloss/Saak/Weichelt '16]

- assume $(\mathbf{A}, \mathbf{B}; \mathbf{M})$ stabilizable, $(\mathbf{C}, \mathbf{A}; \mathbf{M})$ detectable
- $\Rightarrow \exists$ unique, symmetric solution $X^{(*)} = \widehat{\Theta}_r \mathcal{X}^{(*)} \widehat{\Theta}_r^T$ with $\mathcal{R}(\mathcal{X}^{(*)}) = 0$ that stabilizes
$$\left(\begin{bmatrix} A - BB^T X^{(*)} M & \widehat{G} \\ \widehat{G}^T & 0 \end{bmatrix}, \begin{bmatrix} M & 0 \\ 0 & 0 \end{bmatrix} \right)$$
- for $\{X^{(k)}\}_{k=0}^{\infty}$ defined by $X^{(k)} := \widehat{\Theta}_r \mathcal{X}^{(k)} \widehat{\Theta}_r^T$, (1), and $X^{(0)}$ symmetric with $\left(\mathbf{A} - \mathbf{B}\left(\mathbf{K}^{(0)}\right)^T, \mathbf{M}\right)$ stable, it holds that, for $k \geq 1$,
$$X^{(1)} \succeq X^{(2)} \succeq \cdots \succeq X^{(k)} \succeq 0 \quad \text{and} \quad \lim_{k \to \infty} X^{(k)} = X^{(*)}$$
- $\exists\, 0 < \widetilde{\kappa} < \infty$ such that, for $k \geq 1$,
$$||X^{(k+1)} - X^{(*)}||_F \leq \widetilde{\kappa} ||X^{(k)} - X^{(*)}||_F^2$$

Additional Contributions [Bänsch/B./Saak/Weichelt '15,'16]

- Suitable approximation framework for Raymond's projected boundary control input.
- Proposed method directly iterates on the feedback matrix $K \in \mathbb{R}^{n \times n_r}$.
- Initial feedback for index-2 DAE systems using a special eigenvalue shifting technique.
- Improved ADI shift computation for index-2 DAE systems (Penzl- and projection shifts).

Feedback Stabilization for Index-2 DAE Systems — Remarks/Open Problems —

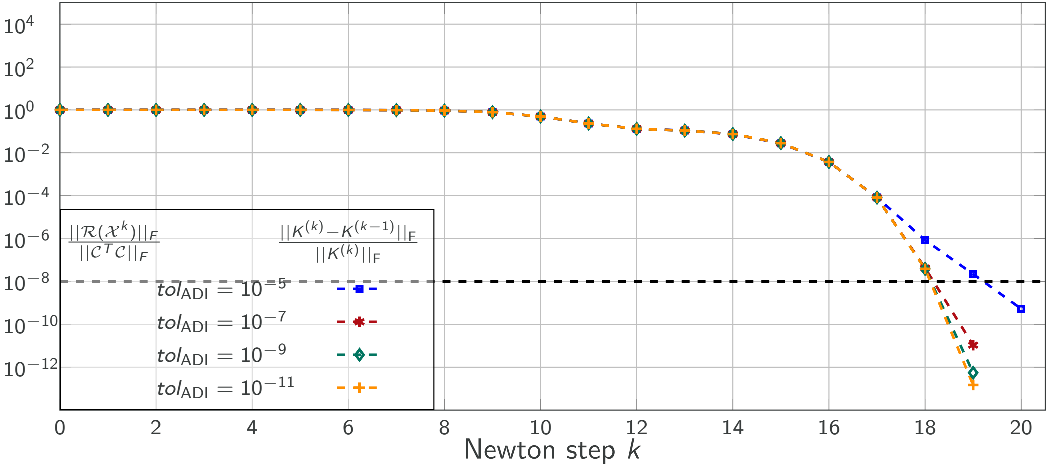
Additional Contributions [Bänsch/B./Saak/Weichelt '15,'16]

- Suitable approximation framework for Raymond's projected boundary control input.
- Proposed method directly iterates on the feedback matrix $K \in \mathbb{R}^{n \times n_r}$.
- Initial feedback for index-2 DAE systems using a special eigenvalue shifting technique.
- Improved ADI shift computation for index-2 DAE systems (Penzl- and projection shifts).

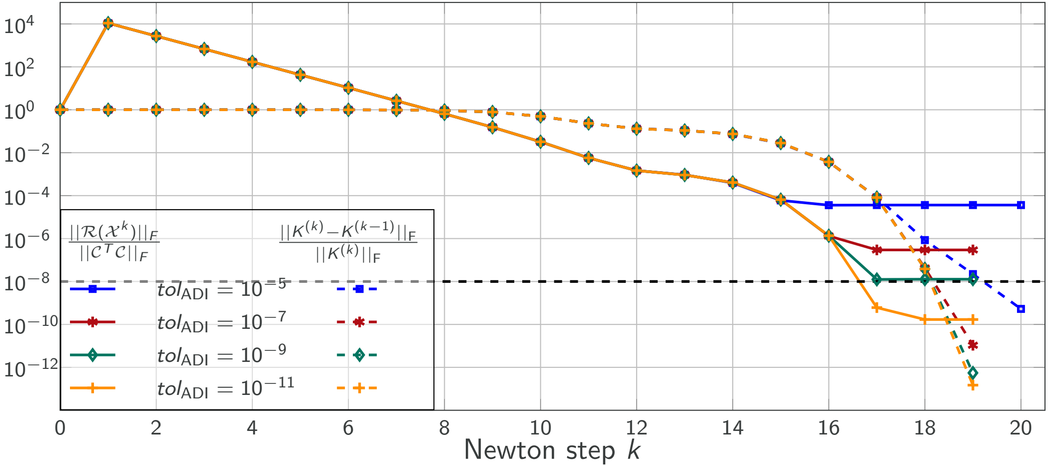
Problems up to here

- Determination of suitable stopping criteria/tolerances.
- Computation of projected residuals is very costly ($\approx$ 10x ADI step).
  $\Rightarrow$ use relative change of feedback matrix [B./Li/Penzl '08]

NSE scenario: $\mathrm{Re} = 500$, $n = 5\,468$, $\lambda = 10^2$, $tol_{\mathrm{Newton}} = 10^{-8}$



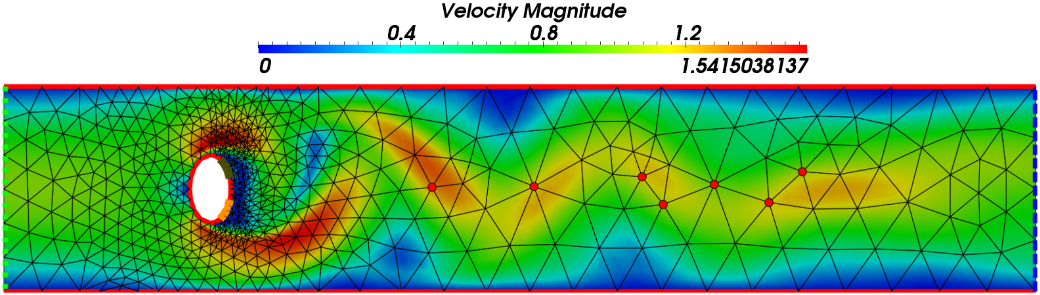
NSE scenario: $\mathrm{Re} = 500$, $n = 5\,468$, $\lambda = 10^2$, $tol_{\mathrm{Newton}} = 10^{-8}$



- Coefficients of GCARE are large-scale matrices (resulting from FE discretization).
- Quadratic system matrices A, $M = M^T \in \mathbb{R}^{n \times n}$ are sparse.

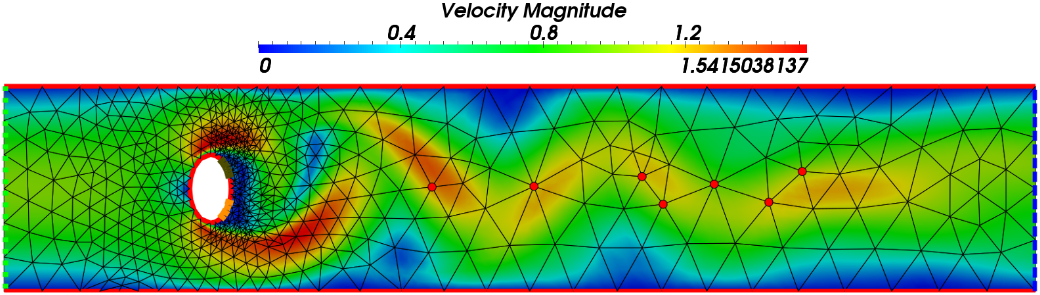
$$\mathcal{R}(X) = C^T C + A^T XM + MXA - MXBB^T XM$$

- Coefficients of GCARE are large-scale matrices (resulting from FE discretization).
- Quadratic system matrices A, $M = M^T \in \mathbb{R}^{n \times n}$ are sparse.



Kármán vortex street

- Coefficients of GCARE are large-scale matrices (resulting from FE discretization).
- Quadratic system matrices A, $M = M^T \in \mathbb{R}^{n\times n}$ are sparse.
- In-/output matrices are rectangular and dense: $B \in \mathbb{R}^{n\times n_r}$, $C \in \mathbb{R}^{n_a\times n}$ with $n_r + n_a \ll n$.



Kármán vortex street

- Coefficients of GCARE are large-scale matrices (resulting from FE discretization).
- Quadratic system matrices A, $M = M^T \in \mathbb{R}^{n \times n}$ are sparse.
- In-/output matrices are rectangular and dense: $B \in \mathbb{R}^{n \times n_r}$, $C \in \mathbb{R}^{n_a \times n}$ with $n_r + n_a \ll n$.
- Unique stabilizing solution $X \in \mathbb{R}^{n \times n}$ is symmetric, positive-semidefinite, but dense [Lancaster/Rodman '95], [B./Heinkenschloss/Saak/Weichelt '16].

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

[B./Kürschner/Saak '14/'15].

Accelerated Solution of Riccati Equations
— Structure —

- Coefficients of GCARE are large-scale matrices (resulting from FE discretization).
- Quadratic system matrices A, $M = M^T \in \mathbb{R}^{n \times n}$ are sparse.
- In-/output matrices are rectangular and dense: $B \in \mathbb{R}^{n \times n_r}$, $C \in \mathbb{R}^{n_a \times n}$ with $n_r + n_a \ll n$.
- Unique stabilizing solution $X \in \mathbb{R}^{n \times n}$ is symmetric, positive-semidefinite, but dense [Lancaster/Rodman '95], [B./Heinkenschloss/Saak/Weichelt '16].

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

[B./Kürschner/Saak '14/'15].

Accelerated Solution of Riccati Equations
— Structure —

- Coefficients of GCARE are large-scale matrices (resulting from FE discretization).
- Quadratic system matrices A, $M = M^T \in \mathbb{R}^{n \times n}$ are sparse.
- In-/output matrices are rectangular and dense: $B \in \mathbb{R}^{n \times n_r}$, $C \in \mathbb{R}^{n_a \times n}$ with $n_r + n_a \ll n$.
- Unique stabilizing solution $X \in \mathbb{R}^{n \times n}$ is symmetric, positive-semidefinite, but dense [Lancaster/Rodman '95], [B./Heinkenschloss/Saak/Weichelt '16].
- Singular values of X decay rapidly [Grasedyck '04], [B./Bujanović '16]
  $\Rightarrow X = ZZ^T$ exists, with $Z \in \mathbb{R}^{n \times m}$, $n_r + n_a < m \ll n$.

$$\mathcal{R}(X) = C^T C + A^T XM + MXA - MXBB^T XM$$

[B./Kürschner/Saak '14/'15].

- Coefficients of GCARE are large-scale matrices (resulting from FE discretization).
- Quadratic system matrices A, $M = M^T \in \mathbb{R}^{n \times n}$ are sparse.
- In-/output matrices are rectangular and dense: $B \in \mathbb{R}^{n \times n_r}$, $C \in \mathbb{R}^{n_a \times n}$ with $n_r + n_a \ll n$.
- Unique stabilizing solution $X \in \mathbb{R}^{n \times n}$ is symmetric, positive-semidefinite, but dense [Lancaster/Rodman '95], [B./Heinkenschloss/Saak/Weichelt '16].
- Singular values of X decay rapidly [Grasedyck '04], [B./Bujanović '16]
  $\Rightarrow X = ZZ^T$ exists, with $Z \in \mathbb{R}^{n \times m}$, $n_r + n_a < m \ll n$.

$$\mathcal{R}(ZZ^T) = C^T C + A^T ZZ^T M + MZZ^T A - MZZ^T BB^T ZZ^T M$$

[B./Kürschner/Saak '14/'15].

- Coefficients of GCARE are large-scale matrices (resulting from FE discretization).
- Quadratic system matrices A, $M = M^T \in \mathbb{R}^{n \times n}$ are sparse.
- In-/output matrices are rectangular and dense: $B \in \mathbb{R}^{n \times n_r}$, $C \in \mathbb{R}^{n_a \times n}$ with $n_r + n_a \ll n$.
- Unique stabilizing solution $X \in \mathbb{R}^{n \times n}$ is symmetric, positive-semidefinite, but dense [Lancaster/Rodman '95], [B./Heinkenschloss/Saak/Weichelt '16].
- Singular values of X decay rapidly [Grasedyck '04], [B./Bujanović '16] $\Rightarrow X = ZZ^T$ exists, with $Z \in \mathbb{R}^{n \times m}$, $n_r + n_a < m \ll n$.
- Residual is of low rank; $R(ZZ^T) = WW^T$, $W \in \mathbb{R}^{n \times k}$, $k \leq 2n_r + n_a \ll n$

$$WW^T = C^T C + A^T ZZ^T M + MZZ^T A - MZZ^T BB^T ZZ^T M$$

[B./Kürschner/Saak '14/'15].

- Nested iteration depends on accuracy of different nesting levels that influence each other.

- Nested iteration depends on accuracy of different nesting levels that influence each other.
  $\Rightarrow$ **inexact Kleinman–Newton method** **[Feitzinger/Hylla/Sachs '09]**

- Nested iteration depends on accuracy of different nesting levels that influence each other.
  ⇒ **inexact Kleinman–Newton method** **[Feitzinger/Hylla/Sachs '09]**

- Convergence theory in [Feitzinger/Hylla/Sachs '09] is not applicable in the low-rank case.

- Nested iteration depends on accuracy of different nesting levels that influence each other.
  ⇒ **inexact Kleinman–Newton method** **[Feitzinger/Hylla/Sachs '09]**
- Kleinman–Newton method converges globally, but often

$$||\mathcal{R}(X^{(1)})||_F \gg ||\mathcal{R}(X^{(0)})||_F.$$

- Convergence theory in [Feitzinger/Hylla/Sachs '09] is not applicable in the low-rank case.

- Nested iteration depends on accuracy of different nesting levels that influence each other.
  $\Rightarrow$ **inexact Kleinman–Newton method** **[Feitzinger/Hylla/Sachs '09]**
- Kleinman–Newton method converges globally, but often

$$||\mathcal{R}(X^{(1)})||_F \gg ||\mathcal{R}(X^{(0)})||_F.$$

$\Rightarrow$ **Kleinman–Newton with exact line search** **[B./Byers '98]**

- Convergence theory in [Feitzinger/Hylla/Sachs '09] is not applicable in the low-rank case.

- Nested iteration depends on accuracy of different nesting levels that influence each other.
  $\Rightarrow$ **inexact Kleinman–Newton method** **[Feitzinger/Hylla/Sachs '09]**
- Kleinman–Newton method converges globally, but often

$$||\mathcal{R}(X^{(1)})||_F \gg ||\mathcal{R}(X^{(0)})||_F.$$

  $\Rightarrow$ **Kleinman–Newton with exact line search** **[B./Byers '98]**

- Convergence theory in [Feitzinger/Hylla/Sachs '09] is not applicable in the low-rank case.

- Step size computation in [B./Byers '98] involves dense residuals, therefore, it is not applicable in large-scale case.

- Nested iteration depends on accuracy of different nesting levels that influence each other.
  $\Rightarrow$ **inexact Kleinman–Newton method** **[Feitzinger/Hylla/Sachs '09]**
- Kleinman–Newton method converges globally, but often

$$||\mathcal{R}(X^{(1)})||_F \gg ||\mathcal{R}(X^{(0)})||_F.$$

  $\Rightarrow$ **Kleinman–Newton with exact line search** **[B./Byers '98]**

$\Rightarrow$ **inexact low-rank Kleinman–Newton-ADI with line search**
[B./Heinkenschloss/Saak/Weichelt '16/'18]

- combination yields convergence proof
- efficient implementation exploits low-rank structure

- Nested iteration depends on accuracy of different nesting levels that influence each other.
  $\Rightarrow$ **inexact Kleinman–Newton method** **[Feitzinger/Hylla/Sachs '09]**
- Kleinman–Newton method converges globally, but often

$$||\mathcal{R}(X^{(1)})||_F \gg ||\mathcal{R}(X^{(0)})||_F.$$

  $\Rightarrow$ **Kleinman–Newton with exact line search** **[B./Byers '98]**

$\Rightarrow$ **inexact low-rank Kleinman–Newton-ADI with line search**

[B./Heinkenschloss/Saak/Weichelt '16/'18]

- combination yields convergence proof
- efficient implementation exploits low-rank structure
- drastically reduced amount of ADI steps + step size computation "for free"

- Nested iteration depends on accuracy of different nesting levels that influence each other.
  $\Rightarrow$ **inexact Kleinman–Newton method** **[Feitzinger/Hylla/Sachs '09]**
- Kleinman–Newton method converges globally, but often

$$||\mathcal{R}(X^{(1)})||_F \gg ||\mathcal{R}(X^{(0)})||_F.$$

  $\Rightarrow$ **Kleinman–Newton with exact line search** **[B./Byers '98]**

$\Rightarrow$ **inexact low-rank Kleinman–Newton-ADI with line search**
[B./Heinkenschloss/Saak/Weichelt '16/'18]

- combination yields convergence proof
- efficient implementation exploits low-rank structure
- drastically reduced amount of ADI steps + step size computation "for free"
- **extension to index-2 DAE case "straight forward"**

Theorem **[B./Heinkenschloss/Saak/Weichelt '16]**

Set $\tau_k \in (0,1)$ and assume: $(\mathcal{A},\mathcal{B};\mathcal{M})$ stabilizable, $(\mathcal{C},\mathcal{A};\mathcal{M})$ detectable, and $\exists\, \widetilde{\mathcal{X}}^{(k+1)} \succeq 0\ \forall k$ that solves

$$(\mathcal{A}-\mathcal{B}\mathcal{K}^{(k)})^T \widetilde{\mathcal{X}}^{(k+1)}\mathcal{M} + \mathcal{M}\widetilde{\mathcal{X}}^{(k+1)}(\mathcal{A}-\mathcal{B}\mathcal{K}^{(k)}) = -\mathcal{C}^T\mathcal{C} - (\mathcal{K}^{(k)})^T\mathcal{K}^{(k)} + \mathcal{L}^{(k+1)}$$

such that

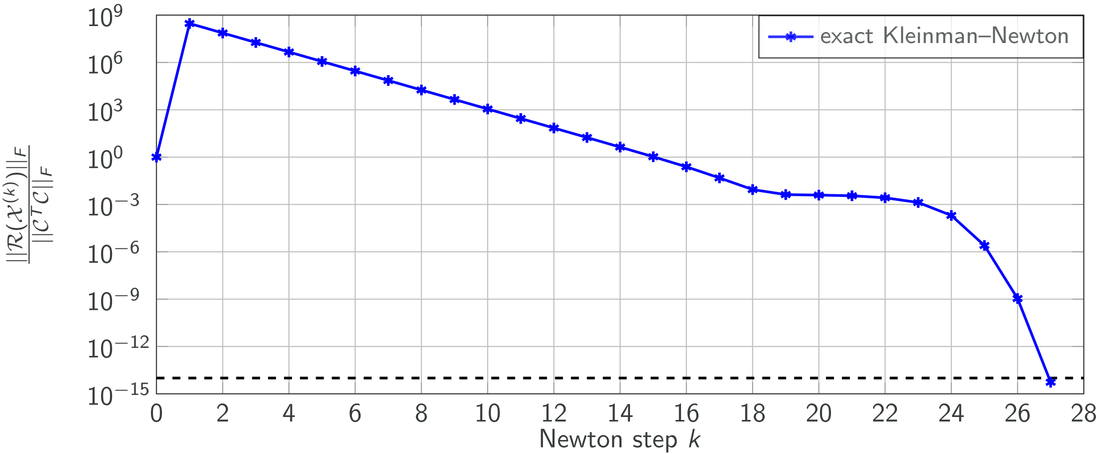
$$||\mathcal{L}^{(k+1)}||_F \leq \tau_k ||\mathcal{R}(\mathcal{X}^{(k)})||_F.$$

Find $\xi_k \in (0,1]$ such that $||\mathcal{R}(\mathcal{X}^{(k)} + \xi_k \mathcal{S}^{(k)})||_F \leq (1-\xi_k\alpha)||\mathcal{R}(\mathcal{X}^{(k)})||_F$ and set

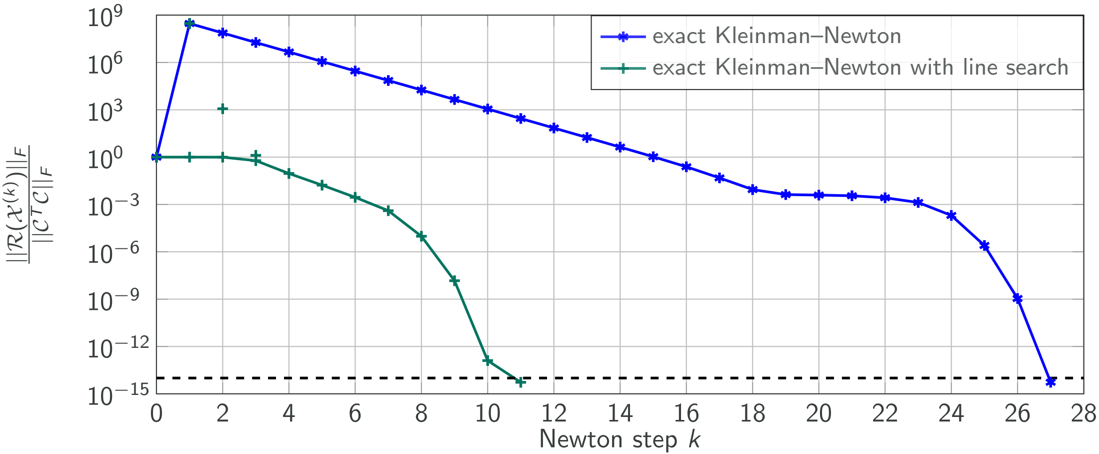
$$\mathcal{X}^{(k+1)} = (1-\xi_k)\mathcal{X}^{(k)} + \xi_k \widetilde{\mathcal{X}}^{(k+1)}.$$

1. **IF** $\xi_k \geq \xi_{\min} > 0\ \forall k \ \Rightarrow\ \|\mathcal{R}(\mathcal{X}^{(k)})\|_F \to 0$.
2. **IF** $\mathcal{X}^{(k)} \succeq 0$, and $(\mathcal{A}-\mathcal{B}\mathcal{B}^T\mathcal{X}^{(k)}, \mathcal{M})$ stable for $k \geq K > 0 \ \Rightarrow\ \mathcal{X}^{(k)} \to \mathcal{X}^{(*)}$ ($\mathcal{X}^{(*)} \succeq 0$ the unique stabilizing solution).

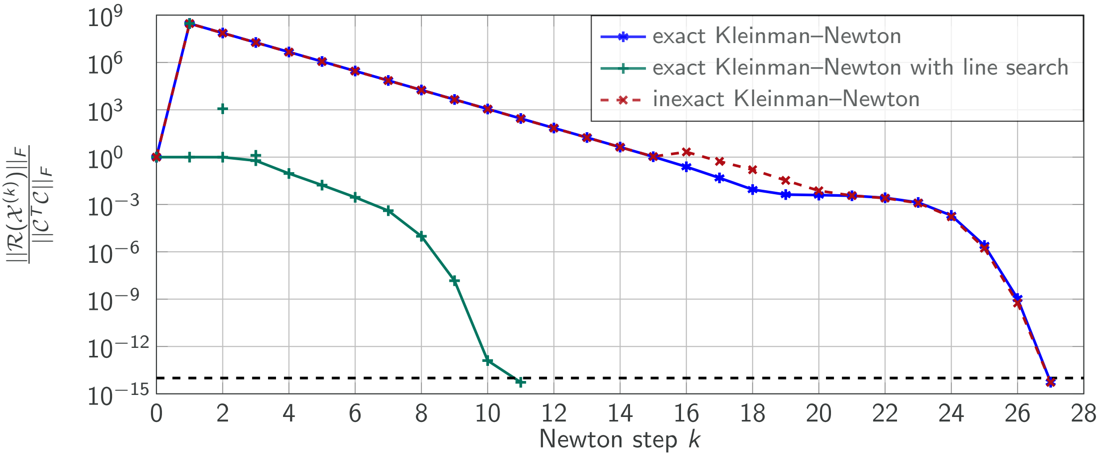
NSE scenario: Re $= 500$, Level 1, $\lambda = 10^4$, $tol_{\text{Newton}} = 10^{-14}$



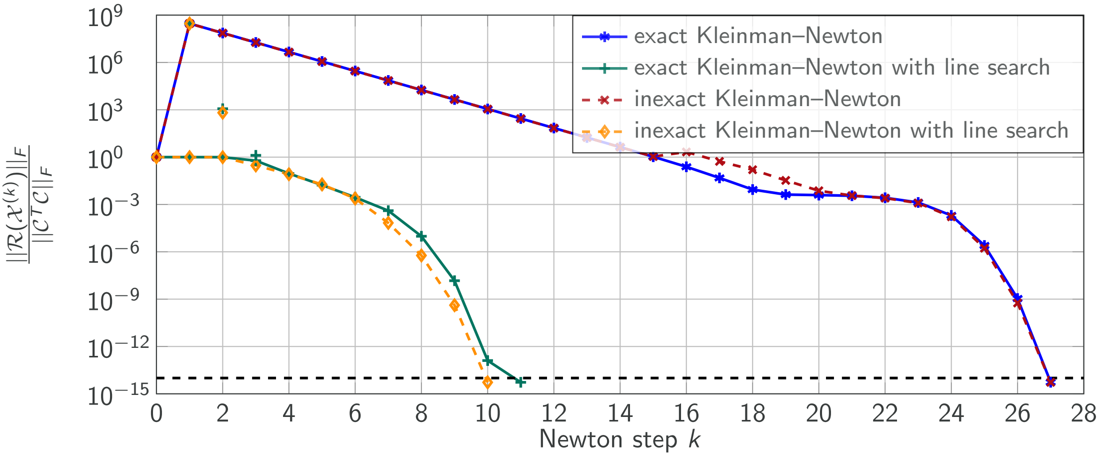
NSE scenario: Re $= 500$, Level 1, $\lambda = 10^4$, $tol_{\text{Newton}} = 10^{-14}$



NSE scenario: Re $= 500$, Level 1, $\lambda = 10^4$, $tol_{\text{Newton}} = 10^{-14}$



NSE scenario: Re $= 500$, Level 1, $\lambda = 10^4$, $tol_{\text{Newton}} = 10^{-14}$



NSE scenario: Re $= 500$, Level 1, $\lambda = 10^4$, $tol_{\text{Newton}} = 10^{-14}$

	exact KN	exact KN+LS	inexact KN	inexact KN+LS
$\#Newt$	27	11	27	10
$\#ADI$	3185	1351	852	549
$t_{\text{Newt-ADI}}$	1304.769	540.984	331.871	222.295
t_{shift}	29.998	12.568	7.370	5.507
t_{LS}	–		–	
t_{total}	**1334.767**	**553.581**	**339.241**	**227.824**

Table: Numbers of steps and timings in seconds.

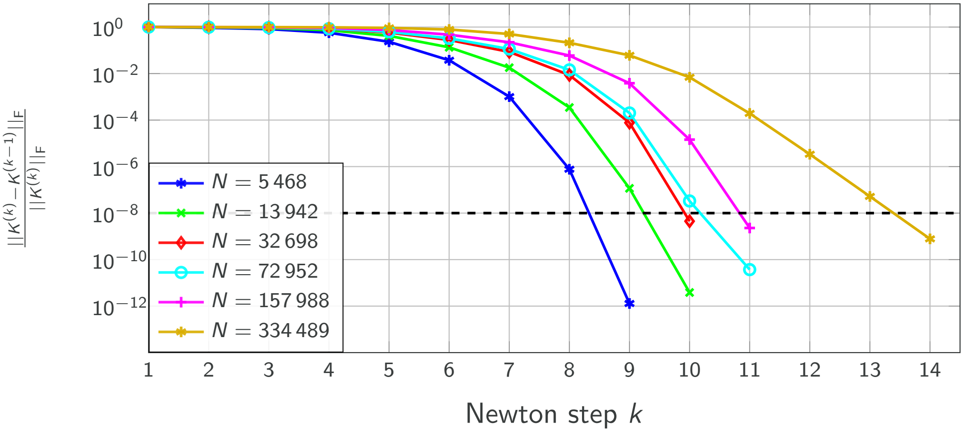
NSE scenario: Re $= 500$, Level 1, $\lambda = 10^4$, $tol_{\text{Newton}} = 10^{-14}$

	exact KN	exact KN+LS	inexact KN	inexact KN+LS
$\#Newt$	27	11	27	10
$\#ADI$	3185	1351	852	549
$t_{\text{Newt-ADI}}$	1304.769	540.984	331.871	222.295
t_{shift}	29.998	12.568	7.370	5.507
t_{LS}	–	**0.029**	–	**0.023**
t_{total}	**1334.767**	**553.581**	**339.241**	**227.824**

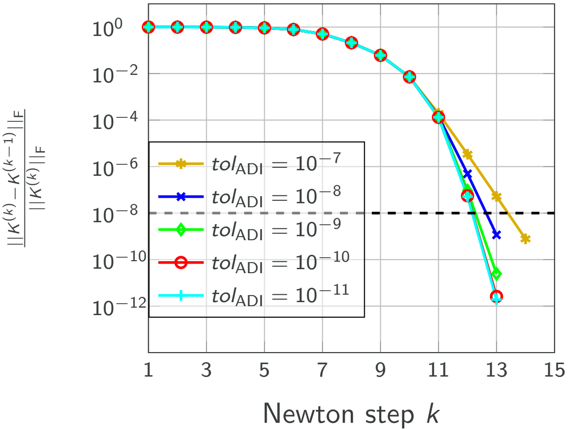
Table: Numbers of steps and timings in seconds.

Accelerated Solution of Riccati Equations
Feedback Stabilization for Index-2 DAE systems

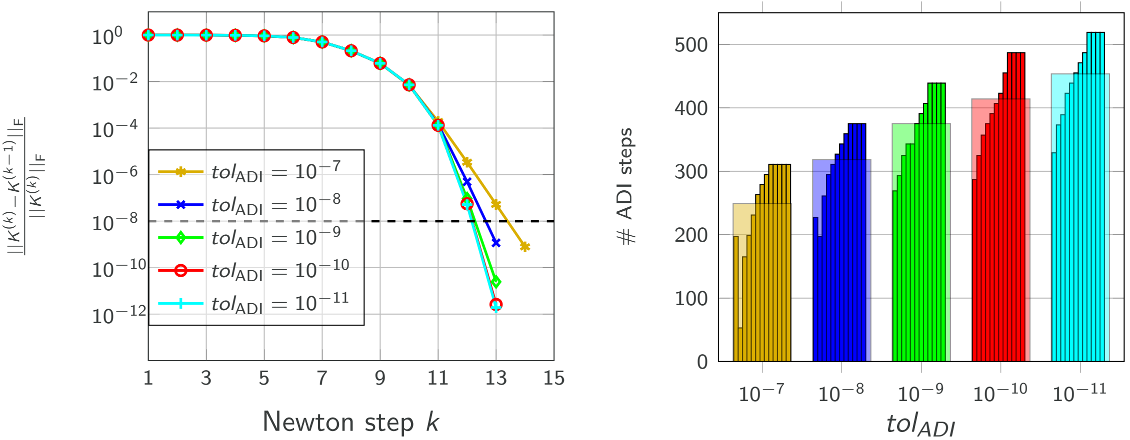
NSE scenario: Re $= 500$, $tol_{\text{ADI}} = 10^{-7}$, $tol_{\text{Newton}} = 10^{-8}$



NSE scenario: NSE scenario: $\mathrm{Re} = 500$, $tol_{\mathrm{Newton}} = 10^{-8}$, $N = 334\,489$



NSE scenario: NSE scenario: Re $= 500$, $tol_{\text{Newton}} = 10^{-8}$, $N = 334\,489$



Conclusions

Main Contributions

- Analyzed **Riccati-based feedback** for **scalar** and **vector-valued transport** problems.
- Wide-spread usability tailored for standard **inf-sup stable finite element** discretizations.
- Established **specially tailored Kleinman–Newton-ADI** that **avoids explicit projections**.
- **Suitable preconditioners** for multi-field flow problems have been developed.
- **Ongoing research** in similar areas has been **incorporated**.
- Major run time improvements due to combination of **inexact Newton** and **line search**.
- Established **new convergence proofs** that were verified by **extensive numerical tests**.

Conclusions

Main Contributions

- Analyzed **Riccati-based feedback** for **scalar** and **vector-valued transport** problems.
- Wide-spread usability tailored for standard **inf-sup stable finite element** discretizations.
- Established **specially tailored Kleinman–Newton-ADI** that **avoids explicit projections**.
- **Suitable preconditioners** for multi-field flow problems have been developed.
- **Ongoing research** in similar areas has been **incorporated**.
- Major run time improvements due to combination of **inexact Newton** and **line search**.
- Established **new convergence proofs** that were verified by **extensive numerical tests**.

$\Rightarrow$ Showed **overall usability** of new approach by a closed-loop forward simulation.

References

- E. Bänsch and P. Benner, *Stabilization of incompressible flow problems by Riccati-based feedback*, in Constrained Optimization and Optimal Control for Partial Differential Equations, vol. 160 of International Series of Numerical Mathematics, Birkhäuser, 2012, pp. 5–20.
- P. Benner, J. Saak, M. Stoll, and H. K. Weichelt, *Efficient solution of large-scale saddle point systems arising in Riccati-based boundary feedback stabilization of incompressible Stokes flow*, **SIAM J. Sci. Comput.**, 35 (2013), pp. S150–S170.
- P. Benner, J. Saak, M. Stoll, and H. K. Weichelt, *Efficient Solvers for Large-Scale Saddle Point Systems Arising in Feedback Stabilization of Multi-Field Flow Problems*, in System Modeling and Optimization, vol. 443 of IFIP Adv. Inf. Commun. Technol., New York, 2014, Springer, pp. 11–20.
- E. Bänsch, P. Benner, J. Saak, and H. K. Weichelt, *Optimal control-based feedback stabilization of multi-field flow problems*, in Trends in PDE Constrained Optimization, vol. 165 of Internat. Ser. Numer. Math., Birkhäuser, Basel, 2014, pp. 173–188.
- E. Bänsch, P. Benner, J. Saak, and H. K. Weichelt, *Riccati-based boundary feedback stabilization of incompressible Navier-Stokes flows*, **SIAM J. Sci. Comput.**, 37 (2015), pp. A832–A858.
- P. Benner, M. Heinkenschloss, J. Saak, and H. K. Weichelt, *An inexact low-rank Newton-ADI method for large-scale algebraic Riccati equations*, **Appl. Numer. Math.**, 108 (2016), pp. 125–142.
- P. Benner, M. Heinkenschloss, J. Saak, and H. K. Weichelt, *Efficient solution of large-scale algebraic Riccati equations associated with index-2 DAEs via the inexact low-rank Newton-ADI method*, **arXiv:1804.01410**, April 2018.