Abstract

We discuss the feedback stabilization of flow problems described by the incompressible Navier-Stokes equations. In the last decade, a series of papers by Raymond and co-workers showed that for small perturbations, the deviation from a nominal flow, defined by a possibly unstable solution of the steady Navier-Stokes equations, can be steered to zero at an exponential convergence rate using an LQR problem for the velocity field projected onto a suitable space of divergence-free functions. We show how to solve this LQR problem numerically using the associated algebraic (operator) Riccati equation. The key idea is to avoid the explicit Helmholtz projection onto the divergence-free vector fields by employing a saddle point formulation discussed already by Heinkenschloss, Sorensen, and Sun (SIAM J. Sci. Comp. 30:1038-1063, 2008) in the context of balanced truncation model reduction. Also, a number of other issues such as initializing Newton's method for the algebraic Riccati equations, need to be solved to derive a working algorithm for the numerical solution of the flow stabilization problem. We will show how the computed feedback control using this approach effectively stabilizes unstable flows using as test examples von Karman vortex shedding and the coupled systems of a reactive substance transported by an incompressible fluid.

Numerical Solution of the Feedback Control Problem for Navier-Stokes Equations

Peter Benner[1,2] Jens Saak[1,2] Heiko K. Weichelt[2]

[1] Max Planck Institute for Dynamics of Complex Technical Systems Magdeburg,
Research group Computational Methods in Systems and Control Theory

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

IFIP TC 7 / 2013 System Modelling and Optimization,
Alpen-Adria Universität Klagenfurt (AAU), Austria, September 8-13, 2013

Minisymposion
On Optimal Feedback Control for Partial Differential Equations:
Theory and Numerical Methods

Overview

Introduction
Model Problems

Flow Models

Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \nu \Delta \vec{v} + \nabla p = \vec{f}$$
$$\operatorname{div} \vec{v} = 0$$

Navier-Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \frac{1}{\mathrm{Re}} \Delta \vec{v} + (\vec{v} \cdot \nabla)\vec{v} + \nabla p = \vec{f}$$
$$\operatorname{div} \vec{v} = 0$$

- defined on $(0, \infty) \times \Omega$, $\Omega \subset \mathbb{R}^2$ bounded and "smooth enough" $\Gamma = \partial\Omega$
- $+$ boundary and initial conditions

- models describe incompressible, instationary flow
- viscosity $\nu \in \mathbb{R}^+$, (NSE: Reynolds number $\mathrm{Re} = \frac{v_{ch} \cdot d_{ch}}{\nu} \in \mathbb{R}^+$)
- initial boundary value problem with additional algebraic constraints

Introduction

Model Problems

Flow Models

Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \nu \Delta \vec{v} + \nabla p = \vec{f}$$
$$\operatorname{div} \vec{v} = 0$$

Navier-Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \frac{1}{\mathrm{Re}} \Delta \vec{v} + (\vec{v} \cdot \nabla)\vec{v} + \nabla p = \vec{f}$$
$$\operatorname{div} \vec{v} = 0$$

Introduction

Model Problems

Flow Models

Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \nu \Delta \vec{v} + \nabla p = \vec{f}$$
$$\operatorname{div} \vec{v} = 0$$

Navier-Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \frac{1}{\mathrm{Re}} \Delta \vec{v} + (\vec{v} \cdot \nabla)\vec{v} + \nabla p = \vec{f}$$
$$\operatorname{div} \vec{v} = 0$$

Diffusion-Convection Models

Concentration Equation

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

Temperature Equation

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

- defined on $(0, \infty) \times \Omega$, $\Omega \subset \mathbb{R}^2$ bounded and "smooth enough" $\Gamma = \partial\Omega$
- $+$ boundary and initial conditions

- models describe diffusion and convection process
- Schmidt number $\mathrm{Sc} \in \mathbb{R}^+$, Prandtl number $\mathrm{Pr} \in \mathbb{R}^+$

Introduction

Model Problems

Flow Models

Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \nu \Delta \vec{v} + \nabla p = \vec{f}$$
$$\operatorname{div} \vec{v} = 0$$

Navier-Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \frac{1}{\mathrm{Re}} \Delta \vec{v} + (\vec{v} \cdot \nabla)\vec{v} + \nabla p = \vec{f}$$
$$\operatorname{div} \vec{v} = 0$$

Diffusion-Convection Models

Concentration Equation

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

Temperature Equation

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

- **Scenario 1:** Feedback stabilization of flow field around stationary trajectory in "von Kármán Vortex Street".

Introduction

Model Problems

Flow Models

Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \nu \Delta \vec{v} + \nabla p = \vec{f}$$
$$\operatorname{div} \vec{v} = 0$$

Navier-Stokes Equations

$$\frac{\partial \vec{v}}{\partial t} - \frac{1}{\mathrm{Re}} \Delta \vec{v} + (\vec{v} \cdot \nabla) \vec{v} + \nabla p = \vec{f}$$
$$\operatorname{div} \vec{v} = 0$$

Diffusion-Convection Models

Concentration Equation

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

Temperature Equation

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

- **Scenario 2:** Feedback stabilization of coupled flow and diffusion-convection field in a reactor model.

Introduction

Basic Ideas of Feedback Stabilization

- **Motivation:**
  - ↪ Stabilize flow profiles.
  - ↪ Attenuate external perturbations.
  - ↪ Influence flow via boundary conditions.

Introduction

Basic Ideas of Feedback Stabilization

- **Motivation:**
  - ↪ Stabilize flow profiles.
  - ↪ Attenuate external perturbations.
  - ↪ Influence flow via boundary conditions.
- Riccati-based feedback stabilization with boundary control input.
  - ↪ Use *linear quadratic regulator* **(LQR)** approach.
  - ↪ Influence the model via **boundary control**.
  - ↪ **Stabilize** the flow around a desired flow profile (stationary trajectory) that is used as **linearization point**.

Introduction

Basic Ideas of Feedback Stabilization

- **Motivation:**
  - ↪ Stabilize flow profiles.
  - ↪ Attenuate external perturbations.
  - ↪ Influence flow via boundary conditions.
- Riccati-based feedback stabilization with boundary control input.
  - ↪ Use *linear quadratic regulator* **(LQR)** approach.
  - ↪ Influence the model via **boundary control**.
  - ↪ **Stabilize** the flow around a desired flow profile (stationary trajectory) that is used as **linearization point**.
- Analytical approach by [Raymond since 2005].
  - ↪ Uses *Leray projector* to project onto the correct subspace.
  - ↪ Extended to finite dimensional controllers [Raymond/Thevenet '10].

Introduction

Basic Ideas of Feedback Stabilization

- **Motivation:**
  - ↪ Stabilize flow profiles.
  - ↪ Attenuate external perturbations.
  - ↪ Influence flow via boundary conditions.
- Riccati-based feedback stabilization with boundary control input.
  - ↪ Use *linear quadratic regulator* **(LQR)** approach.
  - ↪ Influence the model via **boundary control**.
  - ↪ **Stabilize** the flow around a desired flow profile (stationary trajectory) that is used as **linearization point**.
- Analytical approach by [Raymond since 2005].
  - ↪ Uses *Leray projector* to project onto the correct subspace.
  - ↪ Extended to finite dimensional controllers [Raymond/Thevenet '10].
- Ideas for numerical treatment based on [Bänsch/Benner '10].
  - ↪ Consider **linearized Navier-Stokes equations for 2D**.
  - ↪ Discrete **projection** idea by [Heinkenschloss/Sorensen/Sun '08].
  - ↪ Use *Newton-ADI* method to compute **optimal control**.

Introduction

Analytical Approach

[Raymond since 2005]

- Feedback boundary stabilization of (Navier)-Stokes flow problems.
- Linearize around a given stationary trajectory.
- Formulate stabilization problem for the perturbation.

Introduction

Analytical Approach

[RAYMOND since 2005]

- Feedback boundary stabilization of (Navier)-Stokes flow problems.
- Linearize around a given stationary trajectory.
- Formulate stabilization problem for the perturbation.

Problem

- Partial differential equation with additional algebraic constrains.

  $\hookrightarrow$ Need projector onto correct solution manifold.

  $\hookrightarrow$ "Hide" constraints inside the function space.

Introduction

Analytical Approach

[RAYMOND since 2005]

- Feedback boundary stabilization of (Navier)-Stokes flow problems.
- Linearize around a given stationary trajectory.
- Formulate stabilization problem for the perturbation.

Problem

- Partial differential equation with additional algebraic constrains.
  $\hookrightarrow$ Need projector onto correct solution manifold.
  $\hookrightarrow$ "Hide" constraints inside the function space.

Leray Projector

- Orthogonal projection onto divergence-free function space.
  $\hookrightarrow$ LQR approach can be applied.
  $\hookrightarrow$ Perturbation vanishes and flow achieves given trajectory.

Introduction

Leray Projection

Helmholtz Decomposition [Girault/Raviart '86]

- Splitting a vector field $\vec{v} \in (L^2(\mathbb{R}^d))^d$ into:
  - Divergence-free component: $\vec{v}_{div} \in \mathbf{H}_{div0}(\mathbb{R}^d)$
  - Curl-free component: $\vec{v}_{curl} \in \mathbf{H}_{curl0}(\mathbb{R}^d)$

  $\hookrightarrow \exists$ stream-function ψ and potential-function p:

$$\vec{v} = \vec{v}_{div} + \vec{v}_{curl}$$

with $\vec{v}_{div} = \operatorname{curl}\psi$ $(\operatorname{div}\vec{v}_{div} = 0)$ and $\vec{v}_{curl} = \nabla p$ $(\operatorname{curl}\vec{v}_{curl} = 0)$.

Introduction

Leray Projection

Helmholtz Decomposition [Girault/Raviart '86]

- Splitting a vector field $\vec{v} \in (L^2(\mathbb{R}^d))^d$ into:
  - Divergence-free component: $\vec{v}_{div} \in \mathbf{H}_{div0}(\mathbb{R}^d)$
  - Curl-free component: $\vec{v}_{curl} \in \mathbf{H}_{curl0}(\mathbb{R}^d)$
    $\hookrightarrow \exists$ stream-function ψ and potential-function p:

$$\vec{v} = \vec{v}_{div} + \vec{v}_{curl} \quad \Rightarrow \quad (L^2(\mathbb{R}^d))^d = \mathbf{H}_{div0}(\mathbb{R}^d) \oplus^{\perp} \mathbf{H}_{curl0}(\mathbb{R}^d)$$

with $\vec{v}_{div} = \operatorname{curl}\psi$ $(\operatorname{div}\vec{v}_{div} = 0)$ and $\vec{v}_{curl} = \nabla p$ $(\operatorname{curl}\vec{v}_{curl} = 0)$.

Introduction

Leray Projection

Helmholtz Decomposition [Girault/Raviart '86]

- Splitting a vector field $\vec{v} \in (L^2(\mathbb{R}^d))^d$ into:
  - Divergence-free component: $\vec{v}_{div} \in \mathbf{H}_{div0}(\mathbb{R}^d)$
  - Curl-free component: $\vec{v}_{curl} \in \mathbf{H}_{curl0}(\mathbb{R}^d)$
    $\hookrightarrow \exists$ stream-function ψ and potential-function p:

$$\vec{v} = \vec{v}_{div} + \vec{v}_{curl} \quad \Rightarrow \quad (L^2(\mathbb{R}^d))^d = \mathbf{H}_{div0}(\mathbb{R}^d) \oplus^{\perp} \mathbf{H}_{curl0}(\mathbb{R}^d)$$

with $\vec{v}_{div} = \operatorname{curl} \psi$ ($\operatorname{div} \vec{v}_{div} = 0$) and $\vec{v}_{curl} = \nabla p$ ($\operatorname{curl} \vec{v}_{curl} = 0$).

Leray Projector P

$P : \vec{v} \mapsto \vec{v}_{div}$, which means for a given $\vec{v}$ solve

$$\begin{aligned} \vec{v}_{div} + \nabla p &= \vec{v}, \\ \operatorname{div} \vec{v}_{div} &= 0, \end{aligned}$$

to compute the divergence-free component $\vec{v}_{div}$.

Discretized Control Systems

Finite Element Discretization

- Applying a standard finite element discretization to the linearized flow/coupled flow problems yields

$$M\frac{d}{dt}\mathbf{v}(t) = A\mathbf{v}(t) + G\mathbf{p}(t) + \mathbf{f}(t), \tag{1a}$$

$$0 = G^T\mathbf{v}(t). \tag{1b}$$

Discretized Control Systems

Finite Element Discretization

- Applying a standard finite element discretization to the linearized flow/coupled flow problems yields

$$M\frac{d}{dt}\mathbf{v}(t) = A\mathbf{v}(t) + G\mathbf{p}(t) + \mathbf{f}(t), \tag{1a}$$

$$0 = G^T\mathbf{v}(t), \tag{1b}$$

$$\mathbf{y}(t) = C\mathbf{v}(t). \tag{1c}$$

Discretized Control Systems

Finite Element Discretization

- Applying a standard finite element discretization to the linearized flow/coupled flow problems yields

$$M\frac{d}{dt}\mathbf{v}(t) = A\mathbf{v}(t) + G\mathbf{p}(t) + B\mathbf{u}(t), \tag{1a}$$

$$0 = G^T\mathbf{v}(t), \tag{1b}$$

$$\mathbf{y}(t) = C\mathbf{v}(t). \tag{1c}$$

Discretized Control Systems

Finite Element Discretization

- Applying a standard finite element discretization to the linearized flow/coupled flow problems yields

$$\boxed{\begin{aligned} M\frac{d}{dt}\mathbf{v}(t) &= A\mathbf{v}(t) + G\mathbf{p}(t) \\ 0 &= G^T\mathbf{v}(t), \end{aligned}} \quad + B\mathbf{u}(t), \tag{1a, 1b}$$

$$\mathbf{y}(t) = C\mathbf{v}(t). \tag{1c}$$

Properties

- Differential algebraic system (DAE) of D-index 2 (if $\tilde{G}$ has full rank).
- Matrix pencil:

$$\left(\begin{bmatrix} A & G \\ G^T & 0 \end{bmatrix}, \begin{bmatrix} M & 0 \\ 0 & 0 \end{bmatrix} \right).$$

Discretized Control Systems

Finite Element Discretization

- Applying a standard finite element discretization to the linearized flow/coupled flow problems yields

$$\boxed{\begin{aligned} M\frac{d}{dt}\mathbf{v}(t) &= A\mathbf{v}(t) + G\mathbf{p}(t) + B\mathbf{u}(t), && \text{(1a)}\\ 0 &= G^T\mathbf{v}(t), && \text{(1b)}\\ \mathbf{y}(t) &= C\mathbf{v}(t). && \text{(1c)} \end{aligned}}$$

Properties

- Differential algebraic system (DAE) of D-index 2 (if $\tilde{G}$ has full rank).
- Matrix pencil:

$$\left(\begin{bmatrix} A & G \\ G^T & 0 \end{bmatrix}, \begin{bmatrix} M & 0 \\ 0 & 0 \end{bmatrix}\right).$$

- Descriptor system with multiple inputs and outputs (MIMO).

Discretized Control Systems

Finite Element Discretization

- Applying a standard finite element discretization to the linearized flow/coupled flow problems yields

$$M\frac{d}{dt}\mathbf{v}(t) = A\mathbf{v}(t) + G\mathbf{p}(t) + B\mathbf{u}(t), \tag{1a}$$

$$0 = G^T\mathbf{v}(t), \tag{1b}$$

$$\mathbf{y}(t) = C\mathbf{v}(t). \tag{1c}$$

Properties

- Differential algebraic system (DAE) of D-index 2 (if $\tilde{G}$ has full rank).
- Matrix pencil:

$$\left(\begin{bmatrix} A & G \\ G^T & 0 \end{bmatrix}, \begin{bmatrix} M & 0 \\ 0 & 0 \end{bmatrix}\right).$$

- Descriptor system with multiple inputs and outputs (MIMO).
- Index reduction to apply LQR approach [Heinkenschloss/Sorensen/Sun '08].

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [HEINKENSCHLOSS/SORENSEN/SUN '08]

- Index reduction for Lyapunov-solver.
- Projector:

$$\Pi := I - G(G^T M^{-1} G)^{-1} G^T M^{-1}.$$

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [Heinkenschloss/Sorensen/Sun '08]

Properties of Π: $(\Pi := I - G(G^TM^{-1}G)^{-1}G^TM^{-1})$

$$\Pi^2 = \Pi \qquad \mathrm{null}(\Pi) = \mathrm{range}(G)$$

$$\Pi M = M\Pi^T \qquad \mathrm{range}(\Pi) = \mathrm{null}(G^TM^{-1})$$

$$G^T\mathbf{x} = 0 \Longleftrightarrow \Pi^T\mathbf{x} = \mathbf{x}$$

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [Heinkenschloss/Sorensen/Sun '08]

Properties of Π: $(\Pi := I - G(G^T M^{-1} G)^{-1} G^T M^{-1})$

$$\Pi^2 = \Pi \qquad \mathrm{null}(\Pi) = \mathrm{range}(G)$$

$$\Pi M = M \Pi^T \qquad \mathrm{range}(\Pi) = \mathrm{null}(G^T M^{-1})$$

$$G^T \mathbf{x} = 0 \iff \Pi^T \mathbf{x} = \mathbf{x}$$

Π^T seems to be discrete Leray projector

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [Heinkenschloss/Sorensen/Sun '08]

- Properties of Π^T: $(\Pi^T := I - M^{-T} G (G^T M^{-1} G)^{-1} G^T)$

$$(\Pi^T)^2 = \Pi^T \qquad \mathrm{null}(\Pi^T) = \mathrm{range}(M^{-1} G)$$

$$\Pi M = M \Pi^T \qquad \mathrm{range}(\Pi^T) = \mathrm{null}(G^T)$$

$$G^T \mathbf{x} = 0 \Longleftrightarrow \Pi^T \mathbf{x} = \mathbf{x}$$

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [HEINKENSCHLOSS/SORENSEN/SUN '08]

- Index reduction for Lyapunov-solver
- Projector: $\Pi := I - G(G^T M^{-1} G)^{-1} G^T M^{-1}$

Properties of Π^T: $(\Pi^T := I - M^{-T} G (G^T M^{-1} G)^{-1} G^T)$

$$(\Pi^T)^2 = \Pi^T \qquad \mathrm{null}(\Pi^T) = \mathrm{range}(M^{-1}G)$$

$$\Pi M = M \Pi^T \qquad \mathrm{range}(\Pi^T) = \mathrm{null}(G^T)$$

$$G^T \mathbf{x} = 0 \iff \Pi^T \mathbf{x} = \mathbf{x}$$

– Projection onto divergence-free functions ($\mathrm{div}\, \vec{v} = 0$).

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [Heinkenschloss/Sorensen/Sun '08]

- Index reduction for Lyapunov-solver
- Projector:

$$\Pi := I - G(G^T M^{-1} G)^{-1} G^T M^{-1}$$

Properties of Π^T: $(\Pi^T := I - M^{-T} G (G^T M^{-1} G)^{-1} G^T)$

$$(\Pi^T)^2 = \Pi^T \qquad \mathrm{null}(\Pi^T) = \mathrm{range}(M^{-1}G)$$

$$\Pi M = M \Pi^T \qquad \mathrm{range}(\Pi^T) = \mathrm{null}(G^T)$$

$$G^T \mathbf{x} = 0 \iff \Pi^T \mathbf{x} = \mathbf{x}$$

- Projection onto divergence-free functions ($\mathrm{div}\, \vec{v} = 0$).
- Nullspace represents curl-free components ($\mathrm{rot}\, \nabla p = 0$).

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [Heinkenschloss/Sorensen/Sun '08]

- Properties of Π^T: $(\Pi^T := I - M^{-T}G(G^TM^{-1}G)^{-1}G^T)$

$$(\Pi^T)^2 = \Pi^T \qquad \mathrm{null}(\Pi^T) = \mathrm{range}(M^{-1}G)$$

$$\Pi M = M\Pi^T \qquad \mathrm{range}(\Pi^T) = \mathrm{null}(G^T)$$

$$G^T\mathbf{x} = 0 \iff \Pi^T\mathbf{x} = \mathbf{x}$$

  - Projection onto divergence-free functions ($\mathrm{div}\,\vec{v} = 0$).
  - Nullspace represents curl-free components ($\mathrm{rot}\,\nabla p = 0$).
  - Symmetric w.r.t. $(.\,,.)_M$ (i.e., the discrete $(.\,,.)_{L_2}$)
    $\Rightarrow$ oblique in $(\mathbb{R}^n, (.\,,.)_2)$ but orthogonal in $(\mathbb{R}^n, (.\,,.)_M)$.

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [HEINKENSCHLOSS/SORENSEN/SUN '08]

- Index reduction for Lyapunov-solver.
- Projector:

$$\Pi^T := I - M^{-1}G(G^TM^{-1}G)^{-1}G^T.$$

Recall $P(\vec{v}) = \vec{w}$:

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

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [Heinkenschloss/Sorensen/Sun '08]

- Index reduction for Lyapunov-solver.
- Projector:

$$\Pi^T := I - M^{-1}G(G^TM^{-1}G)^{-1}G^T.$$

Recall $P(\vec{v}) = \vec{w}$:

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

$$M\mathbf{w} + \quad G\mathbf{p} = \quad M\mathbf{v}$$

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [Heinkenschloss/Sorensen/Sun '08]

- Index reduction for Lyapunov-solver.
- Projector:

$$\Pi^T := I - M^{-1}G(G^TM^{-1}G)^{-1}G^T.$$

Recall $P(\vec{v}) = \vec{w}$:

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

$$G^TM^{-1}M\mathbf{w} + G^TM^{-1}G\mathbf{p} = G^TM^{-1}M\mathbf{v}$$

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [Heinkenschloss/Sorensen/Sun '08]

- Index reduction for Lyapunov-solver.
- Projector:

$$\Pi^T := I - M^{-1}G(G^T M^{-1} G)^{-1} G^T.$$

Recall $P(\vec{v}) = \vec{w}$:

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

$$G^T \mathbf{w} + G^T M^{-1} G \mathbf{p} = G^T M^{-1} M \mathbf{v}$$

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [Heinkenschloss/Sorensen/Sun '08]

- Index reduction for Lyapunov-solver.
- Projector:

$$\Pi^T := I - M^{-1}G(G^TM^{-1}G)^{-1}G^T.$$

Recall $P(\vec{v}) = \vec{w}$:

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

$$G^T\mathbf{w} + G^TM^{-1}G\mathbf{p} = G^T\mathbf{v}$$

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [Heinkenschloss/Sorensen/Sun '08]

- Index reduction for Lyapunov-solver.
- Projector:

$$\Pi^T := I - M^{-1}G(G^T M^{-1} G)^{-1} G^T.$$

Recall $P(\vec{v}) = \vec{w}$:

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

$$G^T\mathbf{w} + G^T M^{-1} G\mathbf{p} = G^T\mathbf{v}$$

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [Heinkenschloss/Sorensen/Sun '08]

- Index reduction for Lyapunov-solver.
- Projector:

$$\Pi^T := I - M^{-1}G(G^TM^{-1}G)^{-1}G^T.$$

Recall $P(\vec{v}) = \vec{w}$:

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

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

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [Heinkenschloss/Sorensen/Sun '08]

- Index reduction for Lyapunov-solver.
- Projector:

$$\Pi^T := I - M^{-1}G(G^TM^{-1}G)^{-1}G^T.$$

Recall $P(\vec{v}) = \vec{w}$:

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

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

$$M\mathbf{w} + G(G^TM^{-1}G)^{-1}G^T\mathbf{v} = M\mathbf{v}$$

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [Heinkenschloss/Sorensen/Sun '08]

- Index reduction for Lyapunov-solver.
- Projector:

$$\Pi^T := I - M^{-1}G(G^TM^{-1}G)^{-1}G^T.$$

Recall $P(\vec{v}) = \vec{w}$:

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

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

$$M\mathbf{w} = M(I - M^{-1}G(G^TM^{-1}G)^{-1}G^T)\mathbf{v}$$

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [Heinkenschloss/Sorensen/Sun '08]

- Index reduction for Lyapunov-solver.
- Projector:

$$\Pi^T := I - M^{-1}G(G^TM^{-1}G)^{-1}G^T.$$

Recall $P(\vec{v}) = \vec{w}$:

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

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

$$\mathbf{w} = (I - M^{-1}G(G^TM^{-1}G)^{-1}G^T)\mathbf{v}$$

Discretized Control Systems

Discrete *Leray* Projection

Projection Method [Heinkenschloss/Sorensen/Sun '08]

- Index reduction for Lyapunov-solver.
- Projector:

$$\Pi^T := I - M^{-1}G(G^TM^{-1}G)^{-1}G^T.$$

Recall $P(\vec{v}) = \vec{w}$:

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

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

$$\mathbf{w} = (I - M^{-1}G(G^TM^{-1}G)^{-1}G^T)\mathbf{v}$$

Leray vs. Π^T

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

Discretized Control Systems

Discrete *Leray* Projection

Projection of Control System

- Projector:

$$\Pi^T := I - M^{-1}G(G^T M^{-1} G)^{-1} G^T.$$

- For $G^T \mathbf{v}(t) = \mathbf{0} \Leftrightarrow \Pi^T \mathbf{v}(t) = \mathbf{v}(t)$.
- Correct solution manifold (*hidden manifold*)

$$\mathbf{0} = G^T M^{-1} A\mathbf{v}(t) + G^T M^{-1} G\mathbf{p}(t) + G^T M^{-1} B\mathbf{u}(t),$$

is invariant under Π^T.

Discretized Control Systems

Discrete *Leray* Projection

Projection of Control System

- Projector:

$$\Pi^T := I - M^{-1}G(G^TM^{-1}G)^{-1}G^T.$$

- For $G^T\mathbf{v}(t) = \mathbf{0} \Leftrightarrow \Pi^T\mathbf{v}(t) = \mathbf{v}(t)$.
- Correct solution manifold (*hidden manifold*)

$$\mathbf{0} = G^TM^{-1}A\mathbf{v}(t) + G^TM^{-1}G\mathbf{p}(t) + G^TM^{-1}B\mathbf{u}(t),$$

  is invariant under Π^T.
- System (1) reduces to

$$\Pi M \Pi^T \frac{d}{dt}\mathbf{v}(t) = \Pi A \Pi^T \mathbf{v}(t) + \Pi B\mathbf{u}(t), \tag{2a}$$

$$\mathbf{y}(t) = C\Pi^T\mathbf{v}(t). \tag{2b}$$

Discretized Control Systems

Discrete *Leray* Projection

Projection of Control System

- Projector:

$$\Pi^T := I - M^{-1}G(G^T M^{-1} G)^{-1} G^T.$$

- For $G^T \mathbf{v}(t) = \mathbf{0} \Leftrightarrow \Pi^T \mathbf{v}(t) = \mathbf{v}(t)$.
- Correct solution manifold (*hidden manifold*)

$$\mathbf{0} = G^T M^{-1} A \mathbf{v}(t) + G^T M^{-1} G \mathbf{p}(t) + G^T M^{-1} B \mathbf{u}(t),$$

is invariant und not invertible, because nullspace of Π is non trivial

- System (1) reduces to

$$\Pi M \Pi^T \frac{d}{dt}\mathbf{v}(t) = \Pi A \Pi^T \mathbf{v}(t) + \Pi B \mathbf{u}(t), \quad (2a)$$

$$\mathbf{y}(t) = C \Pi^T \mathbf{v}(t). \quad (2b)$$

Discretized Control Systems

Discrete *Leray* Projection

Decomposition of Projector [Heinkenschloss/Sorensen/Sun '08]

- Consider decomposition:

$$\Pi = \Theta_l \Theta_r^T,$$

with $\Theta_l, \Theta_r \in \mathbb{R}^{n_v \times (n_v - n_p)}$, such that $\Theta_l^T \Theta_r = I_{(n_v-n_p)\times(n_v-n_p)}$.

Discretized Control Systems

Discrete *Leray* Projection

Decomposition of Projector [Heinkenschloss/Sorensen/Sun '08]

- Consider decomposition:

$$\Pi = \Theta_l \Theta_r^T,$$

with $\Theta_l, \Theta_r \in \mathbb{R}^{n_v \times (n_v - n_p)}$, such that $\Theta_l^T \Theta_r = I_{(n_v - n_p) \times (n_v - n_p)}$.

- Substituting the decomposition into (2) yields

$$\begin{aligned} \Theta_r^T M \Theta_r \frac{d}{dt}\tilde{\mathbf{v}}(t) &= \Theta_r^T A \Theta_r \tilde{\mathbf{v}}(t) + \Theta_r^T B \mathbf{u}(t), \\ \mathbf{y}(t) &= C \Theta_r \tilde{\mathbf{v}}(t). \end{aligned} \tag{3}$$

Discretized Control Systems

Discrete *Leray* Projection

Decomposition of Projector [Heinkenschloss/Sorensen/Sun '08]

- Consider decomposition:

$$\Pi = \Theta_l \Theta_r^T,$$

with $\Theta_l, \Theta_r \in \mathbb{R}^{n_v \times (n_v - n_p)}$, such that $\Theta_l^T \Theta_r = I_{(n_v - n_p) \times (n_v - n_p)}$.

- Substituting the decomposition into (2) yields

$$\begin{aligned} \mathcal{M} \frac{d}{dt} \tilde{\mathbf{v}}(t) &= \mathcal{A} \tilde{\mathbf{v}}(t) + \mathcal{B} \mathbf{u}(t), \\ \mathbf{y}(t) &= \mathcal{C} \tilde{\mathbf{v}}(t), \end{aligned} \tag{3}$$

with $\tilde{\mathbf{v}} \in \mathbb{R}^{n_v - n_p}$ and $\mathcal{M} = \mathcal{M}^T \succ 0$.

Discretized Control Systems

LQR Approach for Projected System

[Benner/Saak/Stoll/W. 12]

Minimize

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

subject to

$$\begin{aligned} \mathcal{M}\frac{d}{dt}\tilde{\mathbf{v}}(t) &= \mathcal{A}\tilde{\mathbf{v}}(t) + \mathcal{B}\mathbf{u}(t), \\ \mathbf{y}(t) &= \mathcal{C}\tilde{\mathbf{v}}(t). \end{aligned} \tag{4}$$

Discretized Control Systems

LQR Approach for Projected System

[BENNER/SAAK/STOLL/W. 12]

Minimize

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

subject to

$$\begin{aligned} \mathcal{M} \frac{d}{dt} \tilde{\mathbf{v}}(t) &= \mathcal{A} \tilde{\mathbf{v}}(t) + \mathcal{B} \mathbf{u}(t), \\ \mathbf{y}(t) &= \mathcal{C} \tilde{\mathbf{v}}(t). \end{aligned} \tag{4}$$

Riccati Based Feedback Approach [e.g.,LOCATELLI '01]

- Optimal control: $\mathbf{u}(t) = -\mathcal{K}\tilde{\mathbf{v}}(t)$.
- Feedback: $\mathcal{K} = \mathcal{B}^T X \mathcal{M}$,

  where X is the solution of the generalized algebraic Riccati equation

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

Work Flow

Scenario 1: NSE on "von Kármán Vortex Street"

PDE: NSE

Goal: $\vec{z} = \vec{v} - \vec{w} \to 0$

$\rightsquigarrow$ Linearized Navier-Stokes equations:

$$\frac{\partial \vec{z}}{\partial t} - \frac{1}{\mathrm{Re}}\Delta \vec{z} + (\vec{z}\cdot\nabla)\vec{w} + (\vec{w}\cdot\nabla)\vec{z} + \nabla p = 0$$

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

defined on $(0, \infty) \times \Omega$

$+$ boundary and initial conditions

LQR

Minimize

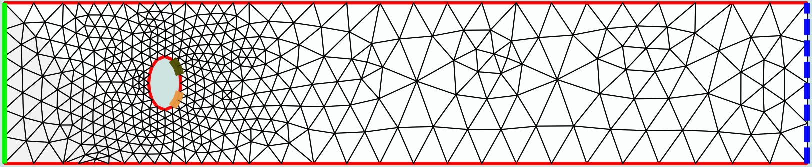
$$\mathcal{J}(\mathbf{y}, \mathbf{u}) = \frac{1}{2}\int_0^\infty \lambda||\mathbf{y}||^2 + ||\mathbf{u}||^2 \,\mathrm{dt}$$

s.t.

$$\begin{bmatrix} M_z & 0 \\ 0 & 0 \end{bmatrix} \frac{d}{dt} \begin{bmatrix} \mathbf{z} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} A_z & G \\ G^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \mathbf{p} \end{bmatrix} + \begin{bmatrix} B_z \\ 0 \end{bmatrix} \mathbf{u}$$

$$\mathbf{y}(t) = C_z \mathbf{z}(t)$$

Domain Ω: *von Kármán vortex street*



Work Flow

Scenario 1: NSE on "von Kármán Vortex Street"

NSE

stationary NSE

PDE: NSE

Goal: $\vec{z} = \vec{v} - \vec{w} \to 0$

$\rightsquigarrow$ Linearized Navier-Stokes equations:

$$\frac{\partial \vec{z}}{\partial t} - \frac{1}{\mathrm{Re}} \Delta \vec{z} + (\vec{z} \cdot \nabla) \vec{w} + (\vec{w} \cdot \nabla) \vec{z} + \nabla p = 0$$

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

defined on $(0, \infty) \times \Omega$

$+$ boundary and initial conditions

LQR

Minimize

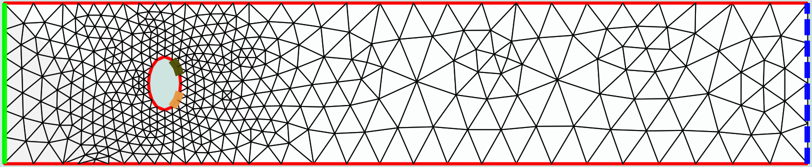
$$\mathcal{J}(\mathbf{y}, \mathbf{u}) = \frac{1}{2} \int_0^\infty \lambda ||\mathbf{y}||^2 + ||\mathbf{u}||^2 \, \mathrm{dt}$$

s.t.

$$\begin{bmatrix} M_z & 0 \\ 0 & 0 \end{bmatrix} \frac{d}{dt} \begin{bmatrix} \mathbf{z} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} A_z & G \\ G^T & 0 \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \mathbf{p} \end{bmatrix} + \begin{bmatrix} B_z \\ 0 \end{bmatrix} \mathbf{u}$$

$$\mathbf{y}(t) = C_z \mathbf{z}(t)$$

Domain Ω: *von Kármán vortex street*



Work Flow

Scenario 1: NSE on "von Kármán Vortex Street"

PDE: NSE

Goal: $\vec{z} = \vec{v} - \vec{w} \to 0$

$\rightsquigarrow$ Linearized Navier-Stokes equations:

$$\frac{\partial \vec{z}}{\partial t} - \frac{1}{\mathrm{Re}}\Delta \vec{z} + (\vec{z} \cdot \nabla)\vec{w} + (\vec{w} \cdot \nabla)\vec{z} + \nabla p = 0$$
$$\operatorname{div} \vec{z} = 0$$

defined on $(0, \infty) \times \Omega$

$+$ boundary and initial conditions

LQR

Minimize

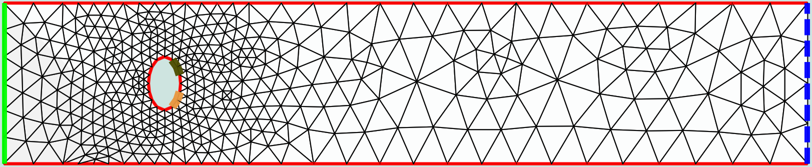
$$\mathcal{J}(\mathbf{y}, \mathbf{u}) = \frac{1}{2}\int_0^\infty \lambda||\mathbf{y}||^2 + ||\mathbf{u}||^2 \,\mathrm{dt}$$

s.t.

$$\mathcal{M}\frac{d}{dt}\tilde{\mathbf{z}} = \mathcal{A}\tilde{\mathbf{z}} + \mathcal{B}\mathbf{u}$$
$$\mathbf{y}(t) = \mathcal{C}\tilde{\mathbf{z}}$$

[Heinkenschloss/Sorensen/Sun '08]

Domain Ω: *von Kármán vortex street*



Work Flow

Scenario 2: NSE Coupled with DCE in Reactor Model

PDE: NSE+DCE

Goal: $\vec{z} = \vec{v} - \vec{w} \to 0, c = c^{(\vec{v})} - c^{(\vec{w})} \to 0$

$\rightsquigarrow$ Linearized coupled system:

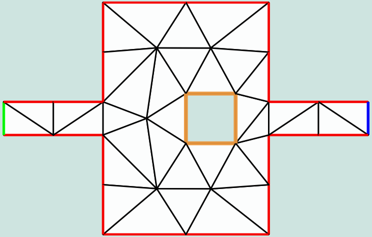
$$\frac{\partial \vec{z}}{\partial t} - \frac{1}{\mathrm{Re}}\Delta \vec{z} + (\vec{z}\cdot\nabla)\vec{w} + (\vec{w}\cdot\nabla)\vec{z} + \nabla p = 0$$

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

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

defined on $(0,\infty)\times\Omega$ plus BC,IC

Domain Ω: Reactor Model



LQR

Minimize

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

s.t.

$$\begin{bmatrix} M_z & 0 & 0 \\ 0 & M_c & 0 \\ 0 & 0 & 0 \end{bmatrix} \frac{d}{dt} \begin{bmatrix} \mathbf{z} \\ \mathbf{c} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} A_z & 0 & G \\ R & A_c & 0 \\ G^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{z} \\ \mathbf{c} \\ \mathbf{p} \end{bmatrix} + \begin{bmatrix} B_z \\ 0 \\ 0 \end{bmatrix} \mathbf{u}$$

$$\mathbf{y}(t) = C_c\mathbf{c}$$

Work Flow

Scenario 2: NSE Coupled with DCE DCE ctor Model

stationary DCE

PDE: NSE+DCE

Goal: $\vec{z} = \vec{v} - \vec{w} \to 0$, $c = c^{(\vec{v})} - c^{(\vec{w})} \to 0$

$\rightsquigarrow$ Linearized coupled system:

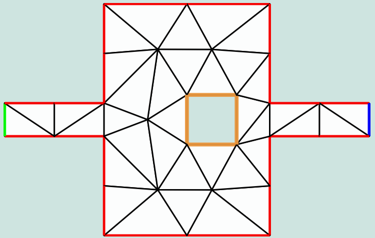
$$\frac{\partial \vec{z}}{\partial t} - \frac{1}{\mathrm{Re}}\Delta\vec{z} + (\vec{z}\cdot\nabla)\vec{w} + (\vec{w}\cdot\nabla)\vec{z} + \nabla p = 0$$

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

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

defined on $(0,\infty)\times\Omega$ plus BC,IC

ain Ω: Reactor Model



LQR

Minimize

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

s.t.

$$\begin{bmatrix} M_z & 0 & 0 \\ 0 & M_c & 0 \\ 0 & 0 & 0 \end{bmatrix} \frac{d}{dt}\begin{bmatrix} \mathbf{z} \\ \mathbf{c} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} A_z & 0 & G \\ R & A_c & 0 \\ G^T & 0 & 0 \end{bmatrix}\begin{bmatrix} \mathbf{z} \\ \mathbf{c} \\ \mathbf{p} \end{bmatrix} + \begin{bmatrix} B_z \\ 0 \\ 0 \end{bmatrix}\mathbf{u}$$

$$\mathbf{y}(t) = C_c\mathbf{c}$$

Work Flow

Scenario 2: NSE Coupled with DCE in Reactor Model

PDE: NSE+DCE

Goal: $\vec{z} = \vec{v} - \vec{w} \to 0$, $c = c^{(\vec{v})} - c^{(\vec{w})} \to 0$

$\rightsquigarrow$ Linearized coupled system:

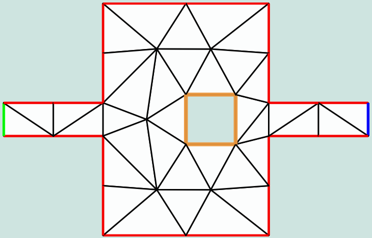
$$\frac{\partial \vec{z}}{\partial t} - \frac{1}{\mathrm{Re}}\Delta \vec{z} + (\vec{z}\cdot\nabla)\vec{w} + (\vec{w}\cdot\nabla)\vec{z} + \nabla p = 0$$

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

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

defined on $(0,\infty)\times\Omega$ plus BC,IC

Domain Ω: Reactor Model



LQR

Minimize

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

s.t.

$$\mathcal{M}\frac{d}{dt}\begin{bmatrix}\tilde{\mathbf{z}}\\ \mathbf{c}\end{bmatrix} = \mathcal{A}\begin{bmatrix}\tilde{\mathbf{z}}\\ \mathbf{c}\end{bmatrix} + \begin{bmatrix}\mathcal{B}\\ 0\end{bmatrix}\mathbf{u}$$

$$\mathbf{y}(t) = C_c\mathbf{c}$$

[Heinkenschloss/Sorensen/Sun '08]

Work Flow

LQR for Nonlinear PDEs with Algebraic Constraints

Continuous Level

- Linearize around a given stationary trajectory.
- Index reduction via projection method. [Heinkenschloss/Sorensen/Sun '08]
- Formulate stabilization problem for the perturbation.

Work Flow

LQR for Nonlinear PDEs with Algebraic Constraints

Continuous Level

- Linearize around a given stationary trajectory.
- Index reduction via projection method. [Heinkenschloss/Sorensen/Sun '08]
- Formulate stabilization problem for the perturbation.

Semi-Discretized Level

- Discretization via inf-sup stable FE (Taylor-Hood-Elements).
- Construct and assemble suitable input and output operators.
- Adapt Newton-ADI approach to deal with projection.

Work Flow

LQR for Nonlinear PDEs with Algebraic Constraints

Continuous Level

- Linearize around a given stationary trajectory.
- Index reduction via projection method. [Heinkenschloss/Sorensen/Sun '08]
- Formulate stabilization problem for the perturbation.

Semi-Discretized Level

- Discretization via inf-sup stable FE (Taylor-Hood-Elements).
- Construct and assemble suitable input and output operators.
- Adapt Newton-ADI approach to deal with projection.

LQR Approach

- Find feedback matrix $\mathcal{K}$ depending on cost functional and projected ODE.

Work Flow

LQR for Nonlinear PDEs with Algebraic Constraints

Continuous Level

- Linearize around a given stationary trajectory.
- Index reduction via projection method. [Heinkenschloss/Sorensen/Sun '08]
- Formulate stabilization problem for the perturbation.

Semi-Discretized Level

- Discretization via inf-sup stable FE (Taylor-Hood-Elements).
- Construct and assemble suitable input and output operators.
- Adapt Newton-ADI approach to deal with projection.

LQR Approach

- Find feedback matrix $\mathcal{K}$ depending on cost functional and projected ODE.

Closed-Loop Simulation

- Apply feedback $\mathcal{K}$ within NAVIER.
- Optimize cost functional design to improve simulation results.

Work Flow

LQR for Nonlinear PDEs with Algebraic Constraints

Continuous Level

- Linearize around a given stationary trajectory.
- Index reduction via projection method. [Heinkenschloss/Sorensen/Sun '08]
- Formulate stabilization problem for the perturbation.

Semi-Discretized Level

- Discretization via inf-sup stable FE (Taylor-Hood-Elements).
- Construct and assemble suitable input and output operators.
- Adapt Newton-ADI approach to deal with projection.

LQR Approach

- Find feedback matrix $\mathcal{K}$ depending on cost functional and projected ODE.

Closed-Loop Simulation

- Apply feedback $\mathcal{K}$ within NAVIER.
- Optimize cost functional design to improve simulation results.

Work Flow

Nested Iteration

Compute feedback matrix $\mathcal{K} = \mathcal{B}^T X \mathcal{M}$ with X solves:

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

Newton Kleinman method

Work Flow

Nested Iteration

Compute feedback matrix $\mathcal{K} = \mathcal{B}^T X \mathcal{M}$ with X solves:

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

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

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

Newton Kleinman method

low rank ADI method

Work Flow

Nested Iteration

Compute feedback matrix $\mathcal{K} = \mathcal{B}^T X \mathcal{M}$ with X solves:

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

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

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

Step i: solve the projected linear system

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)} + q_i\mathcal{M})^T \mathcal{V}_i = \mathcal{Y} \qquad (5)$$

Newton Kleinman method

low rank ADI method

Krylov solver

Work Flow

Nested Iteration

Compute feedback matrix $\mathcal{K} = \mathcal{B}^T X \mathcal{M}$ with X solves:

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

Newton Kleinman method

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

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

low rank ADI method

Step i: solve the projected linear system

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)} + q_i \mathcal{M})^T \mathcal{V}_i = \mathcal{Y} \qquad (5)$$

Krylov solver

Avoid explicit projection using [Heinkenschloss/Sorensen/Sun '08]:

Work Flow

Nested Iteration

Compute feedback matrix $\mathcal{K} = \mathcal{B}^T X \mathcal{M}$ with X solves:

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

Newton Kleinman method

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

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

low rank ADI method

Step i: solve the projected linear system

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)} + q_i \mathcal{M})^T \mathcal{V}_i = \mathcal{Y} \qquad (5)$$

Krylov solver

Avoid explicit projection using [Heinkenschloss/Sorensen/Sun '08]:
Replace (5) and **solve instead** the saddle point system (SPS)

$$\begin{bmatrix} A^T - (K^{(m)})^T B^T + q_i M^T & \tilde{G} \\ \tilde{G}^T & 0 \end{bmatrix} \begin{bmatrix} V_i \\ * \end{bmatrix} = \begin{bmatrix} Y \\ 0 \end{bmatrix}$$

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

Work Flow

Nested Iteration

Compute feedback matrix $\mathcal{K} = \mathcal{B}^T X \mathcal{M}$ with X solves:

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

Newton Kleinman method

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

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

low rank ADI method

Step i: solve the projected linear system

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)} + q_i \mathcal{M})^T \mathcal{V}_i = \mathcal{Y} \qquad (5)$$

Krylov solver

Avoid explicit projection using [Heinkenschloss/Sorensen/Sun '08]:
Replace (5) and **solve instead** the saddle point system (SPS) (using Sherman Morrison Woodbury formula)

$$\begin{bmatrix} A^T - (K^{(m)})^T B^T + q_i M^T & \tilde{G} \\ \tilde{G}^T & 0 \end{bmatrix} \begin{bmatrix} V_i \\ * \end{bmatrix} = \begin{bmatrix} Y \\ 0 \end{bmatrix}$$

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

Work Flow

Nested Iteration

Compute feedback matrix $\mathcal{K} = \mathcal{B}^T X \mathcal{M}$ with X solves:

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

Newton Kleinman method

Step m + 1: solve Lyapunov equation

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

low rank ADI method

Step i: solve the projected linear system

$$(\mathcal{A} - \mathcal{B}\mathcal{K}^{(m)} + q_i \mathcal{M})^T \mathcal{V}_i = \mathcal{Y} \qquad (5)$$

Krylov solver

Avoid explicit projection using [Heinkenschloss/Sorensen/Sun '08]:
Replace (5) and **solve instead** the saddle point system (SPS) (using Sherman Morrison Woodbury formula)

$$\begin{bmatrix} A^T + q_i M^T & \tilde{G} \\ \tilde{G}^T & 0 \end{bmatrix} \begin{bmatrix} V_i \\ * \end{bmatrix} = \begin{bmatrix} \tilde{Y} \\ 0 \end{bmatrix}$$

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

Work Flow

Additional Tasks

- Compute initial feedback for unstable systems.
  - $\hookrightarrow$ Determine the invariant unstable subspace $\mathcal{U}$.
  - $\hookrightarrow$ Solve Bernoulli equation on $\mathcal{U}$ [Benner '11, Amodei/Buchot '12].

Work Flow
Additional Tasks

- Compute initial feedback for unstable systems.
  - $\hookrightarrow$ Determine the invariant unstable subspace $\mathcal{U}$.
  - $\hookrightarrow$ Solve Bernoulli equation on $\mathcal{U}$ [Benner '11, Amodei/Buchot '12].
- Derive an efficient variant of large-scale Newton-ADI.
  - $\hookrightarrow$ Preprint SPP1253-090 [Benner/Saak '10].

Work Flow

Additional Tasks

- Compute initial feedback for unstable systems.
  - $\hookrightarrow$ Determine the invariant unstable subspace $\mathcal{U}$.
  - $\hookrightarrow$ Solve Bernoulli equation on $\mathcal{U}$ [Benner '11, Amodei/Buchot '12].
- Derive an efficient variant of large-scale Newton-ADI.
  - $\hookrightarrow$ Preprint SPP1253-090 [Benner/Saak '10].
- Calculate ADI shift parameters depending on the problem structure.
  - $\hookrightarrow$ Different methods have been tested (Penzl, Wachspress, Saak).
  - $\hookrightarrow$ Infinite eigenvalues of DAE pencil yield additional difficulties.

Work Flow

Additional Tasks

- Compute initial feedback for unstable systems.
  - ↪ Determine the invariant unstable subspace $\mathcal{U}$.
  - ↪ Solve Bernoulli equation on $\mathcal{U}$ [Benner '11, Amodei/Buchot '12].
- Derive an efficient variant of large-scale Newton-ADI.
  - ↪ Preprint SPP1253-090 [Benner/Saak '10].
- Calculate ADI shift parameters depending on the problem structure.
  - ↪ Different methods have been tested (Penzl, Wachspress, Saak).
  - ↪ Infinite eigenvalues of DAE pencil yield additional difficulties.
- Preconditioned iterative solvers for innermost saddle point systems.
  - ↪ Preprint SPP1253-130 for Stokes equations [Benner/Saak/Stoll/W. '12].

Work Flow
Additional Tasks

- Compute initial feedback for unstable systems.
  - ↪ Determine the invariant unstable subspace $\mathcal{U}$.
  - ↪ Solve Bernoulli equation on $\mathcal{U}$ [BENNER '11, AMODEI/BUCHOT '12].
- Derive an efficient variant of large-scale Newton-ADI.
  - ↪ Preprint SPP1253-090 [BENNER/SAAK '10].
- Calculate ADI shift parameters depending on the problem structure.
  - ↪ Different methods have been tested (Penzl, Wachspress, Saak).
  - ↪ Infinite eigenvalues of DAE pencil yield additional difficulties.
- Preconditioned iterative solvers for innermost saddle point systems.
  - ↪ Preprint SPP1253-130 for Stokes equations [BENNER/SAAK/STOLL/W. '12].
- Parameter influence observation during the nested iteration.
  - ↪ 3 stopping criteria, Reynolds & Schmidt number, ADI shifts, regularization parameters in cost functional.

Work Flow

Additional Tasks

SPP1253

- Compute initial feedback for unstable systems.
  - ↪ Determine the invariant unstable subspace $\mathcal{U}$.
  - ↪ Solve Bernoulli equation on $\mathcal{U}$ [Benner '11, Amodei/Buchot '12].
- Derive an efficient variant of large-scale Newton-ADI.
  - ↪ Preprint SPP1253-090 [Benner/Saak '10].
- Calculate ADI shift parameters depending on the problem structure.
  - ↪ Different methods have been tested (Penzl, Wachspress, Saak).
  - ↪ Infinite eigenvalues of DAE pencil yield additional difficulties.
- Preconditioned iterative solvers for innermost saddle point systems.
  - ↪ Preprint SPP1253-130 for Stokes equations [Benner/Saak/Stoll/W. '12].
- Parameter influence observation during the nested iteration.
  - ↪ 3 stopping criteria, Reynolds & Schmidt number, ADI shifts, regularization parameters in cost functional.

Work Flow

Additional Tasks

SPP1253

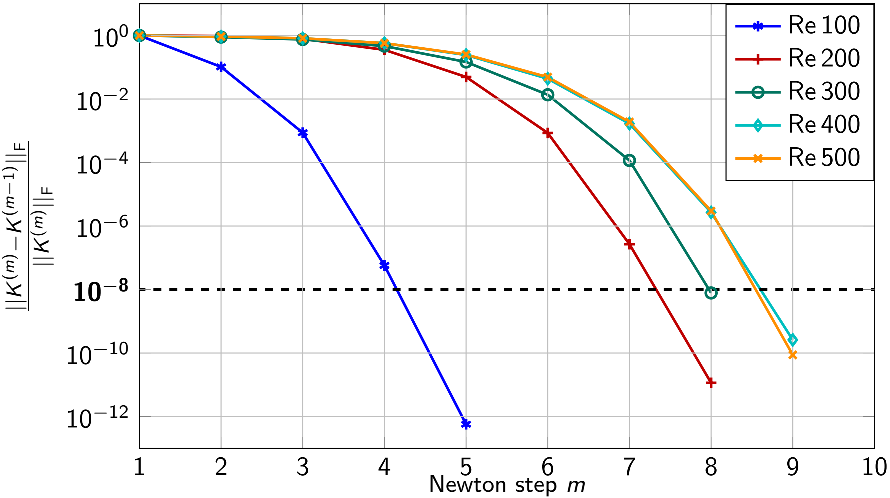
- Compute initial feedback for unstable systems.
  - ↪ Determine the invariant unstable subspace $\mathcal{U}$.
  - ↪ Solve Bernoulli equation on $\mathcal{U}$ [Benner '11, Amodei/Buchot '12].
- Derive an efficient variant of large-scale Newton-ADI.
  - ↪ Preprint SPP1253-090 [Benner/Saak '10].
- Calculate ADI shift parameters depending on the problem structure.
  - ↪ Different methods have been tested (Penzl, Wachspress, Saak).
  - ↪ Infinite eigenvalues of DAE pencil yield additional difficulties.
- Preconditioned iterative solvers for innermost saddle point systems.
  - ↪ Preprint SPP1253-130 for Stokes equations [Benner/Saak/Stoll/W. '12].

work in progress

- Parameter influence observation during the nested iteration.
  - ↪ 3 stopping criteria, Reynolds & Schmidt number, ADI shifts, regularization parameters in cost functional.

Numerical Examples

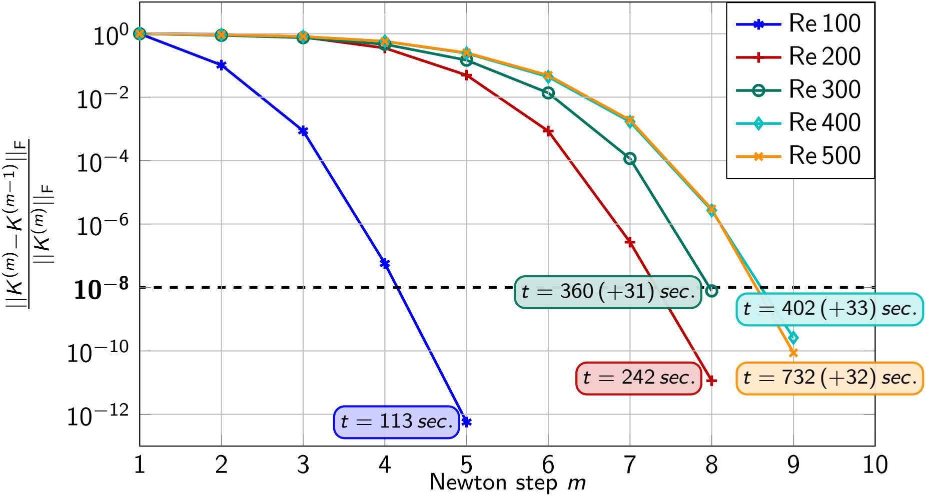
Convergence of Newton-ADI: NSE on von Kármán Vortex Street



Relative change of feedback matrix K for different Reynolds numbers ($\lambda = \rho = 1$, $n = 5468$, direct solver, $tol_{\mathrm{NM}} = 10^{-8}$, $\mathbf{tol_{ADI} = 10^{-4}}$).

Numerical Examples

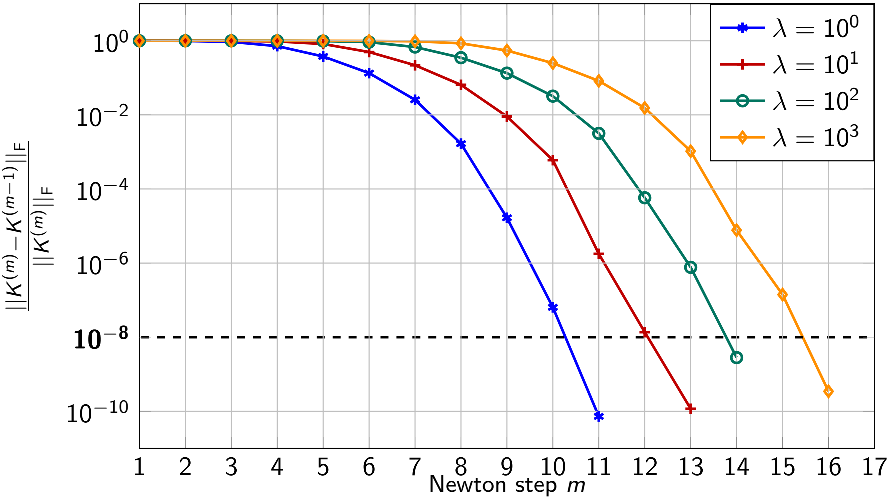
Convergence of Newton-ADI: NSE on von Kármán Vortex Street



Relative change of feedback matrix K for different Reynolds numbers ($\lambda = \rho = 1$, $n = 5468$, direct solver, $tol_{\mathrm{NM}} = 10^{-8}$, $\mathbf{tol_{ADI} = 10^{-4}}$).

Numerical Examples

Convergence of Newton-ADI: NSE Coupled with DCE in Reactor Model

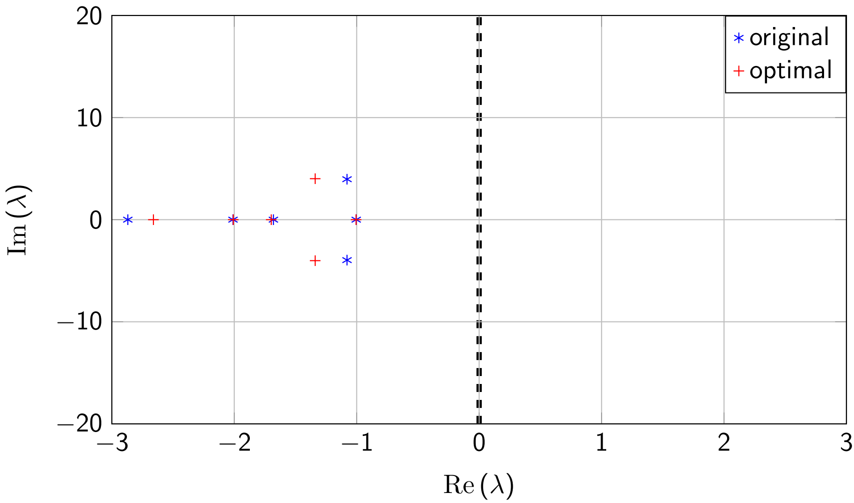


Relative change of feedback matrix K for different output weightings λ
(Re $= Sc = 10$, $\rho = 1$, $n = 6515$, direct solver, $tol_{\mathrm{NM}} = 10^{-8}$, $\mathbf{tol_{ADI} = 10^{-6}}$).

Numerical Examples

Eigenvalue Behavior of NSE on von Kármán Vortex Street

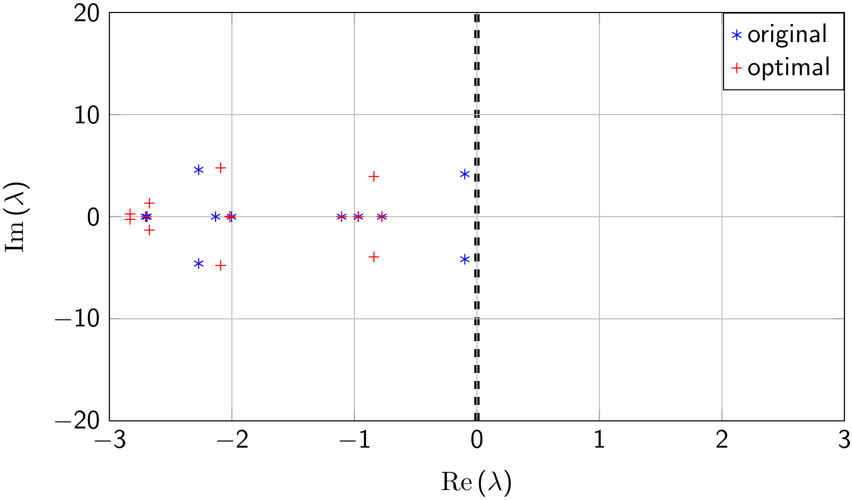
Zoom into eigenvalues for NSE pencil for: Re $= 100$



Numerical Examples

Eigenvalue Behavior of NSE on von Kármán Vortex Street

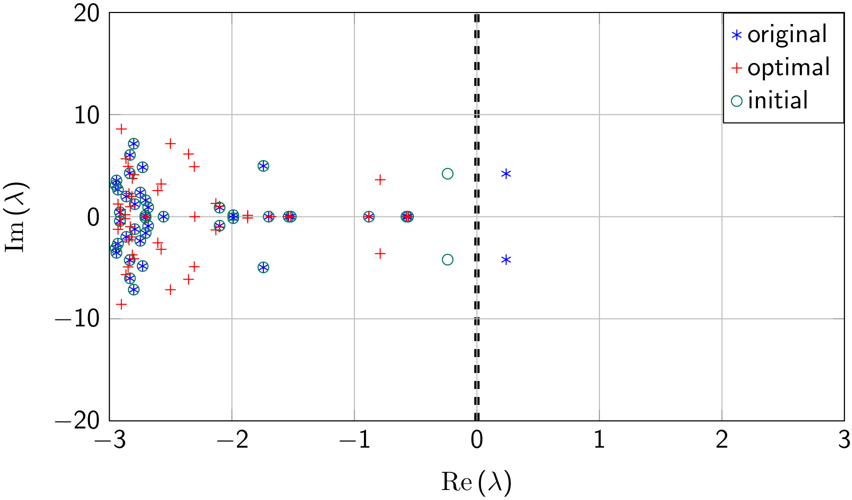
Zoom into eigenvalues for NSE pencil for: Re $= 200$



Numerical Examples

Eigenvalue Behavior of NSE on von Kármán Vortex Street

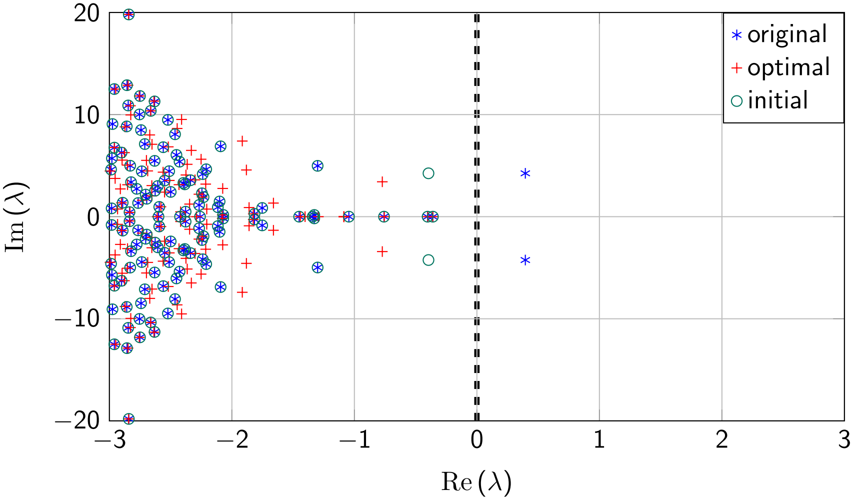
Zoom into eigenvalues for NSE pencil for: Re = 300



Numerical Examples

Eigenvalue Behavior of NSE on von Kármán Vortex Street

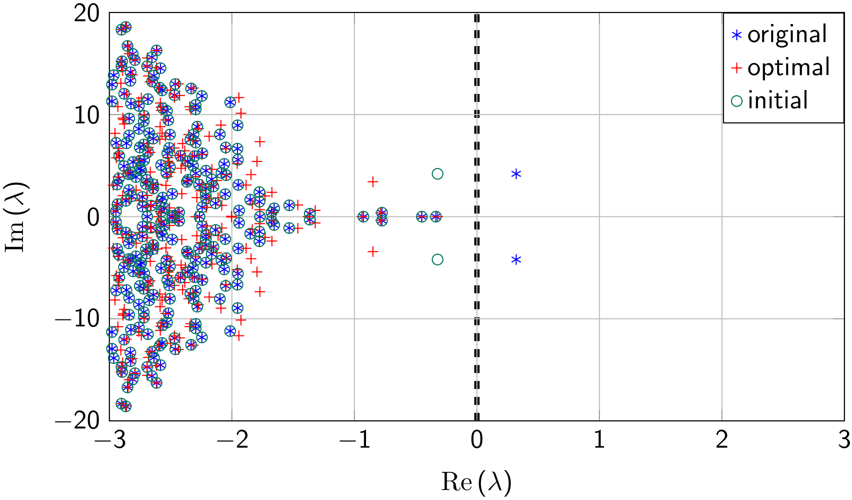
Zoom into eigenvalues for NSE pencil for: Re = 400



Numerical Examples

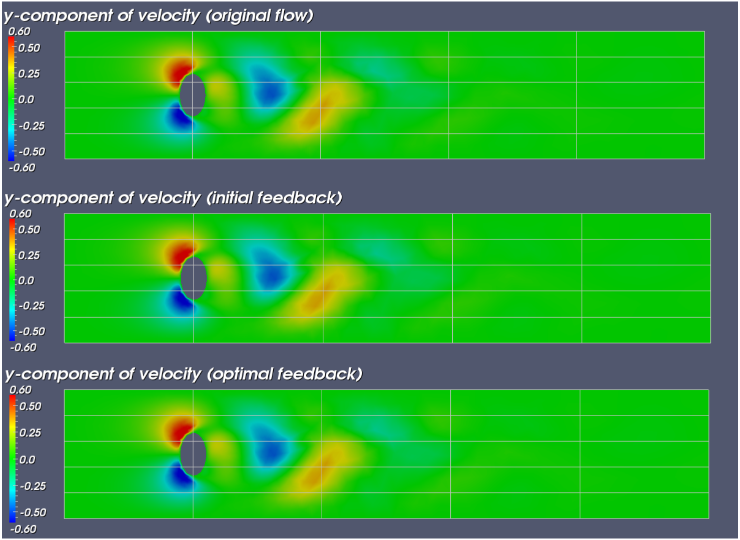
Eigenvalue Behavior of NSE on von Kármán Vortex Street

Zoom into eigenvalues for NSE pencil for: $\mathrm{Re} = 500$



Numerical Examples

Closed-Loop Simulation of NSE on von Kármán Vortex Street for Re = 300



Numerical Examples

Closed-Loop Simulation of NSE on von Kármán Vortex Street for Re = 300

Conclusion

Recent Progress

- Adapted Newton-ADI algorithm for flow problems (DAE structure).
- Closed-loop simulation for NSE.
- Applied method to coupled flow in reactor model.

Conclusion

Recent Progress

- Adapted Newton-ADI algorithm for flow problems (DAE structure).
- Closed-loop simulation for NSE.
- Applied method to coupled flow in reactor model.
- Modulated ADI shift determination for DAE structure.
- Improved input matrix B w.r.t. Raymond's analytic approach.
- Developed suitable output matrix C for reactor model.

Conclusion

Recent Progress

- Adapted Newton-ADI algorithm for flow problems (DAE structure).
- Closed-loop simulation for NSE.
- Applied method to coupled flow in reactor model.
- Modulated ADI shift determination for DAE structure.
- Improved input matrix B w.r.t. Raymond's analytic approach.
- Developed suitable output matrix C for reactor model.

Outlook

- Improve idea of *inexact Newton* to threefold nested iteration.
- Residual based stopping criteria for feedback computation.
- Closed-loop simulation of coupled flow in reactor model.
- Improve Krylov solver via the use of recycling or block techniques.

Conclusion

Recent Progress

- Adapted Newton-ADI algorithm for flow problems (DAE structure).
- Closed-loop simulation for NSE.
- Applied method to coupled flow in reactor model.
- Modulated ADI shift determination for DAE structure.
- Improved input matrix B w.r.t. Raymond's analytic approach.
- Developed suitable output matrix C for reactor model.

Outlook

- Improve idea of *inexact Newton* to threefold nested iteration.
- Residual based stopping criteria for feedback computation.
- Closed-loop simulation of coupled flow in reactor model.
- Improve Krylov solver via the use of recycling or block techniques.

Many thanks for your attention!

Literature

E. Bänsch and P. Benner, *Stabilization of Incompressible Flow Problems by Riccati-Based Feedback*, in Constrained Optimization and Optimal Control for Partial Differential Equations, G. Leugering and S. Engell et al., eds., vol. 160 of International Series of Numerical Mathematics, Birkhäuser, 2012, pp. 5–20.

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

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*, Preprint SPP1253-130, DFG-SPP1253, 2012.
Accepted for publication in SISC Copper Mountain Special Section 2012.

H. Elman, D. Silvester, and A. Wathen, *Finite Elements and Fast Iterative Solvers: with applications in incompressible fluid dynamics*, Oxford University Press, Oxford, 2005.

M. Heinkenschloss, D. C. Sorensen, and K. Sun, *Balanced truncation model reduction for a class of descriptor systems with application to the Oseen equations*, SIAM Journal on Scientific Computing, 30 (2008), pp. 1038–1063.

J. Raymond, *Feedback boundary stabilization of the two-dimensional Navier-Stokes equations*, SIAM Journal on Control and Optimization, 45 (2006), pp. 790–828.