Modern Nonlinear Control

Lecture 14

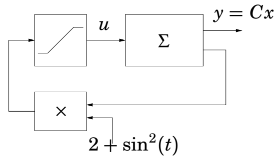
- Observers cont'd

- Lie-brackets and nonlinear controllability
- The parking problem
- Integral Quadratic Constraints
- SOS-tools
- Hybrid control — Piecewise linear systems
- Adaptive control
- Information about master theses projects
- How to find more information

Motivation: Simple example

Consider the following simple feedback system

$$\begin{cases} \begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix}0 & 1\\ -3 & 1\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} + \begin{bmatrix}0\\ -1\end{bmatrix} u = Ax + Bu \qquad (\Sigma) \\ y = \begin{bmatrix}1 & 0\end{bmatrix} x = Cx \\ u = sat(x_2 \cdot (2+\sin^2(t))) \end{cases}$$



Example cont'd

- linear subsystem unstable
- input saturation $\Rightarrow$ *At best local stability.*

——————— Tools ———————

Locally valid Quadratic Contraint (QC) (sector condition)

$$0 \le (\kappa_2 \cdot x_2 - u)(u - \kappa_1 \cdot x_2) =$$

$$\begin{bmatrix}x_1 & x_2 & u\end{bmatrix} \begin{bmatrix} \begin{pmatrix}0 & 0\\ 0 & -3\end{pmatrix} & \begin{pmatrix}0\\ 2\end{pmatrix} \\ \begin{pmatrix}0 & 2\end{pmatrix} & -1 \end{bmatrix} \begin{bmatrix}x_1\\ x_2\\ u\end{bmatrix} \quad \text{for some } |x_2| < c$$

$\kappa_1 = 1$ Lower bound :
'linear feedback stability cond.'
$u = \kappa x_2, \kappa \in (1, \infty)$

$\kappa_2 = 3$ Upper bound :
sector of nonlinearity

Preliminaries

State feedback

$$\begin{cases} \dot{x} = Ax + Bu = Ax + B\phi(x) \\ y = Cx \\ u = \phi(x) \end{cases}$$

Observer feedback

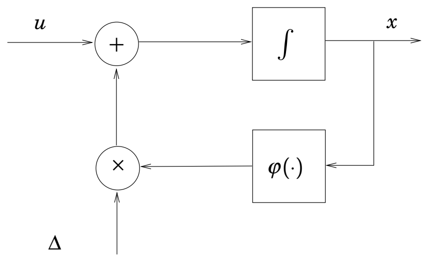
$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \\ \dot{\hat{x}} = A\hat{x} + Bu + L(y - C\hat{x}) \\ u = \phi(\hat{x}) \end{cases}$$

Asymptotically stable for state feedback $u = \phi(x)$

Re-write with *error dynamics* ($e = \hat{x} - x$)

$$\begin{cases} \dot{e} = (A - LC)e \\ \dot{x} = Ax + B\phi(x+e) + LCe \\ u = \phi(\hat{x}) \end{cases}$$

Example: Matched uncertainty



$$\dot{x} = u + \varphi(x)\Delta(t)$$

Example *cont.*

Example:

Exponentially decaying disturbance $\Delta(t) = \Delta(0)e^{-kt}$
linear feedback $u = -cx, \quad c > 0$
$\varphi(x) = x^2$

$$\dot{x} = -cx + \Delta(0)e^{-kt}x^2$$

Similar to peaking problem in the first lecture: Finite escape of solution to infinity if $\Delta(0)x(0) > c + k$

We want to guarantee that x(t) stay bounded for all initial values $x(0)$ and all bounded disturbances $\Delta(t)$

Nonlinear damping

Modify the control law in the previous example as:

$$u = -cx - s(x)x$$

where

$$-s(x)x$$

will be denoted *nonlinear damping.*

Use the Lyapunov function candidate $V = \dfrac{x^2}{2}$

$$\begin{aligned} \dot{V} &= xu + x\varphi(x)\Delta \\ &= -cx^2 - x^2 s(x) + x\varphi(x)\Delta \end{aligned}$$

How to proceed?

Choose

$$s(x) = \kappa\varphi^2(x)$$

to complete the squares!

$$\begin{aligned} \dot{V} &= -cx^2 - x^2 s(x) + x\varphi(x)\Delta \\ &= -cx^2 - \kappa\left[x\varphi - \frac{\Delta}{2\kappa}\right]^2 + \kappa \cdot \frac{\Delta^2}{4\kappa^2} \quad \le -cx^2 + \frac{\Delta^2}{4\kappa} \end{aligned}$$

Note! $\dot{V}$ is negative whenever

$$|x(t)| \ge \frac{\Delta}{2\sqrt{\kappa c}}$$

Can show that $x(t)$ converges to the set

$$R = \left\{ x : |x(t)| \le \frac{\Delta}{2\sqrt{\kappa c}} \right\}$$

i.e. x(t) stays bounded for all bounded disturbances Δ

Remark: The nonlinear damping $-\kappa x \varphi^2(x)$ renders the system Input-To-State Stable (ISS) with respect to the disturbance.

Overview

- To give a glimpse of modern nonlinear control. The theory in this lecture will not be part of the exam*.

Material

- Lecture slides
- Matrial covered in
  - Slotine and Li, pp. 191-262
  - Glad & Ljung Ch 17.1
  - Khalil Ch 13.1

Controllability

Linear case

$$\dot{x} = Ax + Bu$$

All controllability definitions coincide

$$\begin{aligned} &0 \to x(T), \\ &x(0) \to 0, \\ &x(0) \to x(T) \end{aligned}$$

T either fixed or free

Rank condition System is controllable iff

$$W_n = \begin{pmatrix} B & AB & \dots & A^{n-1}B \end{pmatrix} \text{ full rank}$$

Is there a corresponding result for nonlinear systems?

Lie Brackets

Lie bracket between $f(x)$ and $g(x)$ is defined by

$$[f, g] = \frac{\partial g}{\partial x} f - \frac{\partial f}{\partial x} g$$

Example:

$$f = \begin{pmatrix} \cos x_2 \\ x_1 \end{pmatrix}, \qquad g = \begin{pmatrix} x_1 \\ 1 \end{pmatrix},$$

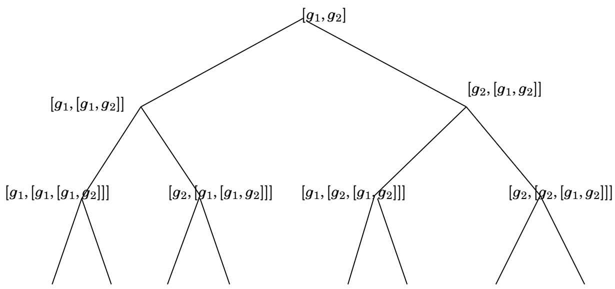
$$\begin{aligned} [f, g] &= \frac{\partial g}{\partial x} f - \frac{\partial f}{\partial x} g \\ &= \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \cos x_2 \\ x_1 \end{pmatrix} - \begin{pmatrix} 0 & -\sin x_2 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ 1 \end{pmatrix} \\ &= \begin{pmatrix} \cos x_2 + \sin x_2 \\ -x_1 \end{pmatrix} \end{aligned}$$

Why interesting?

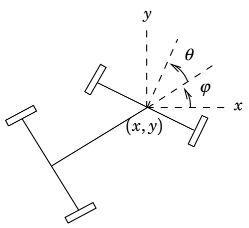
$$\dot{x} = g_1(x)u_1 + g_2(x)u_2$$

- The motion $(u_1, u_2) = \begin{cases} (1,0), & t \in [0, \epsilon] \\ (0,1), & t \in [\epsilon, 2\epsilon] \\ (-1,0), & t \in [2\epsilon, 3\epsilon] \\ (0,-1), & t \in [3\epsilon, 4\epsilon] \end{cases}$
  gives motion $x(4\epsilon) = x(0) + \epsilon^2 [g_1, g_2] + O(\epsilon^3)$
- $\Phi^t_{[g_1,g_2]} = \lim_{n\to\infty} (\Phi^{\sqrt{\frac{t}{n}}}_{-g_2} \Phi^{\sqrt{\frac{t}{n}}}_{-g_1} \Phi^{\sqrt{\frac{t}{n}}}_{g_2} \Phi^{\sqrt{\frac{t}{n}}}_{g_1})^n$
- The system is controllable if the **Lie bracket tree** has full rank (controllable=the states you can reach from $x = 0$ at fixed time T contains a ball around $x = 0$)

The Lie Bracket Tree



Parking Your Car Using Lie-Brackets



$$\frac{d}{dt} \begin{pmatrix} x \\ y \\ \varphi \\ \theta \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} u_1 + \begin{pmatrix} \cos(\varphi + \theta) \\ \sin(\varphi + \theta) \\ \sin(\theta) \\ 0 \end{pmatrix} u_2$$

Parking the Car

Can the car be moved sideways?

Sideways: in the $(-\sin(\varphi), \cos(\varphi), 0, 0)^T$-direction?

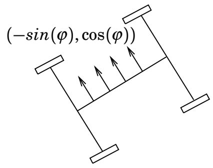
$$\begin{aligned} [g_1, g_2] &= \frac{\partial g_2}{\partial x} g_1 - \frac{\partial g_1}{\partial x} g_2 \\ &= \begin{pmatrix} 0 & 0 & -\sin(\varphi+\theta) & -\sin(\varphi+\theta) \\ 0 & 0 & \cos(\varphi+\theta) & \cos(\varphi+\theta) \\ 0 & 0 & 0 & \cos(\theta) \\ 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} - 0 \\ &= \begin{pmatrix} -\sin(\varphi+\theta) \\ \cos(\varphi+\theta) \\ \cos(\theta) \\ 0 \end{pmatrix} =: g_3 = \text{“wriggle”} \end{aligned}$$

Once More

$$[g_3, g_2] = \frac{\partial g_2}{\partial x} g_3 - \frac{\partial g_3}{\partial x} g_2 = \dots$$

$$= \begin{pmatrix} -\sin(\varphi) \\ \cos(\varphi) \\ 0 \\ 0 \end{pmatrix} = \text{“sideways”}$$

The motion $[g_3, g_2]$ takes the car sideways.

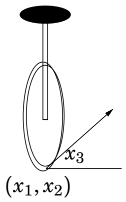


The Parking Theorem

You can get out of any parking lot that is bigger than your car. Use the following control sequence:

Wriggle, Drive, –Wriggle(this requires a cool head), –Drive (repeat).

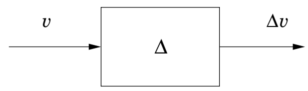
Another example — The unicycle



$$g_1 = \begin{pmatrix} \cos(x_3) \\ \sin(x_3) \\ 0 \end{pmatrix}, \quad g_2 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \quad [g_1, g_2] = \begin{pmatrix} \sin(x_3) \\ -\cos(x_3) \\ 0 \end{pmatrix}$$

Full rank, controllable.

Integral Quadratic Constraint

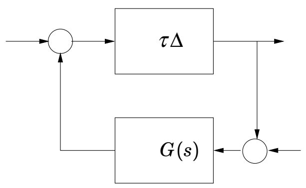


The (possibly nonlinear) operator Δ on $\mathbf{L}_2^m[0, \infty)$ is said to *satisfy the IQC defined by* Π if

$$\int_{-\infty}^{\infty} \begin{bmatrix} \widehat{v}(i\omega) \\ \widehat{(\Delta v)}(i\omega) \end{bmatrix}^* \Pi(i\omega) \begin{bmatrix} \widehat{v}(i\omega) \\ \widehat{(\Delta v)}(i\omega) \end{bmatrix} d\omega \geq 0$$

for all $v \in \mathbf{L}_2[0, \infty)$.

IQC Stability Theorem



Let $G(s)$ be stable and proper and let Δ be causal.

For all $\tau \in [0, 1]$, suppose the loop is well posed and $\tau\Delta$ satisfies the IQC defined by $\Pi(i\omega)$. If

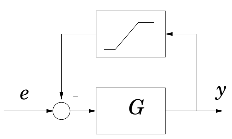
$$\begin{bmatrix} G(i\omega) \\ I \end{bmatrix}^* \Pi(i\omega) \begin{bmatrix} G(i\omega) \\ I \end{bmatrix} < 0 \quad \text{for } \omega \in [0, \infty]$$

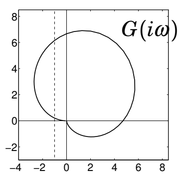
then the feedback system is input/output stable.

Δ structure	$\Pi(i\omega)$	Condition
Δ passive	$\begin{bmatrix} 0 & I \\ I & 0 \end{bmatrix}$	
$\lVert\Delta(i\omega)\rVert \leq 1$	$\begin{bmatrix} x(i\omega)I & 0 \\ 0 & -x(i\omega)I \end{bmatrix}$	$x(i\omega) \geq 0$
$\delta \in [-1, 1]$	$\begin{bmatrix} X(i\omega) & Y(i\omega) \\ Y(i\omega)^* & -X(i\omega) \end{bmatrix}$	$X = X^* \geq 0$, $Y = -Y^*$
$\delta(t) \in [-1, 1]$	$\begin{bmatrix} X & Y \\ Y^T & -X \end{bmatrix}$	
$\Delta(s) = e^{-\theta s} - 1$	$\begin{bmatrix} x(i\omega)\rho(\omega)^2 & 0 \\ 0 & -x(i\omega) \end{bmatrix}$	$\rho(\omega) = 2\max_{\lvert\theta\rvert \leq \theta_0} \sin(\theta\omega/2)$

A Matlab toolbox for system analysis

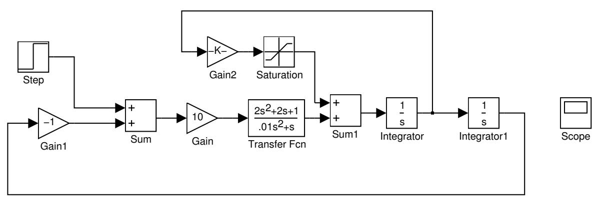
`http://www.ee.mu.oz.au/staff/cykao/`



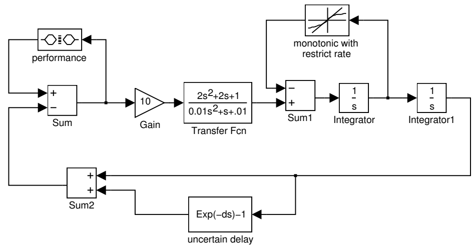



```
>> abst_init_iqc;
>> G = tf([10 0 0],[1 2 2 1]);
>> e = signal
>> w = signal
>> y = -G*(e+w)
>> w==iqc_monotonic(y)
>> iqc_gain_tbx(e,y)
```

A servo with friction

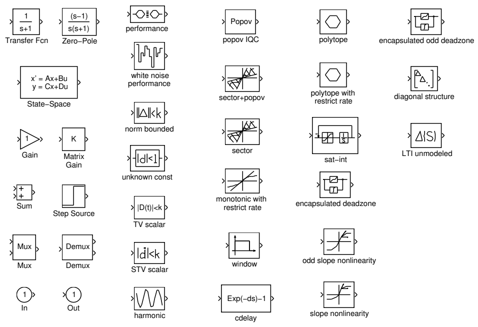


An analysis model defined graphically



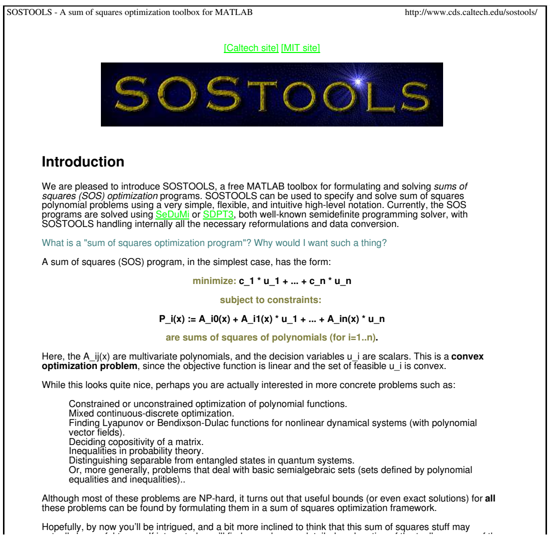

```
ż iqc_gui('fricSYSTEM')

extracting information from fricSYSTEM ...

    scalar inputs:  5
    states:         10
    simple q-forms: 7

    LMI #1    size = 1    states: 0
    LMI #2    size = 1    states: 0
    LMI #3    size = 1    states: 0
    LMI #4    size = 1    states: 0
    LMI #5    size = 1    states: 0

  Solving with 62 decision variables ...

  ans =    4.7139
```

A library of analysis objects



The friction example in text format

```
d=signal;                                      % disturbance signal
e=signal;                                        % error signal
w1=signal;                                       % friction force
w2=signal;                                     % delay perturbation
u=signal;                                        % control force
v=tf(1,[1 0])*(u-w1)                             % velocity
x=tf(1,[1 0])*v;                                 % position
e==d-x-w2;
u==10*tf([2 2 1],[0.01 1 0.01])*e;
w1==iqc_monotonic(v,0,[1 5],10)
w2==iqc_cdelay(x,.01)
iqc_gain_tbx(d,e)
```


SOSTOOLS - A sum of squares optimization toolbox for MATLAB http://www.cds.caltech.edu/sostools/

[Caltech site] [MIT site]

SOSTOOLS

Introduction

We are pleased to introduce SOSTOOLS, a free MATLAB toolbox for formulating and solving *sums of squares (SOS) optimization* programs. SOSTOOLS can be used to specify and solve sum of squares polynomial problems using a very simple, flexible, and intuitive high-level notation. Currently, the SOS programs are solved using SeDuMi or SDPT3, both well-known semidefinite programming solver, with SOSTOOLS handling internally all the necessary reformulations and data conversion.

What is a "sum of squares optimization program"? Why would I want such a thing?

A sum of squares (SOS) program, in the simplest case, has the form:

minimize: c_1 * u_1 + ... + c_n * u_n

subject to constraints:

P_i(x) := A_i0(x) + A_i1(x) * u_1 + ... + A_in(x) * u_n

are sums of squares of polynomials (for i=1..n).

Here, the A_ij(x) are multivariate polynomials, and the decision variables u_i are scalars. This is a **convex optimization problem**, since the objective function is linear and the set of feasible u_i is convex.

While this looks quite nice, perhaps you are actually interested in more concrete problems such as:

Constrained or unconstrained optimization of polynomial functions.
Mixed continuous-discrete optimization.
Finding Lyapunov or Bendixson-Dulac functions for nonlinear dynamical systems (with polynomial vector fields).
Deciding copositivity of a matrix.
Inequalities in probability theory.
Distinguishing separable from entangled states in quantum systems.
Or, more generally, problems that deal with basic semialgebraic sets (sets defined by polynomial equalities and inequalities)..

Although most of these problems are NP-hard, it turns out that useful bounds (or even exact solutions) for **all** these problems can be found by formulating them in a sum of squares optimization framework.

Hopefully, by now you'll be intrigued, and a bit more inclined to think that this sum of squares stuff may

Hybrid Control

Control problems where there is a mixture between continuous states and discrete state variables.

Continuous states: position, velocity, temperature, pressure

Discrete states: on/off variables, controller modes, loss of actuators, loss of sensors, relays, etc

Discontinuous differential equations

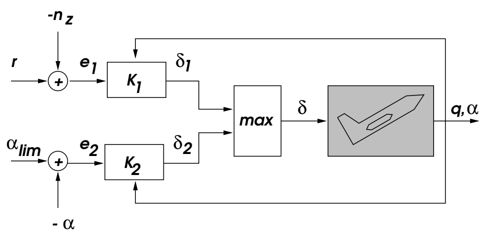
Much active field, much left to understand

Example of hybrid control

Control law that switches between different modes, e.g. between

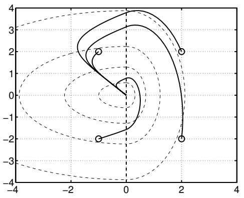
- Time optimal control – during large set point changes
- Linear control – close to set point

Aircraft Example



(Branicky, 1993)

Phase Plane



No common *quadratic* Lyapunov function exists.

$$A_1 = \begin{bmatrix} -5 & -4 \\ -1 & -2 \end{bmatrix} \qquad A_2 = \begin{bmatrix} -2 & -4 \\ 20 & -2 \end{bmatrix}$$

Piecewise quadratic Lyapunov functions

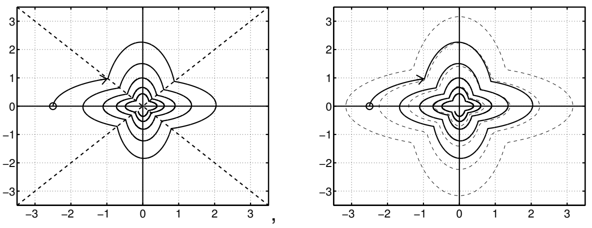
$$V(x) = \begin{cases} x^*Px & \text{if } x_1 < 0 \\ x^*Px + \eta x_1^2 & \text{if } x_1 \geq 0 \end{cases}$$

The matrix inequalities

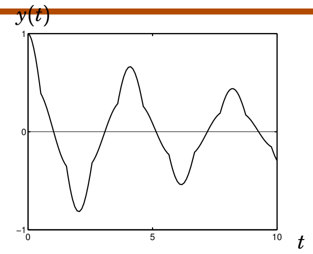
$$\begin{aligned} A_1^*P + PA_1 &< 0 \\ P &> 0 \\ A_2^*(P + \eta E^*E) + (P + \eta E^*E)A_2 &< 0 \\ P + \eta E^*E &> 0 \end{aligned}$$

with $E = [1\ 0]$, have the solution $P = \text{diag}\{1, 3\}$, $\eta = 7$.

Flower Example



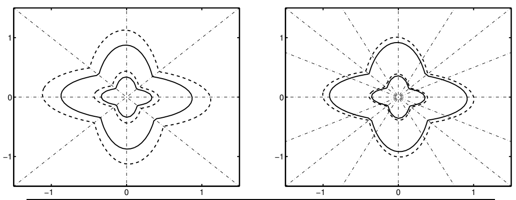
Observability



$$\begin{cases} \dot{x}(t) = A_{i(t)}x(t) \\ y(t) = C_{i(t)}x(t) \end{cases} \quad \text{for } x(t) \in X_{i(t)} \tag{1}$$

Estimate $\int_0^\infty |y|^2 dt$ given $x(0)$

Bounds from piecewise quadratic Lyapunov functions



Number of Cells	Lower bound	Upper bound
4	0.60	2.50
8	1.33	2.18
16	1.65	1.98
32	1.78	1.88

Adaptive Control/Predictive Control

(Ch. 8 in Slotine and Li / examples in Khalil)

Adaptive Control/Predictive Control HT-Lp 1 (5p)

Many techniques from the nonlinear course are useful

- Stability
- Lyapunov theory
- Passivity
- Simulation

Take advantage of your knowledge and read this course!

Department of
Automatic Control

Master's Thesis

The Department always has a number of suggestion for master thesis projects. We can also often help to establish contacts with companies which are interested in master thesis projects, both in Sweden and abroad. The topics were we have thesis projects are modeling and simulation, real-time systems, process control, automotive systems, robotics, and a large range of application areas for control, automation, and real-time systems.

If you are interested in a master thesis with us you should contact Karl-Erik Årzén < Karl-Erik.Arzen@control.lth.se> You can also discuss with the teachers in the different control courses.

The aim and the rules for a Master's Thesis are described in LTH's rules for Master Theses. A Swedish information sheet is available in pdf format.

- Ongoing projects in chronological order
- Recently finished projects

Some examples of Swedish companies which recently have had thesis projects with are Volvo, ABB, Ericsson, Tetra Pak, Alfa Laval, Astra Zeneca, AssiDomän Frövi, Dynasim, Haldex Traction, and TAC.

The Department has good international contacts and can help to arrange thesis projects in other countries. Some universities where master thesis have been made recently are:

- University of Newcastle, Australia
- University of Melbourne, Australia
- Laboratoire d'automatique de Grenoble, France
- University of Coimbra, Potugal
- Swiss Federal Institute of Technology (EPF), Lausanne, Switzerland
- Imperial College, London, UK
- University of California, Berkeley, USA
- California Institute of Technology, USA (as a part of the SURF-program)

Some thesis projects have also been done at companies abroad. Recent examples are:

- ABB, Switzerland and Germany
- Daimler-Chrysler, Berlin
- ZF Lenksysteme, Germany
- General Electric Global Research, Munich

More Information

- Nijmeijer, van der Schaft, Nonlinear Dynamical Control Systems, Springer Verlag. More theory about Lie-bracket theory
- Vidyasagar, Nonlinear systems analysis, Prentice Hall
- Isidori, Nonlinear Control Systems, Springer Verlag

Last Lecture

- A quick scan through the material again. Questions.

Take the rest of the lecture to write down questions for me to the last lecture.