Aims of the Course

Where do we find such systems?
Your mobile phone, your car, your washer, your home
Your energy supplier, your public transportation, your cells

What are the consequences?
The infrastructure of our society relies on their dependability
However, modeling, analysis and control is very challenging

What are you going to learn?
Mathematical principles underlying such systems
How to model, analyse and control hybrid systems
Course Organization

182.732 VU Hybrid Systems (3 ECTS):
   Dedicated to teaching the fundamentals of CPS
   No homeworks, but with a final exam. Midterm wanted?

182.733 LU Hybrid Systems (3 ECTS, Optional):
   Dedicated to applying the knowledge acquired in the VU
   A group project. You may also propose your own project.
Computer network with engine and wings

Computer with eyes, ears and voice

Computer network with engine and wheels
Cyber-Physical Systems (CPS): Orchestrating networked computational resources with physical systems

- Power generation and distribution (Courtesy of General Electric)
- Military systems: E-Corner, Siemens
- Transportation (Air traffic control at SFO)
- Avionics
- Telecommunications
- Factory automation
- Instrumentation (Soleil Synchrotron)
- Automotive (Daimler-Chrysler)
- Building Systems
- Aeronautics
- Avionics
- Wireless Comm
- Factory Automation
- Avionics Aeronautics
- Auto-motive
- Comm Backbone
- Power supply
- Wireless Comm
Prerequisites

Computer Science:

Finite automata theory, logics and boolean algebra
Abstraction, temporal logics, formal verification

Control Theory:

Differential and difference equations, linear algebra
Approximation, observability, controllability, stability
Literature: Books

– Lygeros, Tomlin, Sastry. Hybrid Systems: Modeling analysis and control
– Tabuada. Verification and control of hybrid systems: A symbolic approach
– Lee and Varaiya. Structure and interpretation of signals and systems

– Alur. Principles of Embedded Computation
– Lee and Seshia. Introduction to Embedded Systems: A CPS Approach
– Clarke, Grumberg and Peled. Model checking
Literature: Articles


Literature: Articles


Literature: Articles


Literature: Articles


Verification Tools for Hybrid Systems

HyTech: LHA
http://embedded.eecs.berkeley.edu/research/hytech/

PHAVer: LHA + affine dynamics
http://www-verimag.imag.fr/~frehse/

d/dt: affine dynamics + controller synthesis
http://www-verimag.imag.fr/~tdang/Tool-ddt/ddt.html

Matisse Toolbox: zonotopes
http://www.seas.upenn.edu/~agirard/Software/MATISSE/

HSOLVER: nonlinear systems
http://hsolver.sourceforge.net/

SpaceEx: LHA + affine dynamics
http://spaceex.imag.fr/
Cyber-Physical Systems (CPS): Orchestrating networked computational resources with physical systems

- Power generation and distribution
- Courtesy of General Electric
- Military systems: E-Corner, Siemens
- Transportation (Air traffic control at SFO)
- Avionics
- Telecommunications
- Factory automation
- Instrumentation (Soleil Synchrotron)
- Automotive
- Building Systems
- Aeronautics
- Auto-motive
- Wireless Comm
- Comm Backbone
- Power supply
- Hybrid Systems
Cyber-Physical Systems
Cyber-Physical Models

Cyber Model

Physical Model
Analysis and Synthesis

Cyber Model

Temporal Prop

Physical Model
Physical Model: Signals

Continuous Signal: Function $f : \mathbb{R} \rightarrow \mathbb{R}^n$

Time \quad Value domain

Physical Model

Input Signal \quad Output Signal
Physical Model: Signals

Continuous Signal (SignalCT): Function $f : \mathbb{R} \rightarrow \mathbb{R}^n$

Audio Signals: **Sound**: Time $\rightarrow$ Pressure
Physical Model: Signals

Discrete-time Signal (SignalDT): Function $f : \mathbb{N} \rightarrow \mathbb{R}^n$

Discrete-time audio: Sound : DiscreteTime $\rightarrow$ Pressure

![Graph showing input and output signals with frequency and pressure values.]
Physical Model: Signals

Discrete-space Signal (SignalDS): Function $f : \mathbb{N}^n \rightarrow \mathbb{R}$

Images: $Image : VSpace \times HSpace \rightarrow Intensity$

Physical Model

Input Signal  Output Signal
Physical Model: Signals

Video Signals (SignalVS): Function $f : \mathbb{N} \rightarrow \text{SignalDS}$

Position, Velocity, Acceleration: $f : \mathbb{R} \rightarrow \mathbb{R}^3$

Temperature: $f : \mathbb{R} \rightarrow (\mathbb{R}^3 \rightarrow \mathbb{R})$

Boolean Sequences: $f : \mathbb{N} \rightarrow \mathbb{B}$

Event Stream: $f : \mathbb{N} \rightarrow \text{EventSet}$

Input Signal  →  Physical Model  ← Output Signal
Physical Model: Signals

**Sampling:** Depends on the nature of the function
Physical Model: Systems

**System:** Function $f : \text{Signal} \rightarrow \text{Signal}$

- **Input Signal**
- **Physical Model**
- **Output Signal**

**Input**

**Output**
Physical Model: Systems

System: Function $f : \text{Signal} \rightarrow \text{Signal}$

Transmission: Encoding and Decoding

Security: Encryption and decryption

Storage: Compression and decompression

Quality: Denoising, equalizing, filtering

Control: Transform output to control input
Physical Model: Systems

**System:** Function $f: \text{Signal} \rightarrow \text{Signal}$
Physical Model: Description

Differential Equations: \( \dot{x} = f(x, u, t), \ y = g(x, u, t), \ x(0) = x_0 \)

- Next state equation
- Current output equation
- Initial state

Input Signal  →  Physical Model  →  Output Signal
Physical Model: Description

Differential Equations: \( \dot{x} = f(x,u,t), \ y = g(x,u,t), \ x(0) = x_0 \)

- **State vector:** \( x \in \mathbb{R}^n \), **input vector:** \( u \in \mathbb{R}^k \), **output vector:** \( y \in \mathbb{R}^m \)

- **Next (infinitesimal) state function:** \( f : \mathbb{R}^n \times \mathbb{R}^k \times \mathbb{R} \to \mathbb{R}^n \)
  - Time invariant: \( \dot{x} = f(x,u), \ y = g(x,u), \) no explicit dependence on \( t \)
  - Linear: \( f(a_1 x_1 + a_2 x_2, u, t) = a_1 f(x_1, u, t) + a_2 f(x_2, u, t), \) similar for \( u \)

- **Output (observation) function:** \( g : \mathbb{R}^n \times \mathbb{R}^k \times \mathbb{R} \to \mathbb{R}^m \)
  - Moore: if \( g \) depends only on \( x \)