Robot Control: Motion Control with Velocities

CS 6301 Special Topics: Introduction to Robot Manipulation and Navigation

Professor Yu Xiang

The University of Texas at Dallas

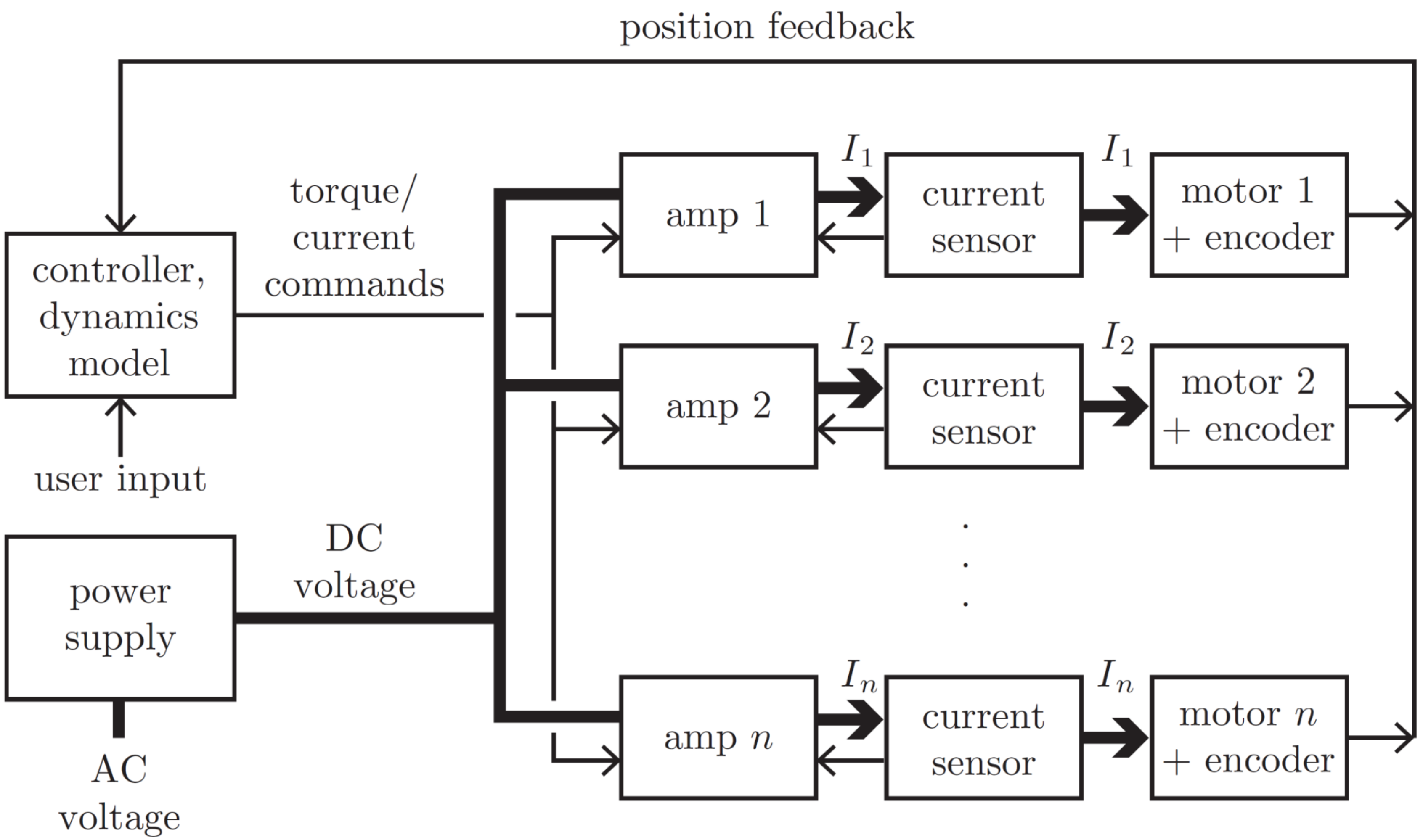
Robot Control

- Convert task specifications to forces and torques at the actuators

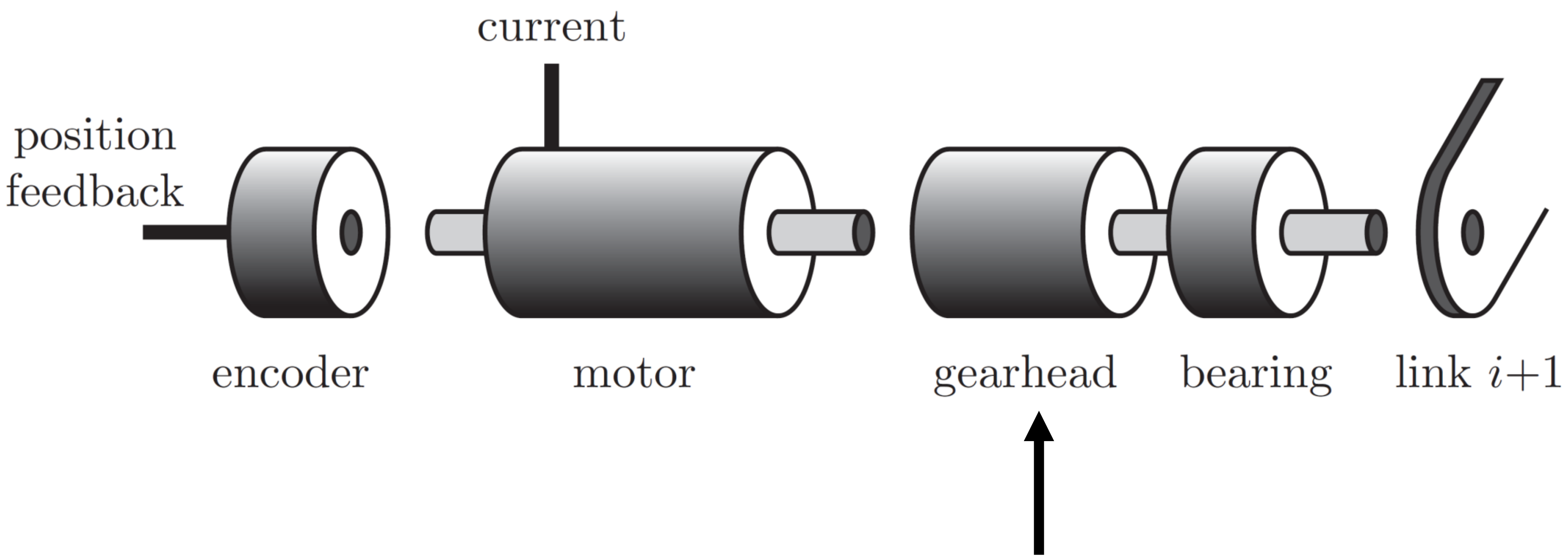
- Types
  - Motion control
  - Force control
  - Hybrid motion-force control
  - Impedance control

- Feedback control
  - Use sensors for position, velocity and force
  - Compare with the desired behavior to compute the control signals

Actuation with DC Electric Motors

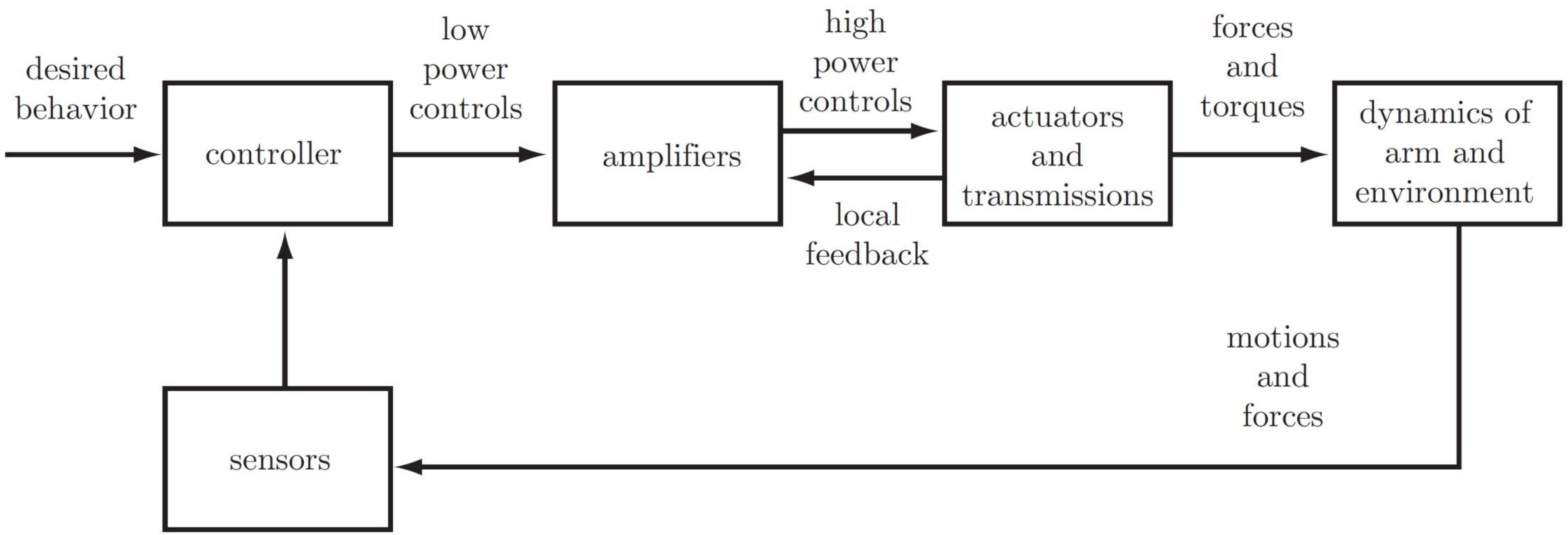


Actuation with DC Electric Motors



Increase torque
Reduce speed

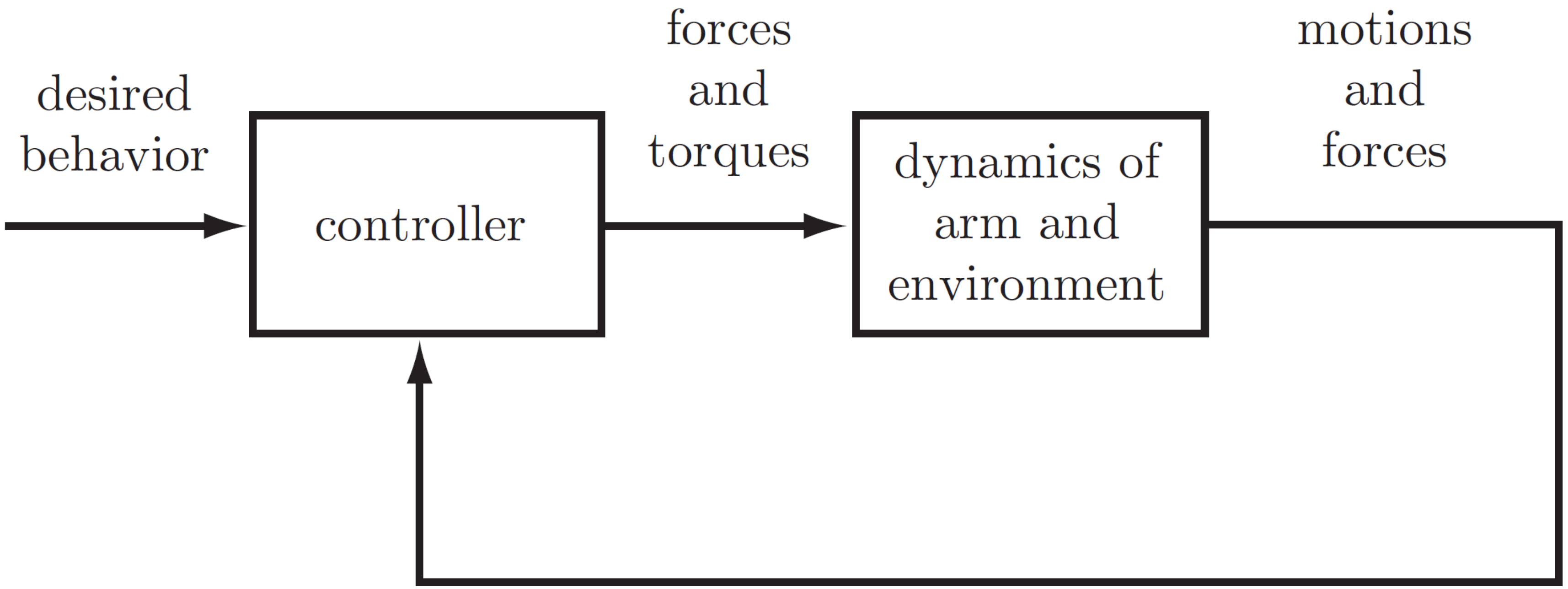
Control System Overview



- Potentiometers, encoders, or resolvers for joint position and angle sensing
- Tachometers for joint velocity sensing
- Joint force-torque sensors
- Multi-axis force-torque sensors at the “wrist" between the end of the arm and the end-effector

Control System Overview

- A simplified system



Controlled Dynamics of a Single Joint

- Desired joint position $\theta_d(t)$
- The current joint position $\theta(t)$
- Joint error $\theta_e(t) = \theta_d(t) - \theta(t)$

- Error dynamics: the differential equation governing the evolution of the joint error

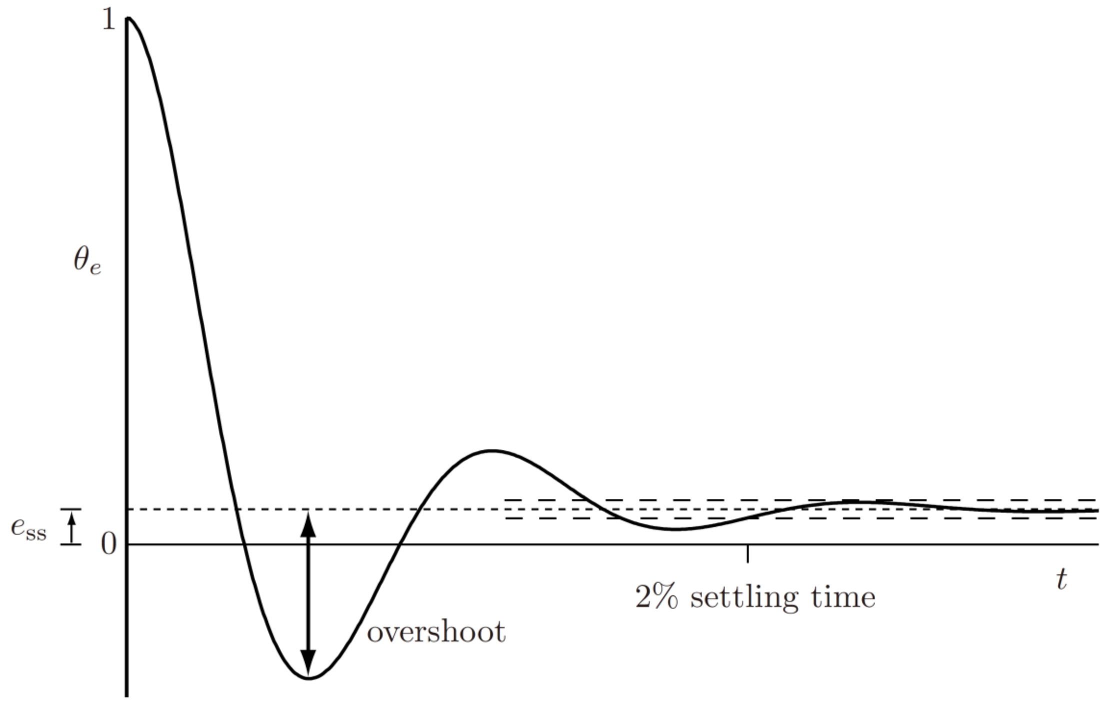
- Feedback controller: create an error dynamics to make $\theta_e(t)$ become zero or a small value when t increases

Error Response

- How well a controller works?
  - Specify a nonzero initial error $\theta_e(0)$ and see how the controller reduces the error

- Error response $\theta_e(t), t > 0$
  - Initial conditions $\theta_e(0) = 1$
    $\dot{\theta}_e(0) = \ddot{\theta}_e(0) = \cdots = 0$

  - Steady-state error $\theta_e(t)$ as $t \to \infty$

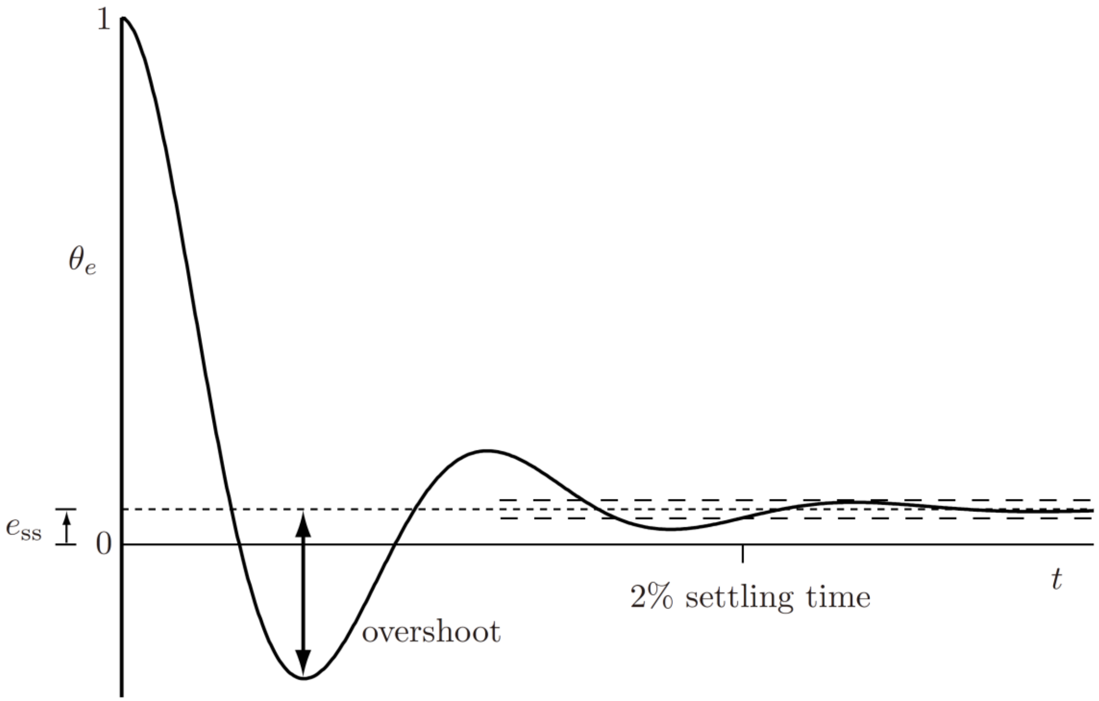


Error Response

- (2%) Settling time: first time T such that $|\theta_e(t) - e_{\text{ss}}| \leq 0.02(\theta_e(0) - e_{\text{ss}})$ for all $t \geq T$

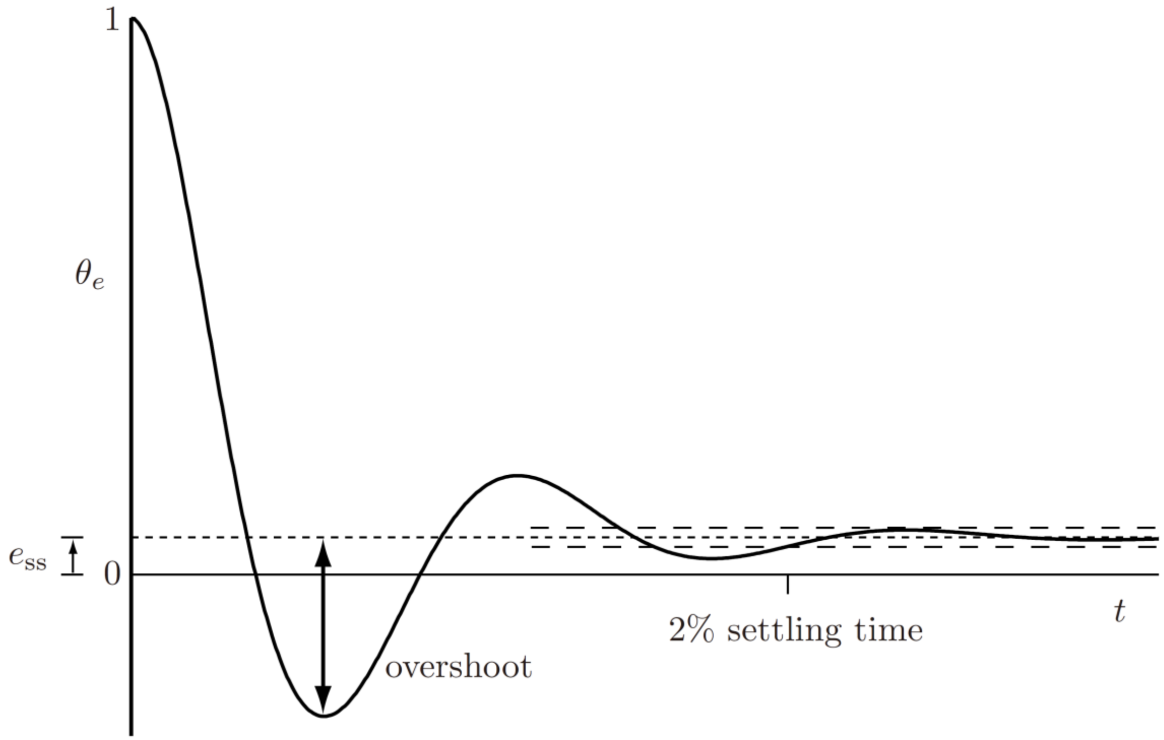
- Overshoot

$$\text{overshoot} = \left| \frac{\theta_{e,\min} - e_{\text{ss}}}{\theta_e(0) - e_{\text{ss}}} \right| \times 100\%$$



Error Response

- A good error response
  - Little or no steady-state error
  - Little or no overshoot
  - A short 2% settling time



Motion Control with Velocity Inputs

- Typically, we assume direct control of the forces or torques at robot joints

- In some cases, we can assume that there is direct control of the joint velocities
  - The velocity of a joint is determined directly by the frequency of the pulse train sent to the stepper motor https://en.wikipedia.org/wiki/Stepper_motor

- Motion control with velocity inputs
  - Given a desired trajectory of a robot in joint space or in task space

$\theta_d(t)$ $X_d(t)$

Motion Control of a Single Joint

- Feedforward control or open-loop control
  - Given a desired joint trajectory $\theta_d(t)$
  - Choose the velocity command $\dot{\theta}(t) = \dot{\theta}_d(t)$
  - Cons: accumulating position errors

- Feedback control
  - Measure the joint position continuously for feedback

Motion Control of a Single Joint

- Proportional controller or P controller

$$\dot{\theta}(t) = K_p(\theta_d(t) - \theta(t)) = K_p\theta_e(t)$$

Control gain

$K_p > 0$

- When $\theta_d(t)$ is a constant $\dot{\theta}_d(t) = 0$

Setpoint control

  - Error dynamics

$$\dot{\theta}_e(t) = \cancel{\dot{\theta}_d(t)}^{\,0} - \dot{\theta}(t)$$

$$\dot{\theta}_e(t) = -K_p\theta_e(t) \quad \rightarrow \quad \dot{\theta}_e(t) + K_p\theta_e(t) = 0$$

First-Order Error Dynamics

$$\dot{\theta}_e(t) + \frac{1}{\tau}\theta_e(t) = 0$$

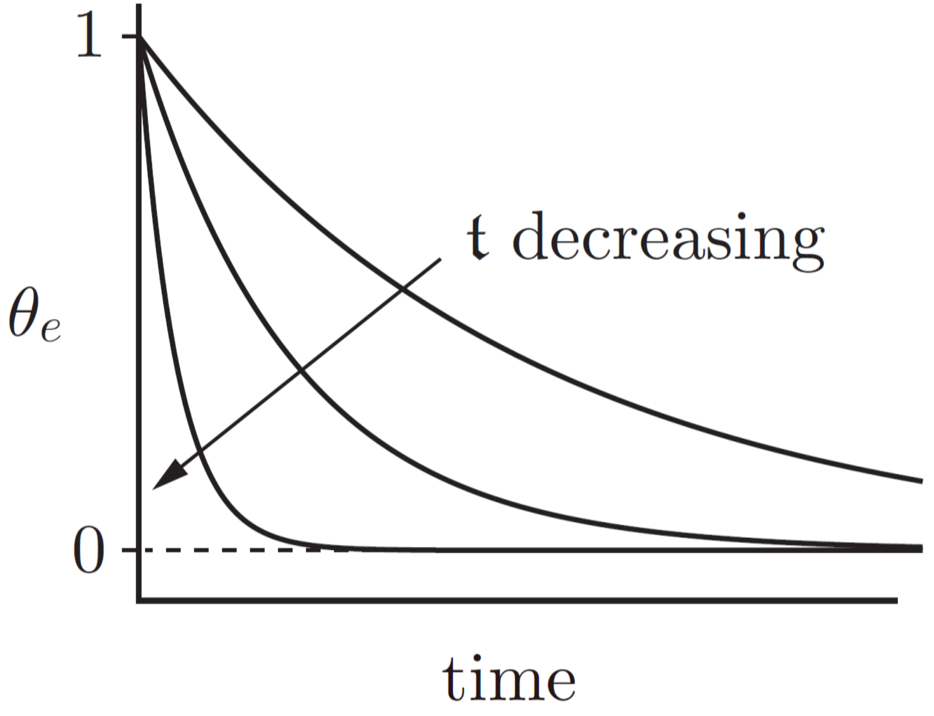
time constant τ

Solution $\theta_e(t) = e^{-t/\tau}\theta_e(0)$

Setpoint control

$$\dot{\theta}_e(t) + K_p\theta_e(t) = 0 \qquad \tau = 1/K_p$$

- 0 steady state error
- No overshoot
- 2% settling time $4/K_p$



P Controller

$$\dot{\theta}(t) = K_p(\theta_d(t) - \theta(t)) = K_p\theta_e(t)$$

- When $\theta_d(t)$ is not constant but $\dot{\theta}_d(t)$ is constant $\dot{\theta}_d(t) = c$
- Error dynamics

$$\dot{\theta}_e(t) = \dot{\theta}_d(t) - \dot{\theta}(t) = c - K_p\theta_e(t)$$

Solution

$$\theta_e(t) = \frac{c}{K_p} + \left(\theta_e(0) - \frac{c}{K_p}\right) e^{-K_p t} \longrightarrow c/K_p$$

steady-state error

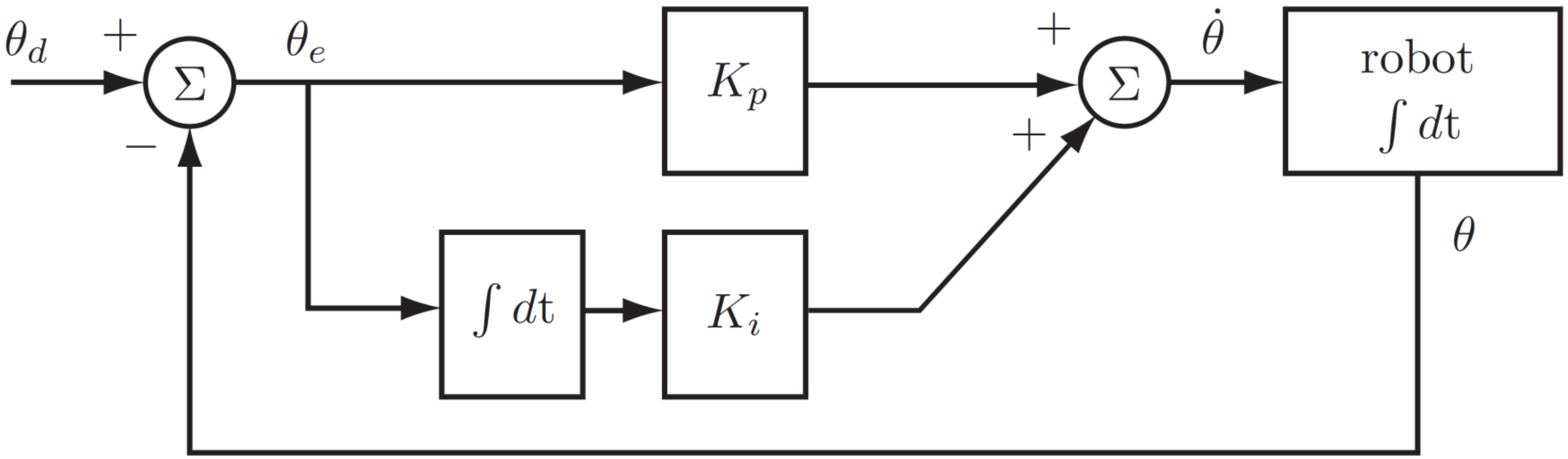
We cannot make K_p arbitrarily large (velocity limit, instability)

PI Controller

- A proportional-integral controller

$$\dot{\theta}(t) = K_p\theta_e(t) + K_i \int_0^t \theta_e(\mathrm{t})\ dt$$

Time-integral of the error



PI Controller

- Error dynamics for a constant $\dot{\theta}_d(t) = c$

$$\dot{\theta}(t) = K_p\theta_e(t) + K_i\int_0^t \theta_e(\mathrm{t})\, d\mathrm{t}$$

$$\dot{\theta}_e(t) = \dot{\theta}_d(t) - \dot{\theta}(t)$$

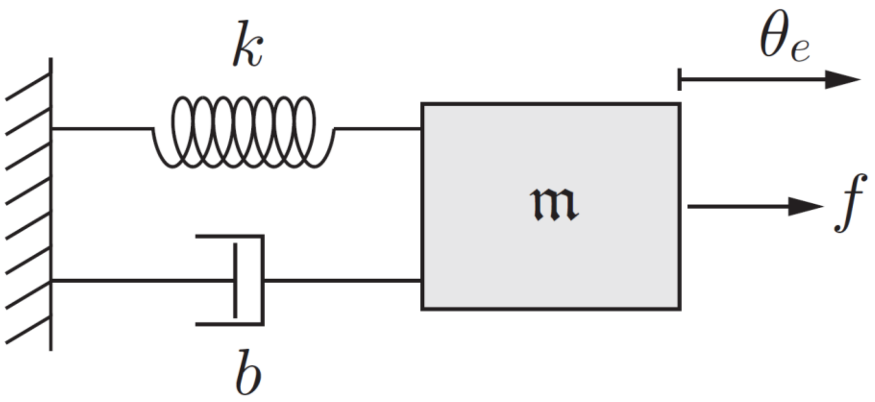
$$\dot{\theta}_e(t) + K_p\theta_e(t) + K_i\int_0^t \theta_e(\mathrm{t})\, d\mathrm{t} = c$$

$$\ddot{\theta}_e(t) + K_p\dot{\theta}_e(t) + K_i\theta_e(t) = 0$$

Second-Order Error Dynamics

PI Controller

- Mass-spring-damper



$$\mathfrak{m}\ddot{\theta}_e + b\dot{\theta}_e + k\theta_e = f$$

$$f = 0 \quad \ddot{\theta}_e(t) + \frac{b}{\mathfrak{m}}\dot{\theta}_e(t) + \frac{k}{\mathfrak{m}}\theta_e(t) = 0$$

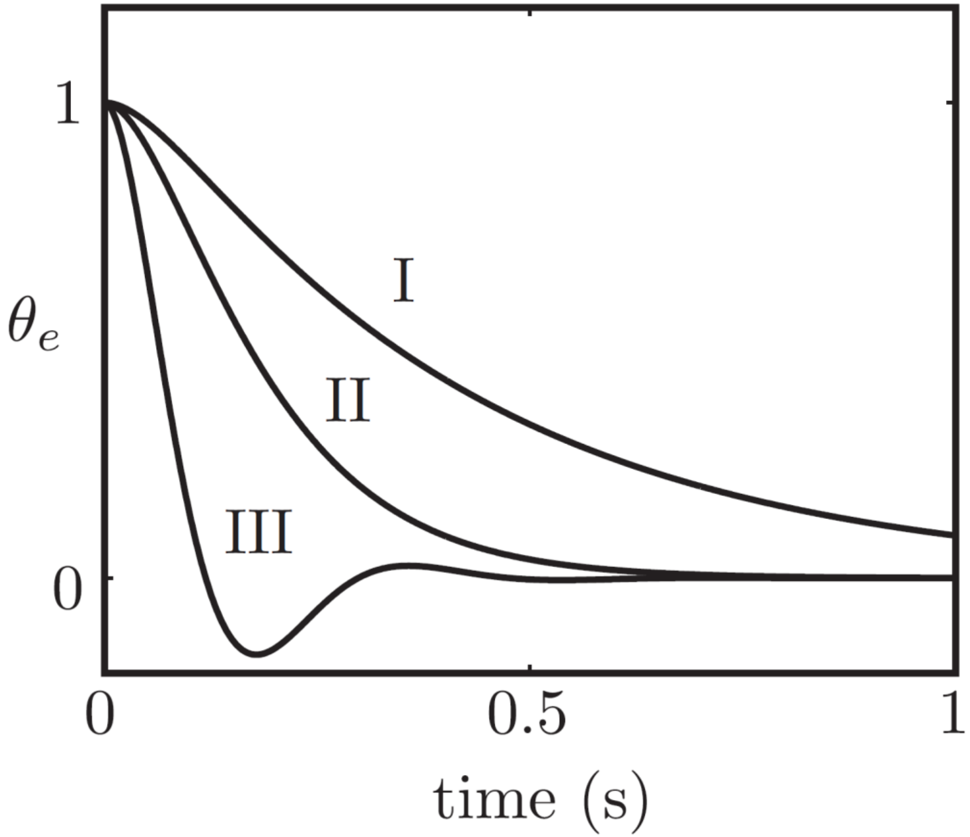
$$\ddot{\theta}_e(t) + K_p\dot{\theta}_e(t) + K_i\theta_e(t) = 0$$

Standard second-order form $\ddot{\theta}_e(t) + 2\zeta\omega_n\dot{\theta}_e(t) + \omega_n^2\theta_e(t) = 0$

natural frequency ω_n damping ratio ζ $\omega_n = \sqrt{K_i}$ $\zeta = K_p/(2\sqrt{K_i})$

PI Controller

Error response



$K_p = 20$ $\quad \zeta = K_p/(2\sqrt{K_i})$

- Overdamped $\zeta = 1.5,\ K_i = 44.4,$ case I
- Critically damped $\zeta = 1,\ K_i = 100,$ case II
- Underdamped $\zeta = 0.5,\ K_i = 400,$ case III

Which one is the best?

Summary

- Robot control
  - Error dynamics

- Motion control
  - P controller
  - PI controller

Further Reading

- Chapter 11 in Kevin M. Lynch and Frank C. Park. Modern Robotics: Mechanics, Planning, and Control. 1st Edition, 2017.