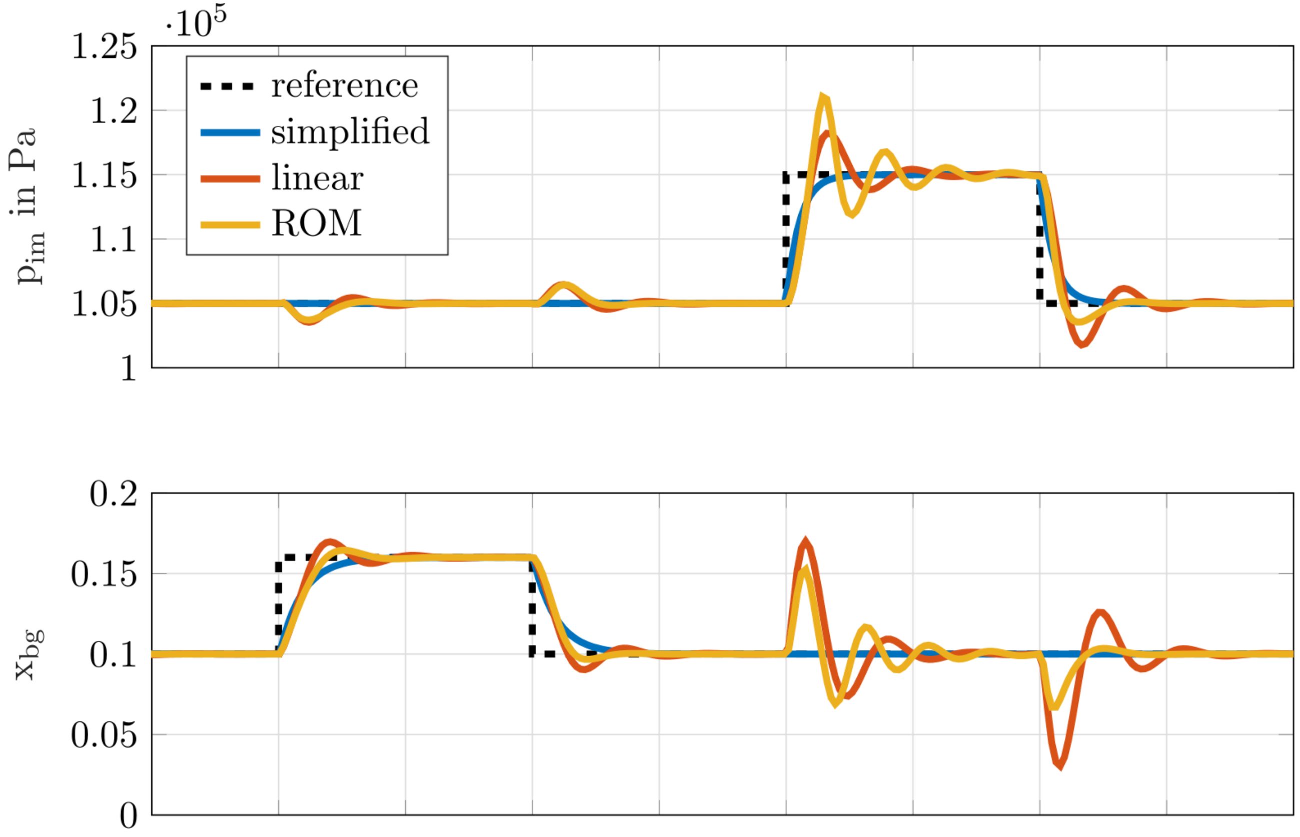
PS04 – Overview

- Review PS03
  - Classical controller design
- PS04 Tasks
  - Unconstrained linear MPC → find and implement analytical solution
    - Calculate the matrices $\mathbf{\Gamma}$, $\mathbf{\Psi}$, $\mathbf{\Upsilon}$, and $\mathbf{\Theta}$
    - Calculate MPC control law $\boldsymbol{K}\_\mathbf{MPC}$
    - Implement controller in Simulink and evaluate its performance

ETH zürich

Review PS03
Decentralized control



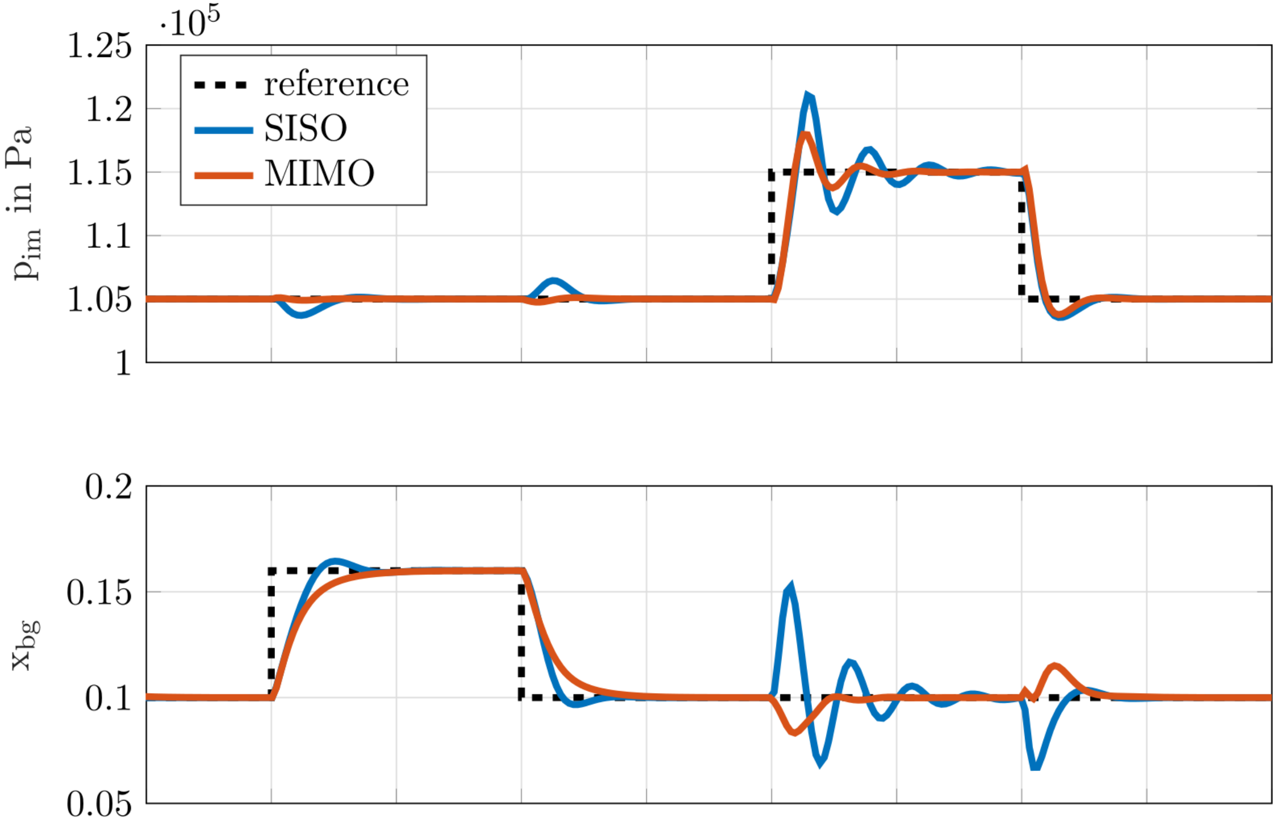
simplified model
- ➢ good control performance

linear model
- ➢ some oscillations

ROM
- ➢ pronounced oscillations

Review PS03

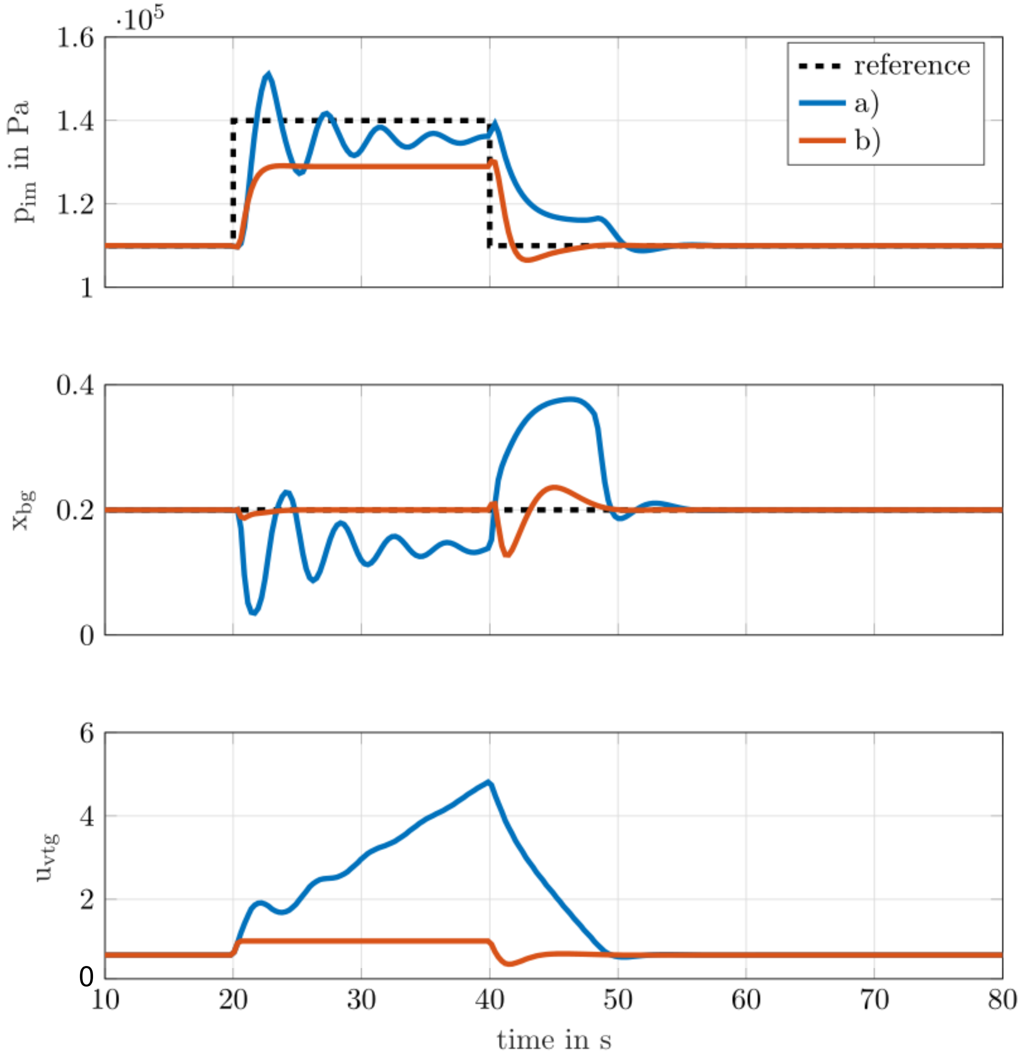
Decoupling



Evaluation on ROM

- even static decoupling significantly reduces oscillations

Review PS03
Anti-reset windup



a) without ARW

b) with ARW

➢ ARW deals with actuator saturations

Review PS03
Summary

- Classical control can be tuned and extended to work reasonably, even for challenging control problems.
- Controller design and tuning can become challenging and requires a lot of experience.

PS04
Goal

- Implement an unconstrained linear model predictive controller
- Build up controller in 3 steps
  - Calculate the matrices $\mathbf{\Gamma}, \mathbf{\Psi}, \mathbf{\Upsilon}$, and $\mathbf{\Theta}$
  - Calculate MPC control law $\boldsymbol{K}_{\mathbf{MPC}}$
  - Implement controller in Simulink and evaluate its performance

Exercise 1

Calculate the matrices $\boldsymbol{\Gamma}$, $\boldsymbol{\Psi}$, $\boldsymbol{\Upsilon}$, and $\boldsymbol{\Theta}$

$$\boldsymbol{\Gamma} = \begin{bmatrix} \boldsymbol{C} & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \boldsymbol{C} \end{bmatrix} \in \mathbb{R}^{m\cdot(N_2-N_1+1)\,\times\, n\cdot(N_2-N_1+1)}$$

$$\boldsymbol{\Psi} = \begin{bmatrix} \boldsymbol{A}^{N_1} \\ \vdots \\ \boldsymbol{A}^{N_2} \end{bmatrix} \in \mathbb{R}^{n\cdot(N_2-N_1+1)\,\times\, n}$$

$$\boldsymbol{\Upsilon} = \begin{bmatrix} \sum_{i=0}^{N_1-1} \boldsymbol{A}^i\,\boldsymbol{B} \\ \vdots \\ \sum_{i=0}^{N_2-1} \boldsymbol{A}^i\,\mathbf{B} \end{bmatrix} \in \mathbb{R}^{n\cdot(N_2-N_1+1)\,\times\, l}$$

$$\boldsymbol{\Theta} = \begin{bmatrix} \boldsymbol{\Lambda}(N_1) & \boldsymbol{\Lambda}(N_1-1) & \dots & \boldsymbol{\Lambda}(N_1-N_u+1) \\ \boldsymbol{\Lambda}(N_1+1) & \boldsymbol{\Lambda}(N_1) & \dots & \boldsymbol{\Lambda}(N_1-N_u+2) \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{\Lambda}(N_2) & \boldsymbol{\Lambda}(N_2-1) & \dots & \boldsymbol{\Lambda}(N_2-N_u+1) \end{bmatrix} \in \mathbb{R}^{n\cdot(N_2-N_1+1)\,\times\, l\cdot N_u}$$

$$\boldsymbol{\Lambda}(i) = \begin{cases} \sum_{j=0}^{i-1} \boldsymbol{A}^j\,\boldsymbol{B} & , i \geq 1 \\ \mathbf{0} & , i < 1 \end{cases} \in \mathbb{R}^{n\,\times\, l}$$

$$\begin{aligned} \boldsymbol{\Delta u}(\cdot|k)^* &= \left(\boldsymbol{\Theta}^T\boldsymbol{\Gamma}^T\boldsymbol{Q}\boldsymbol{\Gamma}\boldsymbol{\Theta} + \boldsymbol{R}\right)^{-1}\boldsymbol{\Theta}^T\boldsymbol{\Gamma}^T\boldsymbol{Q}\,\boldsymbol{e}\,(\cdot|k)\;, \\ \boldsymbol{e}\,(\cdot|k) &= \boldsymbol{r}(\cdot|k) - \boldsymbol{\Gamma}\Big(\boldsymbol{\Psi}\,\boldsymbol{x}(k) + \boldsymbol{\Upsilon}\,\boldsymbol{u}(k-1)\Big)\,. \end{aligned}$$

- Unconstrained linear MPC:
  MPC control law can be calculated explicitly (see lecture slides)

- Discretize system and calculate system response matrices $\boldsymbol{\Gamma}$, $\boldsymbol{\Psi}$, $\boldsymbol{\Upsilon}$, and $\boldsymbol{\Theta}$

Exercise 2

Calculate MPC control law ($\boldsymbol{K_{MPC}}$)

$$\boldsymbol{Q} = \begin{bmatrix} \boldsymbol{Q}_{N_1} & & \boldsymbol{0} \\ & \ddots & \\ \boldsymbol{0} & & \boldsymbol{Q}_{N_2} \end{bmatrix} \in \mathbb{R}^{m\cdot(N_2-N_1+1)\,\times\, m\cdot(N_2-N_1+1)}$$

$$\boldsymbol{R} = \begin{bmatrix} \boldsymbol{R}_0 & & \boldsymbol{0} \\ & \ddots & \\ \boldsymbol{0} & & \boldsymbol{R}_{N_u-1} \end{bmatrix} \in \mathbb{R}^{l\cdot N_u\,\times\, l\cdot N_u}$$

$$\boldsymbol{K}_{\mathrm{MPC}} = [\boldsymbol{I}\ \boldsymbol{0}\ \ldots \boldsymbol{0}]\left(\boldsymbol{\Theta}^T\boldsymbol{\Gamma}^T\boldsymbol{Q}\boldsymbol{\Gamma}\boldsymbol{\Theta} + \boldsymbol{R}\right)^{-1}\boldsymbol{\Theta}^T\boldsymbol{\Gamma}^T\boldsymbol{Q}$$

$$\boldsymbol{u}(k)^* = \boldsymbol{u}(k-1) + \boldsymbol{K}_{\mathrm{MPC}}\,\boldsymbol{e}(\cdot|k)$$

- Set weighting matrices $\boldsymbol{Q}_\mathrm{i}$ and $\boldsymbol{R}_\mathrm{i}$
- Calculate MPC control law $\boldsymbol{K}_{\mathrm{MPC}}$

Exercise 3

Implement controller in Simulink

- Test the developed MPC controller using the
  - linearized model
  - ROM
- Analyze performance

PS04
Provided files

PS04

- **setupPredictionMatrices.m** – Template for function to calculate system response matrices (finalize)
- **ps04_run_ex3.m** – Script to run simulation (no adaptions needed)
- **ps04_ex3_LinearModel** – model to test controller using linearized model (finalize controller implementation)
- **ps04_ex3_ROM** – model to test controller using ROM (copy controller from ps04_ex3_LinearModel.slx)