Goals for today

• Introduce concepts to be covered in the course (w/ context)
• Course structure & administration
Key elements
- Process: input/output system w/ dynamics
- Actuation: mechanism for manipulating process
- Sensing: mechanism for detecting process state
- Compute: compare actual / desired; determine action
- Environment: description of the uncertainty present in the system (bounded set of inputs/behaviors)

Advantages of feedback
- Design of dynamics
- Robustness to uncertainty
- Modularity and interoperability

Disadvantages of feedback
- Increased complexity
- Potential for instability
- Amplification of noise
Important Trends in Control in the Last 15* Years

(Online) Optimization-based control
- Increased use of online optimization (MPC/RHC)
- Use knowledge of (current) constraints & environment to allow performance and adaptability

Layering, architectures, networked control systems
- Command & control at multiple levels of abstraction
- Modularity in product families via layers

Formal methods for analysis, design and synthesis
- Build on work in hybrid and discrete event systems
- Formal methods from computer science, adapted for “cyberphysical” (computing + control) systems

Components → Systems → Enterprise
- Increased scale: supply chains, smart grid, IoT
- Use of modeling, analysis and synthesis techniques at all levels. Integration of “software” with “controls”
Design of Modern (Networked) Control Systems

Examples
- Aerospace systems
- Self-driving cars
- Factory automation/process control
- Smart buildings, grid, transportation

Challenges
- How do we define the layers/interfaces (vertical contracts)
- How do we scale to many devices (horizontal contracts)
- Stability, robustness, security, privacy

Control = dynamics, uncertainty, feedforward, feedback
Example: Autonomous Vehicles (Alice)

Vehicle
• 2005 Ford E-350 Van
• Drive-by-wire steering, brakes, accel

Sensing
• 5 cameras: 2 stereo pairs, roadfinding
• 5 LIDARs: long, med*2, short, bumper
• 2 GPS units + 1 IMU (LN 200)

Computing (2005)
• 6 Dell PowerEdge Servers (P4, 3GHz)
• 1 IBM Quad Core AMD64 (fast!)
• 1 Gb/s switched ethernet

Software
• 15 programs with ~100 exec threads
• 100,000+ lines of executable code
Part 1: Trajectory Generation and Tracking

Goal: find a feasible trajectory that satisfies dynamics/constraints

\[ \min J = \int_{t_0}^{T} q(x, u) \, dt + V(x(T), u(T)) \]
\[ \dot{x} = f(x, u) \quad lb \leq g(x, u) \leq ub \]

Solve as a constrained optimization problem

- Various tricks to get very fast calculations
- Need to update solutions at the rate at which the reference (task description) is modified

Use feedback ("inner loop") to track trajectory

- Trajectory generation provides feasible trajectory plus nominal input
- Feedback used to correct for disturbances and model uncertainties
- Example of “two degree of freedom” design

Nonlinear design
- global nonlinearities
- input saturation
- state space constraints

Simons Institute, 24 Jan 2018
Richard M. Murray, Caltech CDS
Part 2: Sensing and Estimation

Estimate current state + environment future
- Use real-time sensor inputs (GPS, LIDAR, etc) + model of the system/environment
- Techniques can be model-based, data-based (ML), or a combination
- Must take into account uncertainty: noise + unknown actions within the environment

Key tool: stochastic systems
- Model the world as a stochastic dynamical system (eg, GP)
- Model-driven: use model of system dynamics to improve estimate (Kalman filter)
- Data-driven: use sensor measurements over past to estimate future (MHE, ML, etc)
Course Administration

Prerequisites
- Intro controls (CDS 110/Ae 103a)
  - Mainly state space after HW #1
- Ability to use Python (laptop, Colab)

Course syllabus
- Instructors (lecturers, TAs)
- Lectures (MW) + Python (WF)
- Office hours, Q&A forum (Piazza)
- Course outline
- Homework (W → W + grace period)
- Grading scheme, collaboration policy
- Course text and references
- Course load: keep track of hours
- Course ombuds: send e-mail to Richard by Thu evening to volunteer

https://www.cds.caltech.edu/cds112