KTH ROYAL INSTITUTE
OF TECHNOLOGY

Introduction to Model Order Reduction

Lecture 1: Introduction and overview

Henrik Sandberg, Bart Besselink, Madhu N. Belur

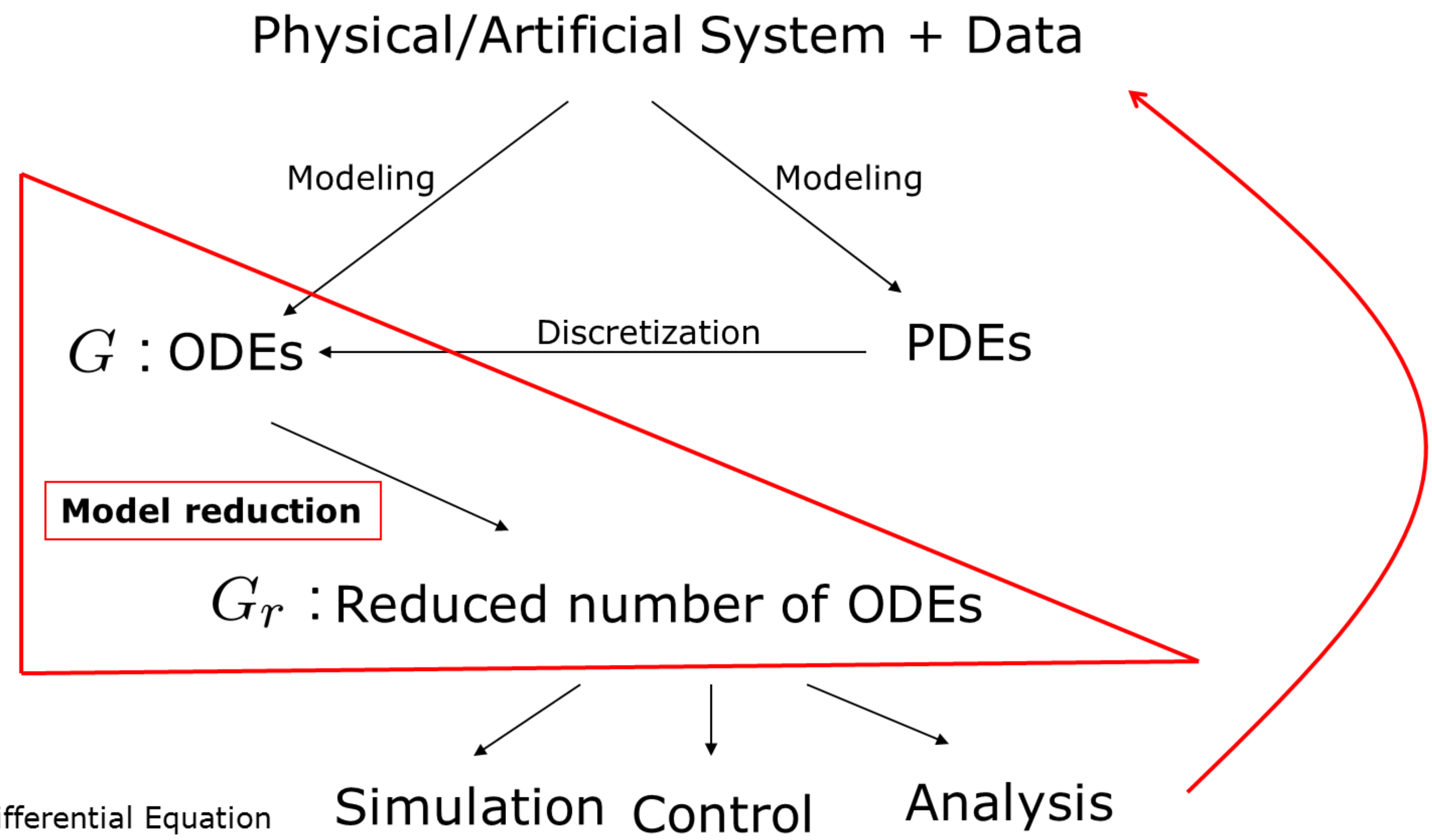
Overview of Today’s Lecture

- What is model (order) reduction? Why is it important?
- What is included in the course? What is **not** included?
- Preliminary program
- What is expected from you? How to pass?
- Sign up for course

Model (Order) Reduction

- **~59 000 000** hits in Google…
- Many different research communities use different forms of model reduction:
  - Fluid dynamics
  - Mechanics
  - Computational biology
  - Circuit design
  - <u>Control theory</u>
  - …
- **Many** heuristics available. More or less well-motivated.
- In early 1980’s some optimal approaches were developed (using AAK-lemma) in control theory.
- Few rigorous methods known for nonlinear systems.

The Big Picture



ODE = Ordinary Differential Equation
PDE = Partial Differential Equation

An Incomplete Problem Formulation

Given an ODE of order *n*

$$G: \quad \dot{x}(t) = f(x(t)), \quad x(t) \in \mathbb{R}^n$$

Find another ODE of order *r*

$$G_r: \quad \dot{z}(t) = f_r(z(t)), \quad z(t) \in \mathbb{R}^r, \, r \ll n$$

with “essentially” the same “properties”.

Not enough information for problem to make complete sense, although this captures the essence of the model-order-reduction problem.

Problem 1: "The standard problem"

Given:

$$G : \begin{cases} \dot{x}(t) = f(x(t), u(t)), \quad x(t) \in \mathbb{R}^n,\ u \in \mathcal{U} \\ y(t) = g(x(t), u(t)) \end{cases}$$

Find:

$$G_r : \begin{cases} \dot{z}(t) = f_r(z(t), u(t)), \quad z(t) \in \mathbb{R}^r,\ u \in \mathcal{U} \\ y_r(t) = g_r(z(t), u(t)) \end{cases}$$

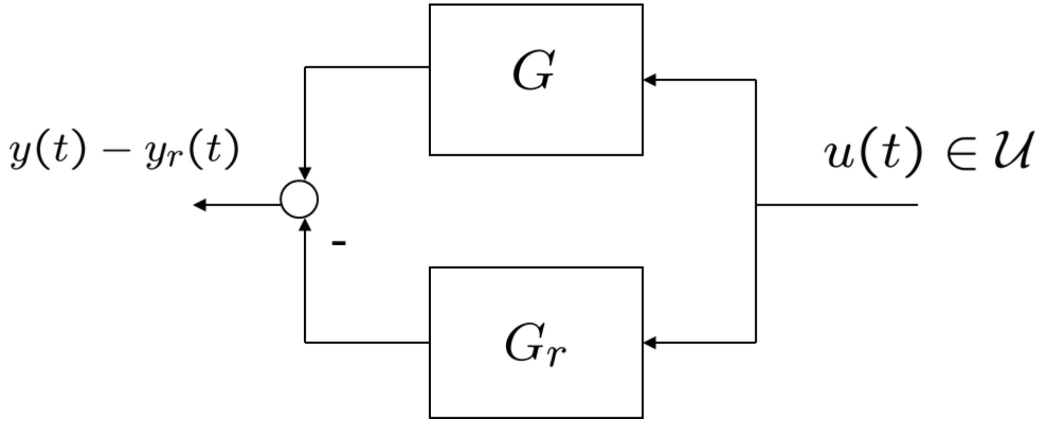
such that

$$\mathrm{misfit}(G, G_r) = \sup_{u \in \mathcal{U}} \frac{\|y - y_r\|}{\|u\|}$$

is small.

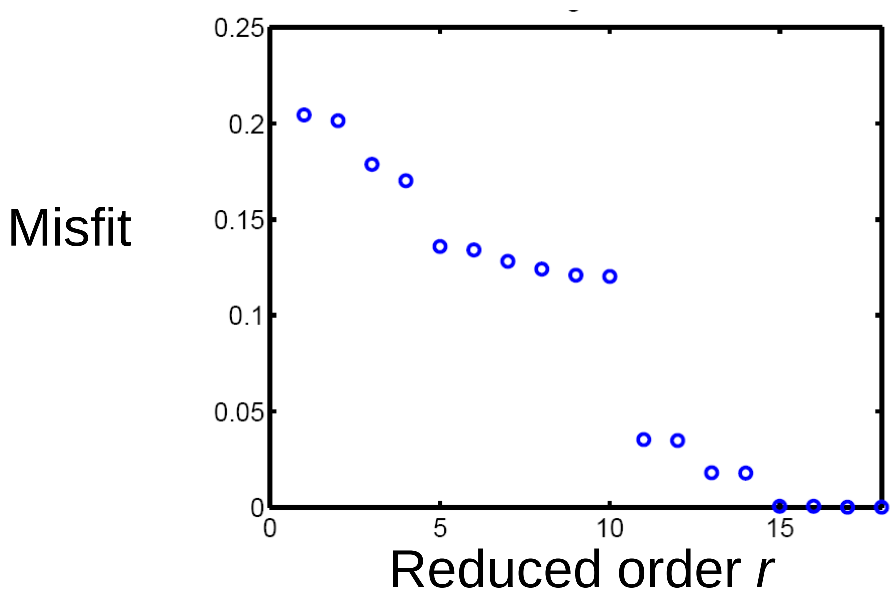
Problem 1 (cont'd)

- Choice of input *u(t)* determines what states are excited. Could also reflect initial conditions *x(0)*.
- Choice of output *y(t)* determines what property of the states we want to preserve.



$$\|y - y_r\| \leq \mathrm{misfit}(G, G_r) \cdot \|u\|, \quad \forall u \in \mathcal{U}$$

Problem 1 (cont'd): Trade-off



- The trade-off curve determines suitable G_r
- Acceptable misfit → Suitable reduced orders
- Expensive to compute exact curve. Bounds often enough:

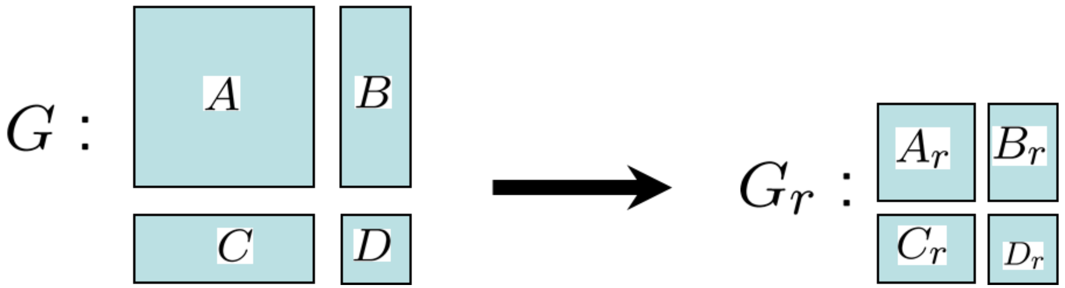
$$\underline{\text{bound}}(r) \leq \text{misfit}(G, G_r) = \sup_{u \in \mathcal{U}} \frac{\|y - y_r\|}{\|u\|} \leq \overline{\text{bound}}(r)$$

Problem 1 (cont'd)

Often the linear problem will be treated:

$$G : \begin{cases} \dot{x}(t) = Ax(t) + Bu(t), & x(t) \in \mathbb{R}^n,\ u \in L_2[0,\infty) \\ y(t) = Cx(t) + Du(t) \end{cases}$$

$$G_r : \begin{cases} \dot{z}(t) = A_r z(t) + B_r u(t), & z(t) \in \mathbb{R}^r,\ u \in L_2[0,\infty) \\ y_r(t) = C_r z(t) + D_r u(t) \end{cases}$$



Problem 1 (cont'd)

A good model-reduction method gives us:

1. bound(*r*) – To help us choose a suitable approximation order *r* **before** the reduced-order model has to be computed; and

2. a reduced-order model (f_r, g_r) alt. (A_r, B_r, C_r, D_r).

Such methods exist for some classes of models (typically linear). Many heuristics fail to provide bound(*r*).

Note: After a reduced-order G_r model is found, usually misfit(G, G_r) can be computed (although it may be expensive)

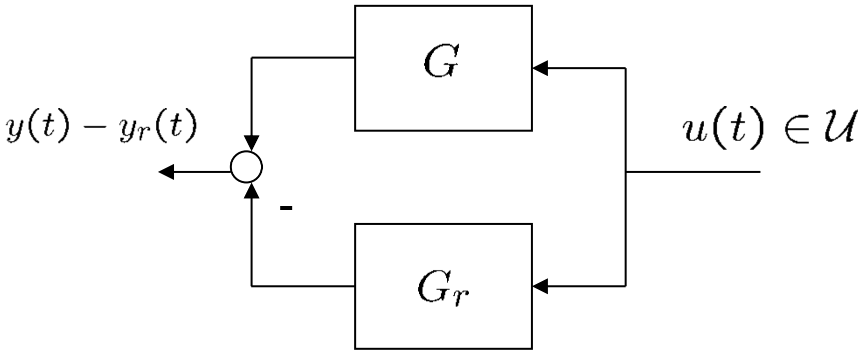
Why Decrease the Order?

- **Simulation:** Each evaluation of $f(x(t),u(t))$ is $O(n^2)$ operations in linear case.
- **Simulation:** Data compression, roughly $O(n^2)$ numbers to store a linear model.

- **Control:** Computation time of LQG controller is $O(n^3)$ operations (solve the Riccati equation).
- **Control:** Optimal controller is at least of order $n \Rightarrow$ can be hard to implement.

- **Analysis:** Curse of dimensionality. Problem complexity often exponential in number of equations (=order).

Why Define Misfit This Way?

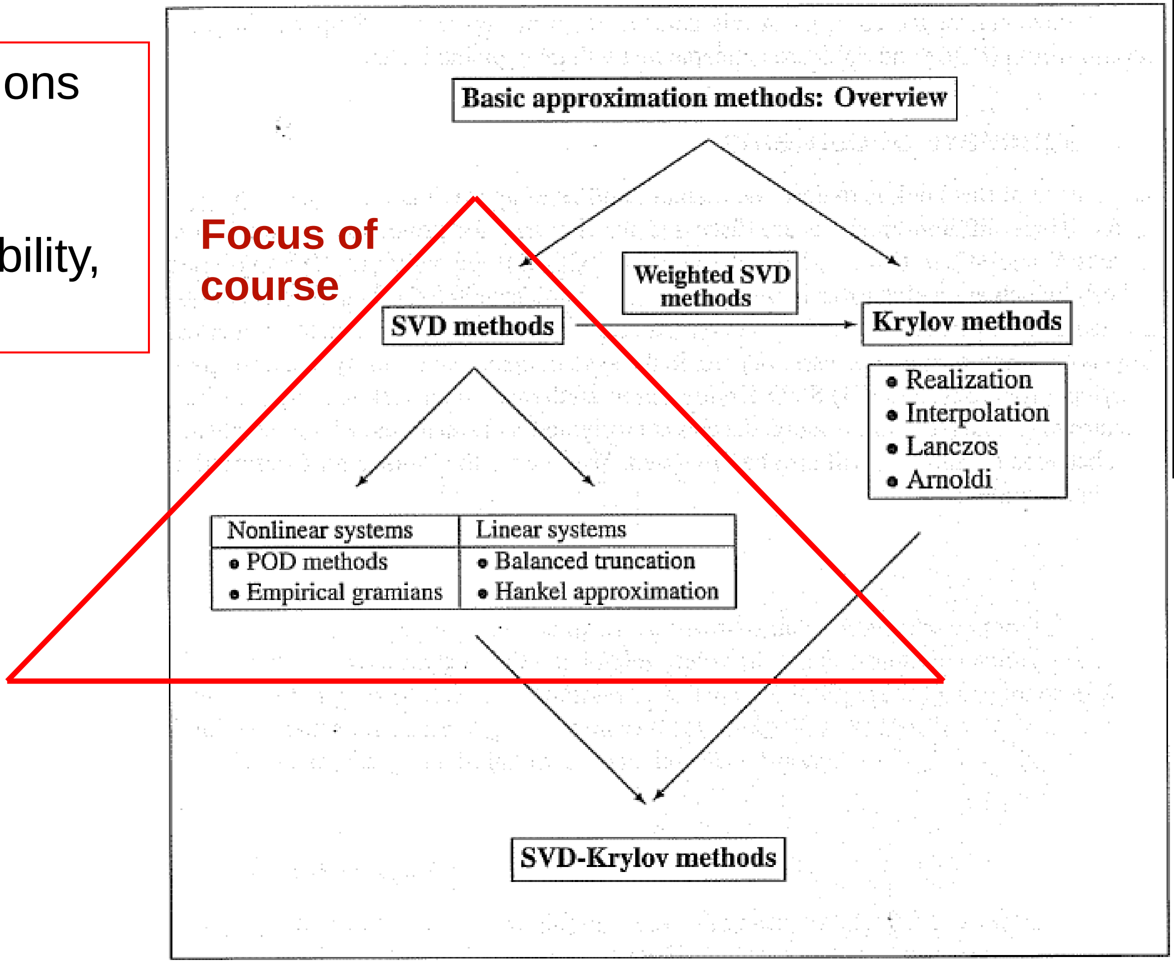


$$\text{misfit}(G, G_r) = \sup_{u \in \mathcal{U}} \frac{\|y - y_r\|}{\|u\|}$$

- Misfit is a measure of the worst-case error of the approximation. Can be pessimistic…
- Other measures are possible, statistical for example.
- Worst-case error often good for control theory (robust control theory).
- Simple expressions of bound(*r*) are available for worst-case errors, but not for statistical error measures.

Classification of Methods

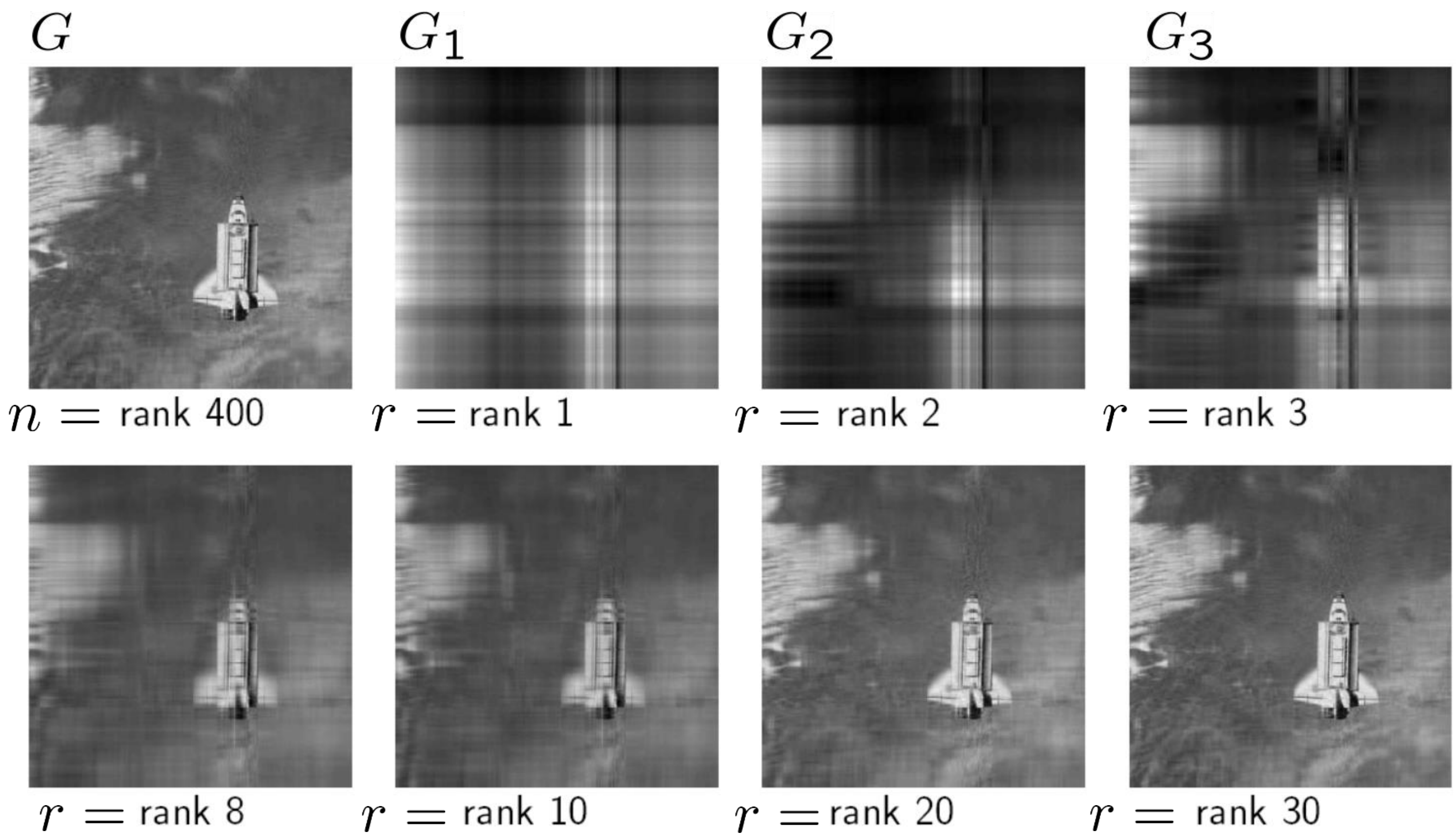
- $O(n^3)$ operations
- *n ~ 1000*
- Provable stability, bound(*r*) exist



- $O(r^2n)$ operations
- *n ~ 10⁶*
- No provable stability, bound(*r*) does not exist

[Figure from *Approximation of Large-Scale Dynamical Systems*]

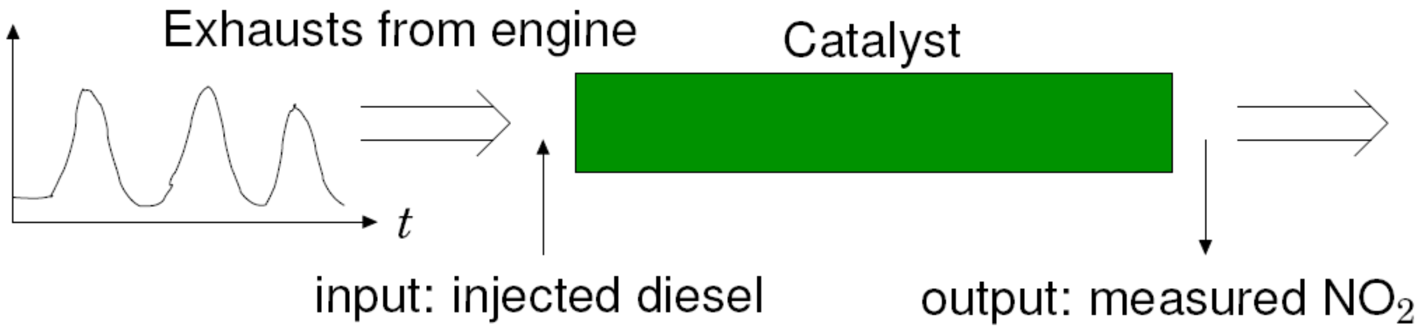
Example 1: Image compression



$$\mathrm{misfit}(G, G_r) = \sigma_{r+1} \quad (\sigma_i = \text{singular values of } G)$$

Example 2: Chemical Reactions

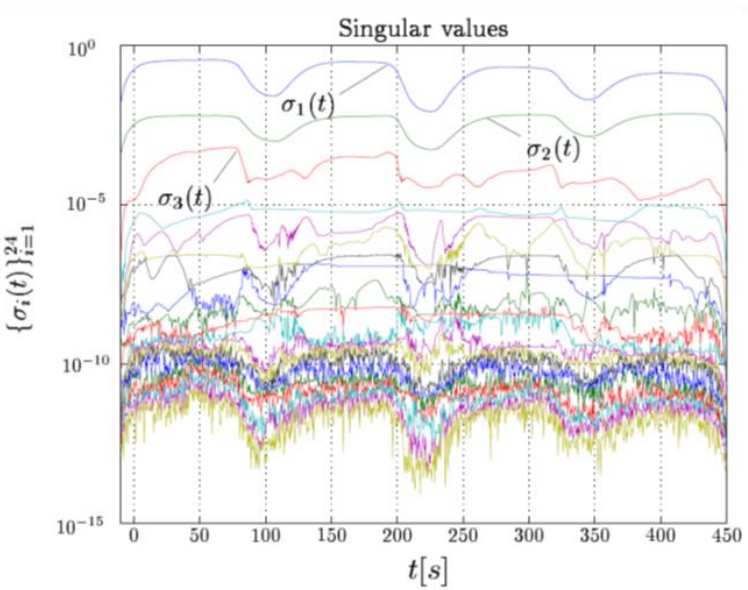
Model reduction of a diesel exhaust catalyst from [Sandberg, 2006].

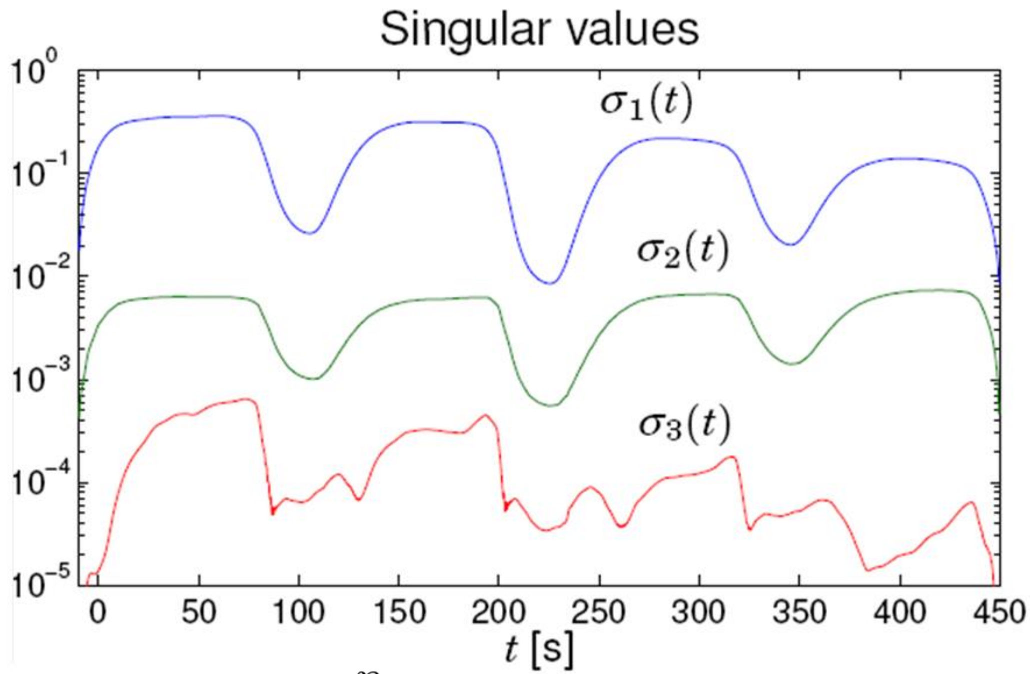


- Reduction of NO_x
- Model by Westerberg et al. ('02)
- 24 nonlinear ODEs. Linearize around pulsating trajectory $\rightarrow \{A(t), B(t), C(t)\}$

Example 2 (cont'd)

Reduced-order vs. misfit trade-off

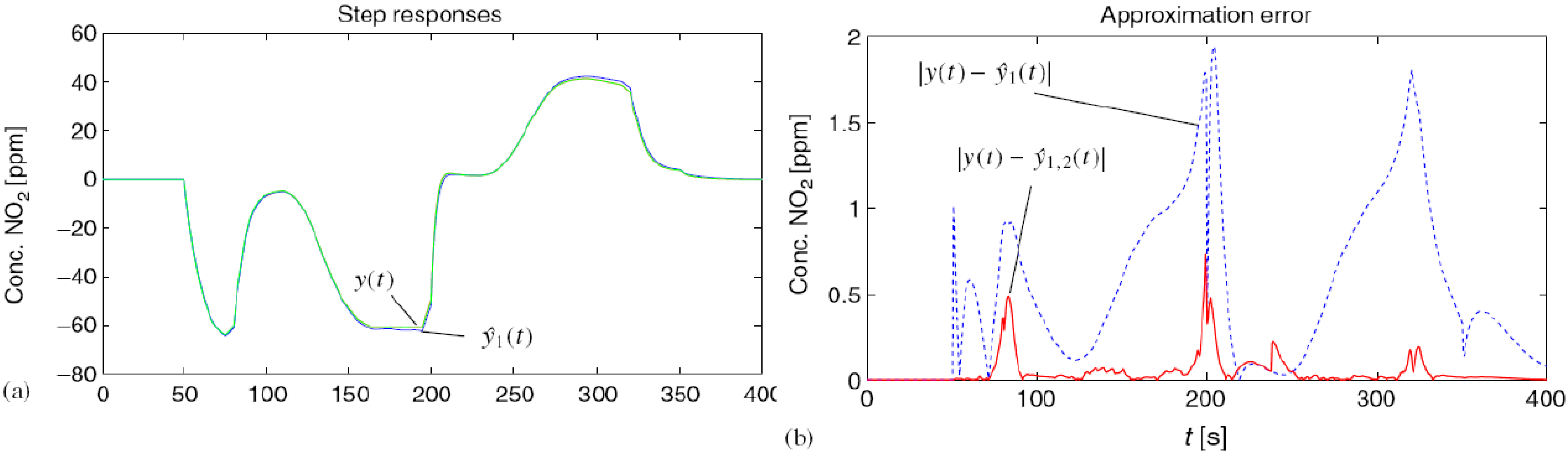




$$\text{misfit}(G, G_r) \leq \overline{\text{bound}}(r) = \sum_{i=r+1}^{n} \text{function}(\sigma_i(t))$$

Example 2 (cont'd)

Verification using *r=1* and *r=2*



Explanation

Kalman decomposition:

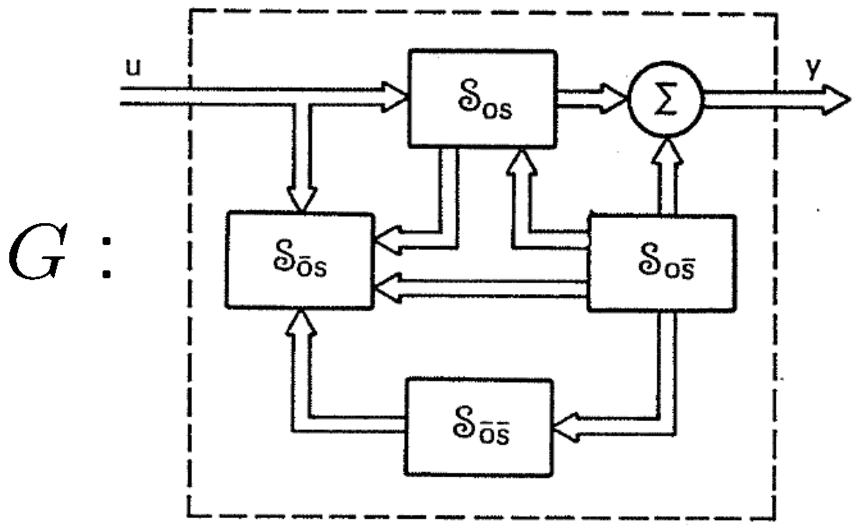


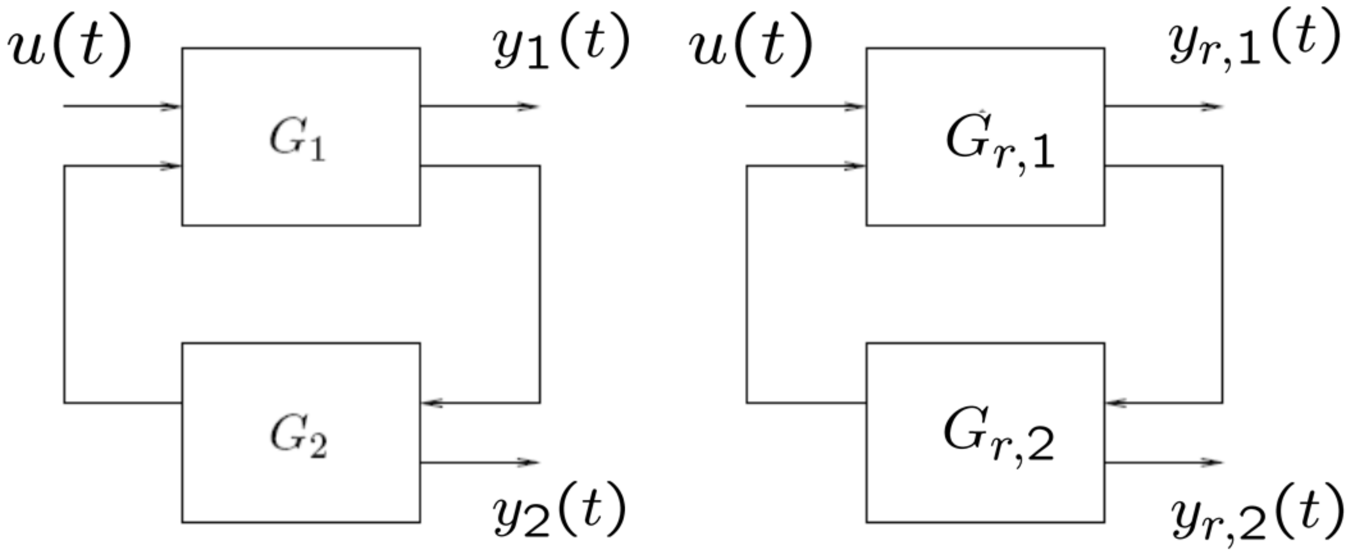
Fig. 6.2. Blockschema som illustrerar Kalmans uppdelning av ett godtyckligt system i delsystemen $\mathcal{S}_{os}$, $\mathcal{S}_{o\bar{s}}$, $\mathcal{S}_{\bar{o}s}$, $\mathcal{S}_{\bar{o}\bar{s}}$.

Only S_{os} contribute to the mapping $u(t) \rightarrow y(t)$.

Also, states in S_{os} do not contribute equally.

$G_r = S_{os}$ is one obvious reduced model candidate, but we can often reduce more with very small misfit!

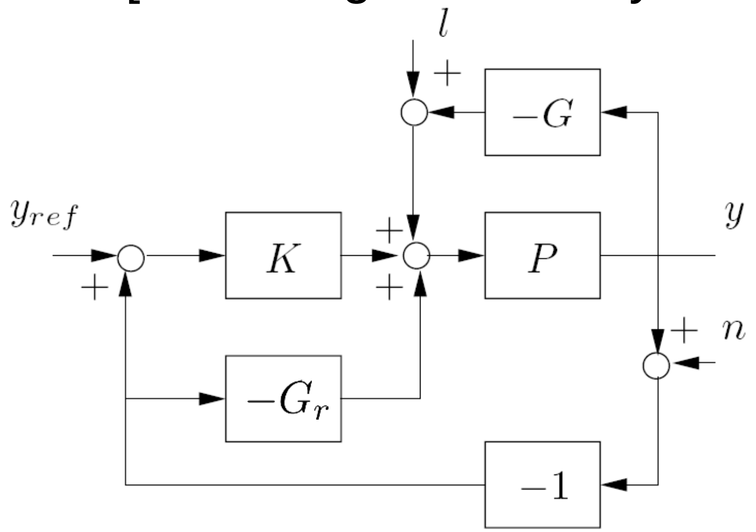
Problem 2: Model Reduction with Structure Constraints



- States in the model *G* are physically constrained to certain blocks, for example.
- Example: G_1 is a plant. G_2 is a controller.

Example 3: Networked Control

Example from [Sandberg and Murray, 2007].

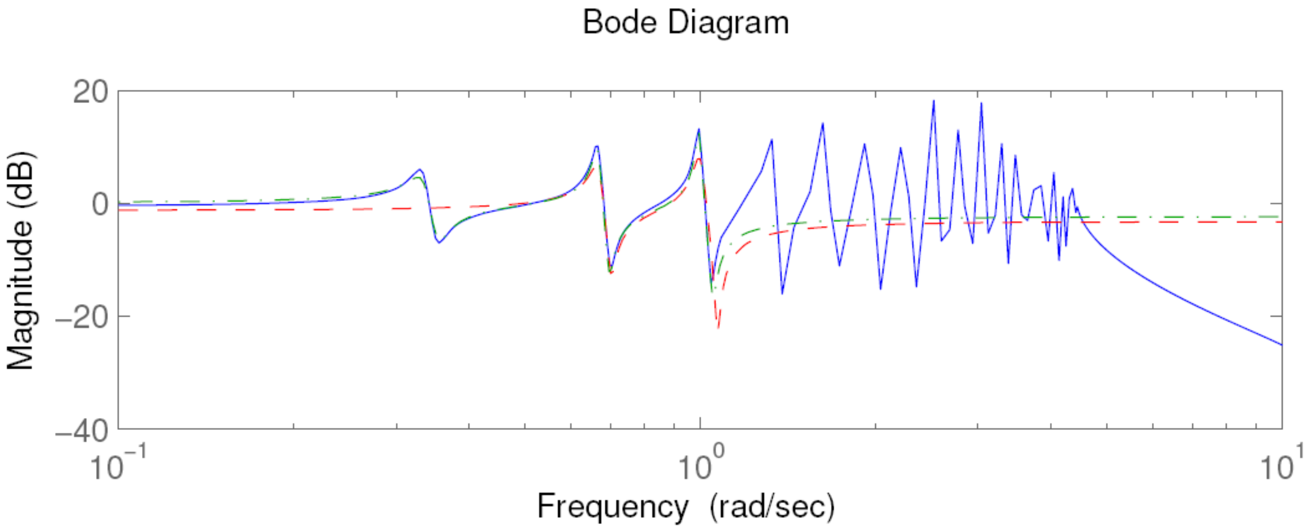


K is a decentralized controller of *P*.

G models *P*'s interaction with the (large) surrounding environment.

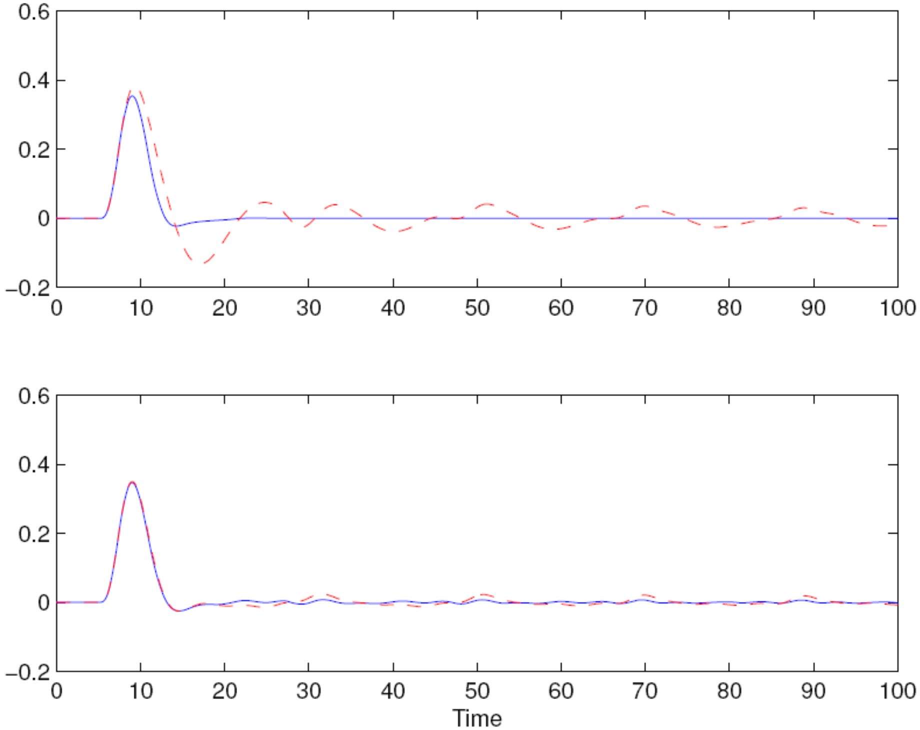
G_r is a local environment model, to be added to controller *K*. How to choose G_r?

Example 3 (cont'd)



- Environment *G* (solid blue) is a highly resonant system.
- In open loop, *G* is hard to reduce. In closed loop, only certain frequencies are important.
- Reduced models: G_4 (dashed red), G_6 (dashed green).

Example 3 (cont'd)

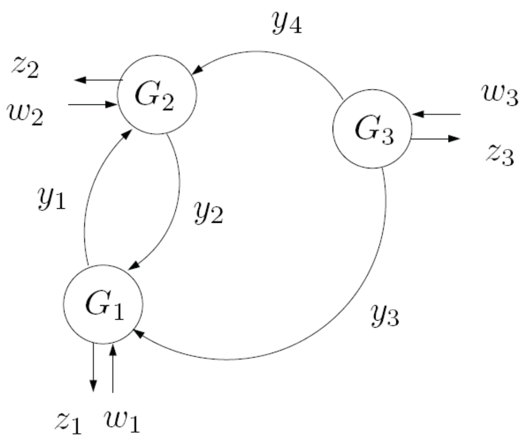


- Upper plot: Load step response with/without $G_r = G$.
- Lower plot: Load step response with G_4 and G_6.
- A low-order environment model can compensate for a very complex environment!

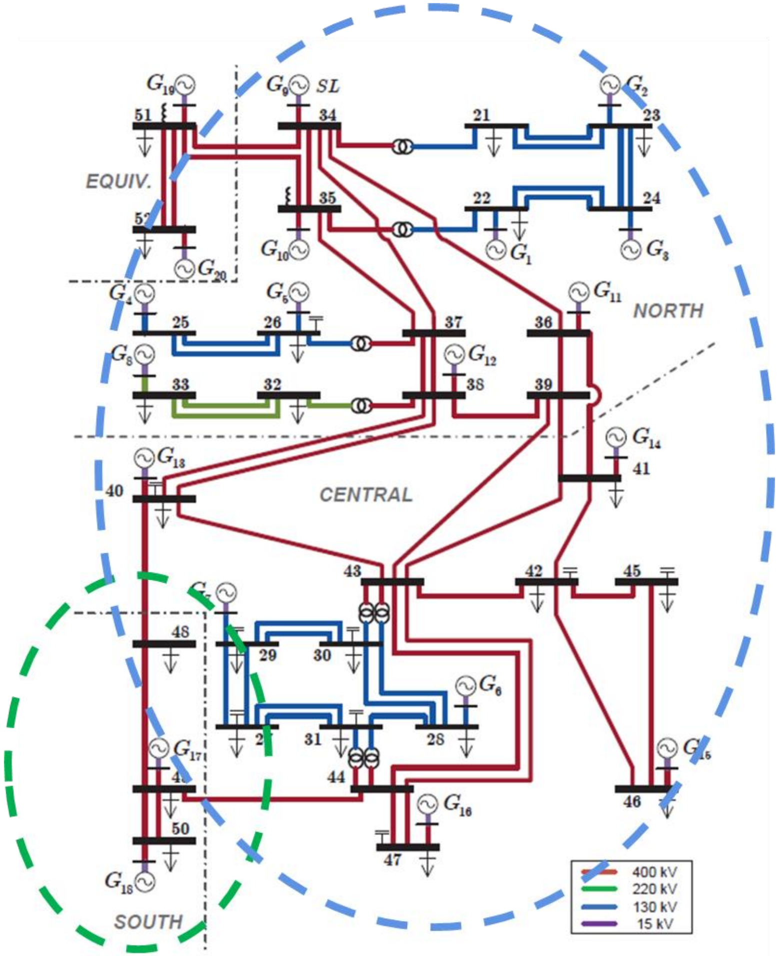
Explanation

- Find proper “inputs” and “outputs” to each subsystem, which reflect the subsystem’s interaction with the global system.
- Then apply methods that solve Problem 1.

- Motivation:
  1. Low-order feedback/feedforward controllers
  2. Large interconnected systems in computer science and biology
  3. Modular model reduction



Example 4: Power Systems, KTH-Nordic32 system



Model info:

- 52 buses
- 52 lines
- 28 transformers
- 20 generators (12 hydro gen.)

Study area: Southern Sweden. Keep this model

External area: Simplify as much as possible

Example from [Sturk, Vanfretti, Chompoobutrgool, Sandberg, 2012, 2014].

Results

- External area has 246 dynamic states.
- *Reduced* external area has 17 dynamic states
- Evaluation on faulty interconnected system:

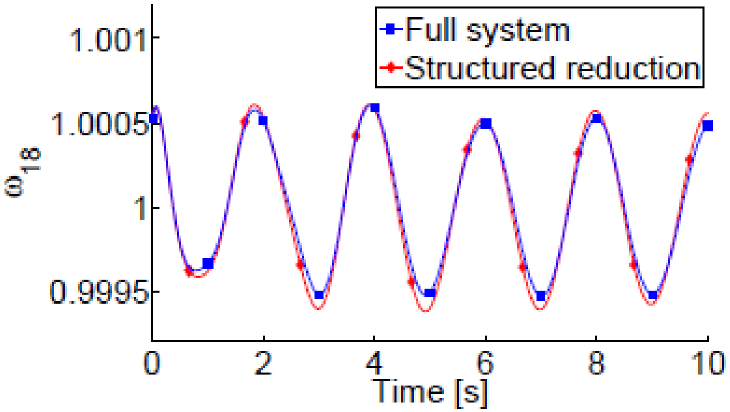


Figure 4.15: Transient of ω_{18} after a 10 ms fault at bus 18.

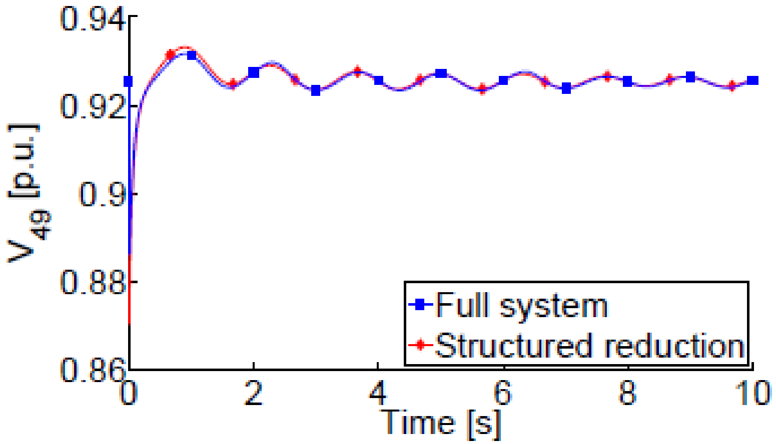


Figure 4.16: Transient of V_{49} after a 10 ms fault at bus 18.

What You Will Learn in the Course

- Norms of signals and systems, some Hilbert space theory.
- Principal Component Analysis (PCA)/Proper Orthogonal Decomposition (POD)/Singular Value Decomposition (SVD).
- Realization theory: Observability and controllability from optimal control/estimation perspective.
- Balanced truncation for linear systems (with extension to nonlinear systems).
- Hankel norm approximation.
- Uncertainty and robustness analysis of models (small-gain theorem), controller reduction.
- Optimization/LMI approaches.
- Behavioral theory (Madhu Belur).

Course Basics

- Graduate level
- Pass/fail
- 7 ECTS
- Course code: ~~FEL3500~~
- Prerequisites:
  1. Linear algebra
  2. Basic systems theory (state-space models, controllability, observability etc.)
  3. Familiarity with MATLAB

Course Material

Two books entirely devoted to model reduction are available:

1. Obinata and Anderson: *Model Reduction for Control Systems Design (online version)*
2. Antoulas: *Approximation of Large-Scale Dynamical Systems*

These books are **not** required for the course (although they are very good). Complete references on webpage.

Parts of these control/optimization books are used

1. Luenberger: Optimization by Vector Space Methods
2. Green and Limebeer: *Linear Robust Control (online version)*
3. Doyle, Francis, and Tannenbaum: *Feedback Control Theory (online version)*

Course Material (cont'd)

- Relevant research articles will be distributed.
- Generally no slides. White/black board will be used.
- Minimalistic lecture notes (PDFs) provided every lecture, containing:
  1. Summary of most important results (generally without proofs)
  2. Exercises
  3. Reading advice

To Get Credits, You Need to Complete…

1. Exercises
   - Exercises handed out with each lecture
   - At the end of the course, at least 75% of the exercises should have been solved and turned in on time
   - Exercises for Lectures 1-4 due **April 17**
   - Exercises for Lectures 5-8 due **May 12**

2. Exam
   - A 24h take-home exam
   - You decide when to take it, but it should be completed at the **latest 3 months** after course ends
   - No cooperation allowed
   - Problems similar to exercises

Next Lecture

- Wednesday April 2 at 13:15-15 in L41.
- We start with the simplest methods:
  - Modal truncation
  - Singular perturbation/residualization
  - Model projection
- First set of exercises handed out.
- Model-reduction method complexity increases with time in the course.
- First exercise session on Friday April 4 is devoted to repetition of basic linear systems concepts, Hilbert spaces, norms, operators,…

- **Hope to see you on Wednesday!**