

# 15-884 – Control 1: Introduction

J. Zico Kolter

October 29, 2013

# Control Problems

- Thus far in the course, we have focused on allocation problems (e.g., optimal power flow) that are static in time
- Many tasks in sustainability have large *temporal* components
- We'll use the term “control tasks” to talk broadly about settings where we take actions over time in a dynamical system to minimize some total cost

- Control tasks are ubiquitous in energy/sustainability problems
  - Heating/cooling of buildings in response to occupancy
  - Storing electrical power in response to renewable resources and electricity pricing
  - Maximum power point tracking for renewable energy systems
  - Scheduling generation over time in the power grid

## Some definitions

- **State:**  $x \in \mathbb{R}^n$ 
  - Captures those elements of system that are relevant to its evolution but not controlled by agent
- **Control input:**  $u \in \mathbb{R}^m$ 
  - Captures elements that are directly specified by the agent
- **Policy/controller:**  $\pi : \mathbb{R}^n \rightarrow \mathbb{R}^m$ 
  - $u = \pi(x)$  prescribes input  $u$  as a function of state  $x$

# Dynamical Systems

- A dynamical system describes how state evolves over time, in response to control inputs
- Typical setting: *first-order* dynamics (next state or time derivative depends only on current state and control)
  - Discrete-time:  $x_{t+1} = f(x_t, u_t)$
  - Continuous-time:  $\frac{dx}{dt} \equiv \dot{x} = f(x, u)$
  - Can (approximately) transform between these two via Euler integration

$$x_{t+1} = x_t + (\Delta t)f(x_t, u_t)$$

- Lots of extensions and special cases of dynamical systems (we will cover these in more detail in the next lecture)
  - *Partially observable systems*: can't observe state  $x_t$  directly, just some observation  $y_t \in \mathbb{R}^p$

$$y_t = h(x_t)$$

- *Stochastic systems*: some uncertainty or noise in the dynamics

$$x_{t+1} = f(x_t, u_t) + \epsilon_t$$

- *Markov Decision Processes*: a setting with general dynamics and stochasticity, typically applied to discrete state / discrete action settings

## Control as Optimization

- Basis of *optimal control* is the notion of a cost function

$$C : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}_+$$

- $C(x, u)$  captures “badness” of being in state  $x$  and executing action  $u$ .

- Goal is to pick actions that minimize the sum of costs over some horizon  $T$  (also called cost-to-go, value function)

$$J(u_{1:T}) = \sum_{t=1}^T C(x_t, u_t) \quad \text{subject to} \quad x_{t+1} = f(x_t, u_t)$$

$$J(u_{1:T}) = \int_0^T C(x, u) dt \quad \text{subject to} \quad \dot{x} = f(x, t)$$

- Optimization formulation of (deterministic) optimal control

$$\underset{x_{1:T}, u_{1:T}}{\text{minimize}} \quad \sum_{t=1}^T C(x_t, u_t)$$

$$\text{subject to } x_{t+1} = f(x_t, u_t), \quad t = 1, \dots, T - 1$$

$$g(x_{1:T}, u_{1:T}) \leq 0, \quad (\text{arbitrary inequality constraints})$$

$$h(x_{1:T}, u_{1:T}) = 0, \quad (\text{arbitrary equality constraints})$$

- A very general way of formulating optimal control, but solving the optimization problem is still hard, except for special cases of systems



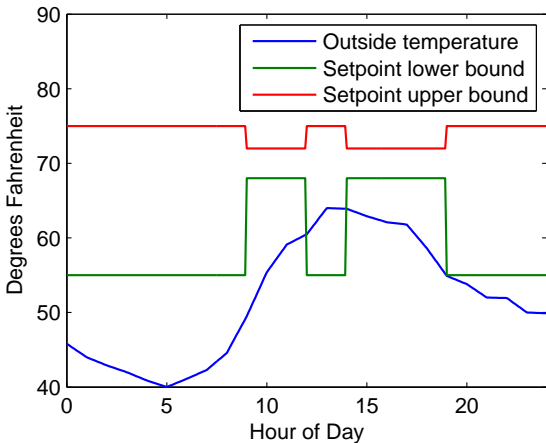
- Today, we'll focus on a specific case that can be solved efficiently: affine system + convex cost and constraints

$$\begin{aligned} & \underset{x_{1:T}, u_{1:T}}{\text{minimize}} && \sum_{t=1}^T C(x_t, u_t) \\ & \text{subject to} && x_{t+1} = Ax_t + Bu_t + a_t, \quad t = 1, \dots, T-1 \\ & && g(x_{1:T}, u_{1:T}) \leq 0 \\ & && h(x_{1:T}, u_{1:T}) = 0 \end{aligned}$$

- Can again be solved using our off-the-shelf solvers (e.g. YALMIP)

## Example 1: Building Heating

- Maintain building temperature within setpoints:



- Heat transfer dynamics (temperature  $T$ )

$$\rho c_p V \frac{dT}{dt} = -UA(T - T^{(\text{external})})$$

- Define state space and control

$x_t$  = internal temperature at time  $t$

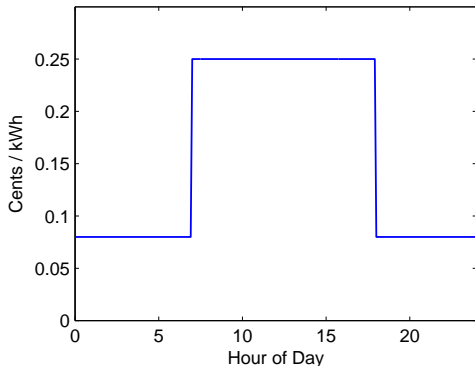
$u_t$  = additional injected energy (i.e., heating) at time  $t$

- Leads to discrete time approximation

$$x_{t+1} = x_t + k(T_t^{(\text{external})} - x_t) + bu_t$$

- A linear dynamical system
- Physical constants and time step all folded into constants  $k, b$

- Time-based pricing of electricity:



- Task: minimize cost of electricity while maintaining building within heating bounds

- Formulate this as the optimization problem (LP):

$$\underset{x,u}{\text{minimize}} \quad c^T u$$

$$\text{subject to} \quad x_{t+1} = x_t + k(T_t^{(\text{external})} - x_t) + bu_t$$

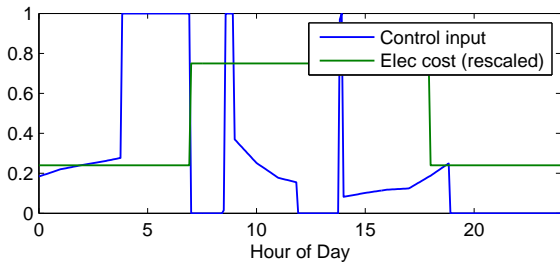
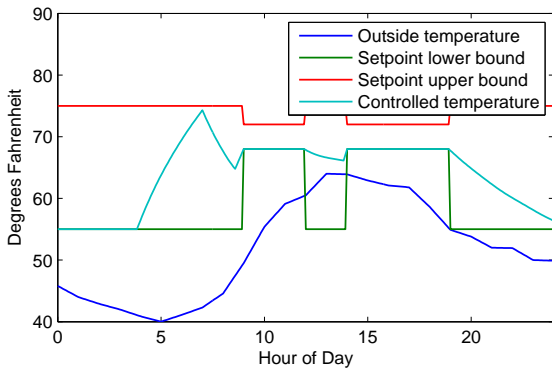
$$x_1 = T_1$$

$$T_l \leq x \leq T_u$$

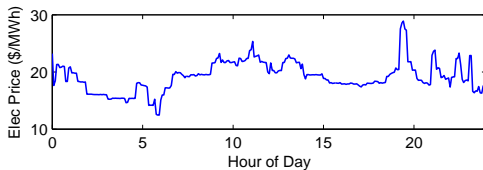
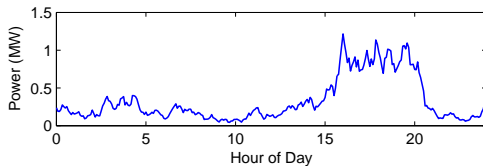
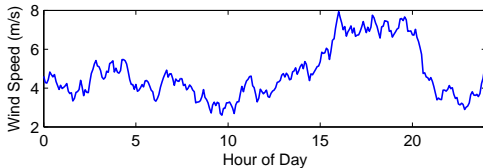
$$0 \leq u \leq 1$$

- MATLAB code:

```
x = sdpvar(T,1);
u = sdpvar(T,1);
C = [x(1) == 55];
for i=1:T-1,
    C = [C; x(i+1) == x(i) + k*(Te(i)-x(i)) + b*u(i)];
end
C = [C; x >= Tl; x <= Tu];
C = [C; u >= 0; u <= 1];
solvesdp(C, c'*u);
```



## Example 2: Energy Storage for Wind



- Lacking storage, we need to just sell energy generated at time  $t$  at the current market price
- If we have an energy storage system attached to the turbine, we can store energy if price is low, sell when high



- State space and control:

$x_t =$  total energy stored in battery at time  $t$

$u_t =$  energy put into or taken out of battery at time  $t$

- System dynamics

$$x_{t+1} = x_t + u_t \quad (0 \leq x_t \leq E_{\max})$$

- External variables:

$e_t =$  wind energy generated at time  $t$

$c_t =$  cost of electricity at time  $t$

- Optimization formulation

$$\underset{x,u}{\text{minimize}} \quad c^T(e - u)$$

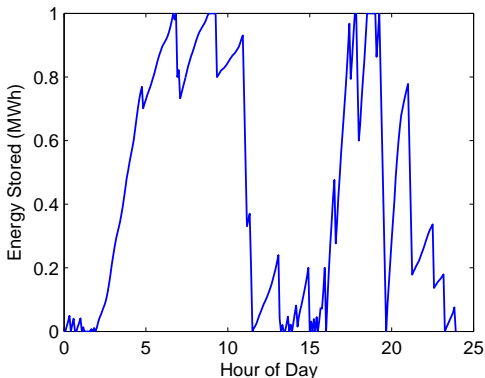
$$\text{subject to} \quad x_{t+1} = x_t + u_t, \quad t = 1, \dots, T - 1$$

$$0 \leq x \leq E_{\max}$$

$$-E_{\text{ramp}} \leq u \leq E_{\text{ramp}}$$

$$e - u \geq 0$$

- Solution (energy stored over the day):



In this case, make 17% more money from including storage (but real advantage is when wind penetration high, for avoiding spinning reserve)

## A preview of issues to come

- These examples were very simple dynamical systems; how do we handle more complex dynamics?
- In our optimization problems we assumed the dynamics were deterministic and the future was known; how do we deal with stochasticity and uncertain predictions?
- Exact optimal solutions will often be impossible to obtain, but approximate methods like *Model Predictive Control* often work very well in practice.