Robot Control: Motion Control

CS 6341 Robotics
Professor Yu Xiang
The University of Texas at Dallas

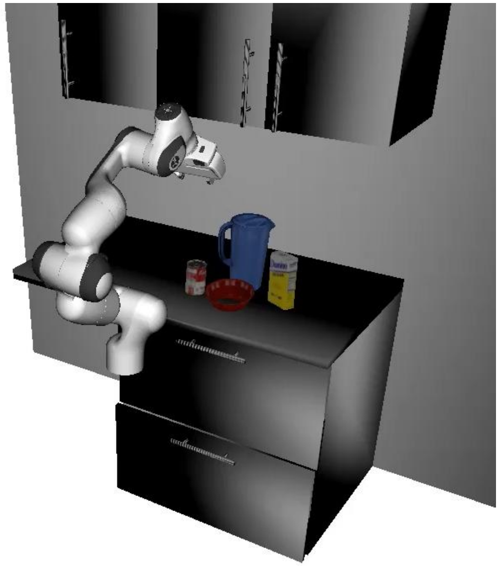
Motion Control

- Goal: follow a given robot trajectory
  - Trajectory of desired end-effector configuration $X_d(t)$
  - Trajectory of desired joint positions $\theta_d(t)$

Can include

$$\dot{\theta}_d(t)$$

$$\ddot{\theta}_d(t)$$



Motion Control

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

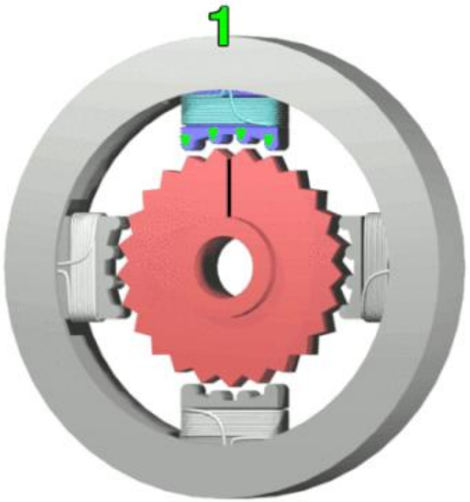
task specifications → Controller → Forces and torques

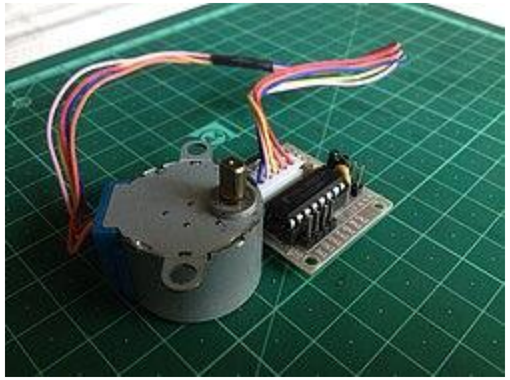
- In some cases, we can assume that there is direct control of the joint velocities

task specifications → Controller → Joint velocities

Stepper Motors

- 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





Stepper motors are best for low-speed, precise motion, but not ideal for high-speed or high-torque robotic applications

- A **stepper motor** moves in **discrete angular steps** (e.g., 1.8° per step)
- Send **step pulses** at a certain rate (say, 1000 pulses per second)
- Angular velocity

$$\omega = (\text{step angle}) \times (\text{pulse rate})$$

$$\omega = 1.8^\circ \times 1000 = 1800^\circ/s = 5 \text{ revolutions per second}$$

Motion Control of a Single Joint

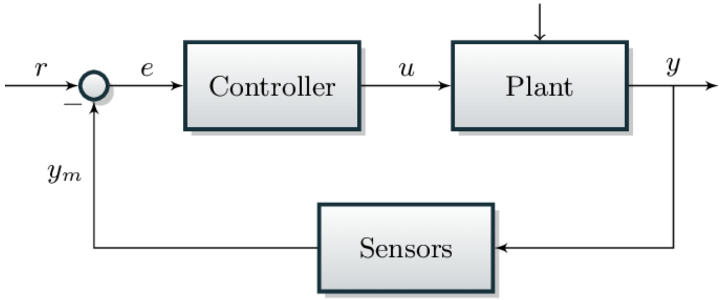
- Feedforward control or open-loop control
  - Given a desired joint trajectory $\theta_d(t)$
  - Choose the velocity command $\dot{\theta}(t) = \dot{\theta}_d(t)$
  - Cons: accumulating position errors

Central difference

$$\dot{\theta}(t_k) \approx \frac{\theta_{k+1} - \theta_{k-1}}{2\Delta t}$$

- In the **real world**, the commanded velocity ≠ actual velocity due to many factors:
  - **Model errors:** friction, backlash, mass uncertainty, actuator nonlinearities
  - **External disturbances:** load variations, gravity effects, contact forces
  - **Motor imperfections:** voltage/current conversion, sensor noise, delay
  - **Numerical drift:** discrete integration errors over time

Imagine you tell a car to drive **exactly 10 m/s for 10 s**, expecting it to move 100 m.
If it actually moves at 9.9 m/s (1% slower), it travels only 99 m — a **1 m error after 10 s**.



- Feedback control
  - Measure the joint position continuously for feedback

Motion Control of a Single Joint

- Proportional controller or P controller

Control rule

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

Control gain

$$K_p > 0$$

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

Setpoint control

  - Error dynamics

$$\dot{\theta}_e(t) = \cancelto{0}{\dot{\theta}_d(t)} - \dot{\theta}(t)$$

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

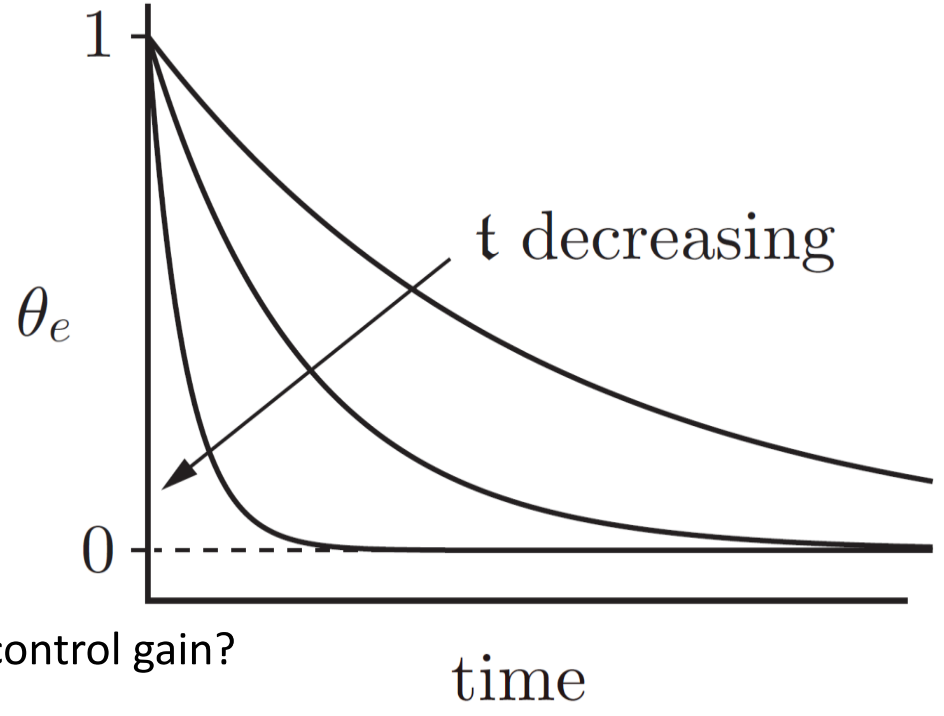
First-Order Error Dynamics

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 $4/K_p$

How shall we choose the control gain?

Larger K_p is better

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

velocity setpoint control

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

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

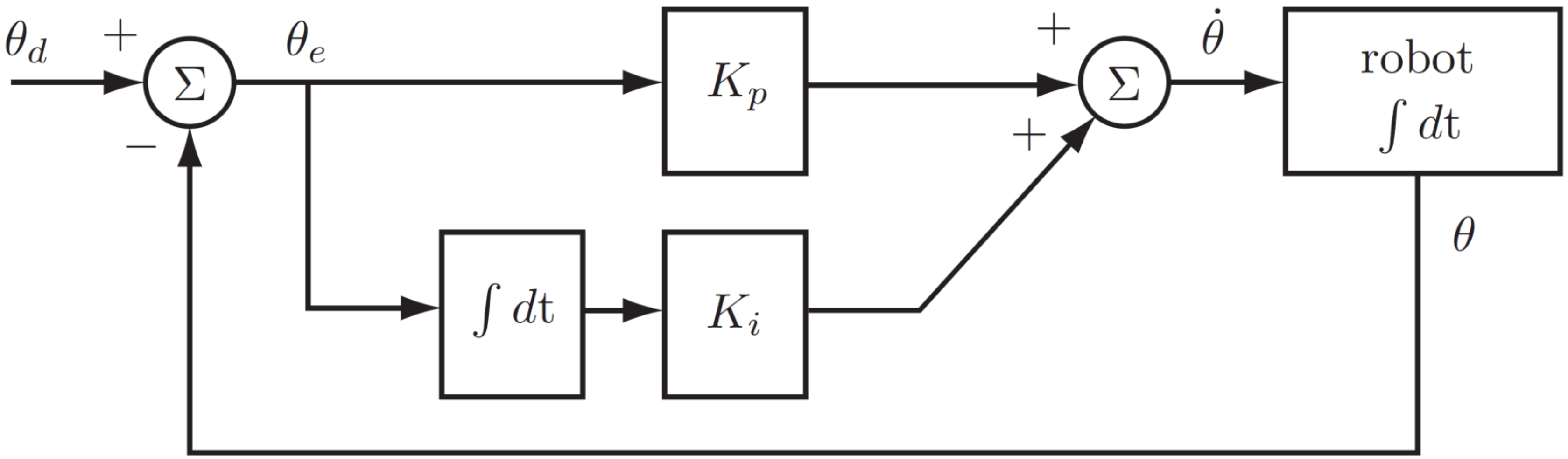
We cannot make K_p arbitrarily large (velocity limit, instability)

PI Controller

- A proportional-integral controller

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

Time-integral of the error



PI Controller

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

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

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

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

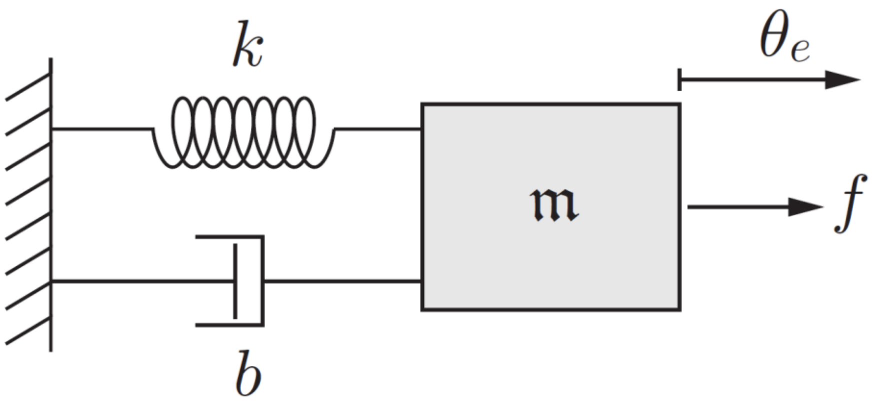
$$\ddot{\theta}_e(t) + K_p \dot{\theta}_e(t) + K_i \theta_e(t) = 0$$

Second-Order Error Dynamics

PI Controller

- Mass-spring-damper

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



Natural frequency $\omega_n = \sqrt{\frac{k}{m}}$

Damping ratio $\zeta = \frac{b}{2\sqrt{km}}$ (zeta)

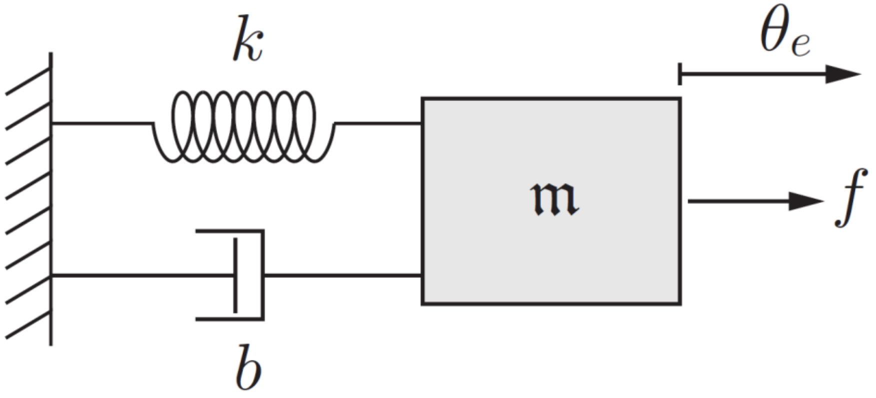
$$\frac{k}{m} = \omega_n^2$$

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

$$\frac{b}{m} = 2\zeta\omega_n$$

PI Controller

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



$$\ddot{\theta}_e(t) + 2\zeta\omega_n\dot{\theta}_e(t) + \omega_n^2\theta_e(t) = 0$$

Standard second-order form

natural frequency ω_n

damping ratio ζ

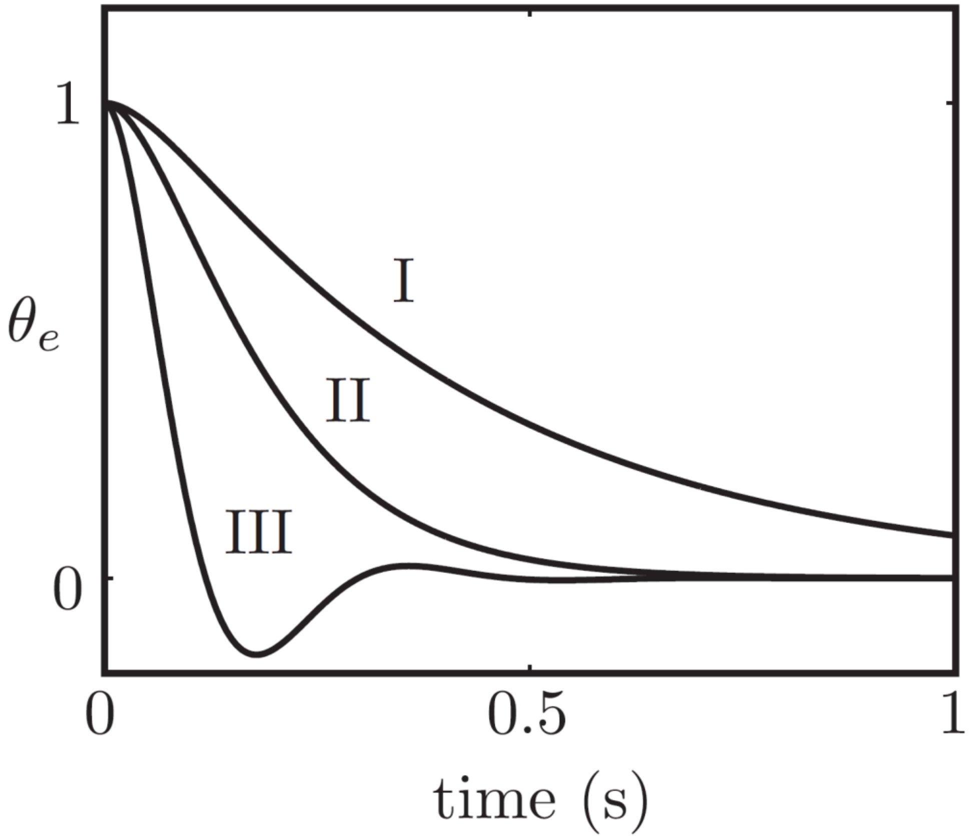
$$\ddot{\theta}_e(t) + K_p\dot{\theta}_e(t) + K_i\theta_e(t) = 0$$

$$\omega_n = \sqrt{K_i}$$

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

PI Controller

Error response



Damping Ratio (ζ)	Behavior	Motion	Damped Frequency
$\zeta = 0$	Undamped	Pure oscillation	ω_n
$0 < \zeta < 1$	Underdamped	Oscillates and decays	$\omega_n\sqrt{(1-\zeta^2)}$
$\zeta = 1$	Critically damped	No oscillation, fastest return	0
$\zeta > 1$	Overdamped	No oscillation, slow return	*Imaginary* (no real ω_n)

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

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

Which one is the best?

Motion Control with Velocity Inputs

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

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

- Proportional controller or P controller

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

Control gain

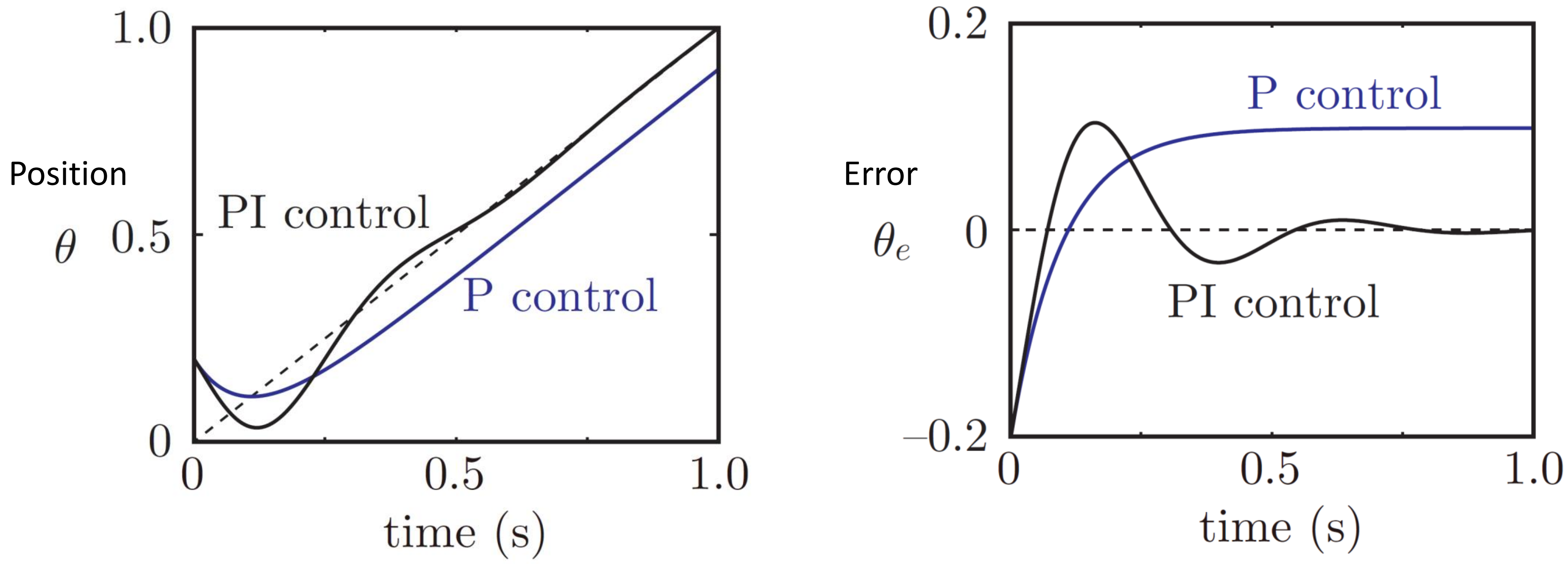
$$K_p > 0$$

- Proportional-integral controller or PI controller

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

Comparison between P Controller and PI Controller

$$\dot{\theta}_d(t) = c$$ velocity setpoint control



Reference trajectory (dashed)

Feedforward Plus Feedback Control

- Feedback control: an error is required before the joint begins to move

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

- Feedforward plus feedback control: Initiate motion before any error accumulates

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

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

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

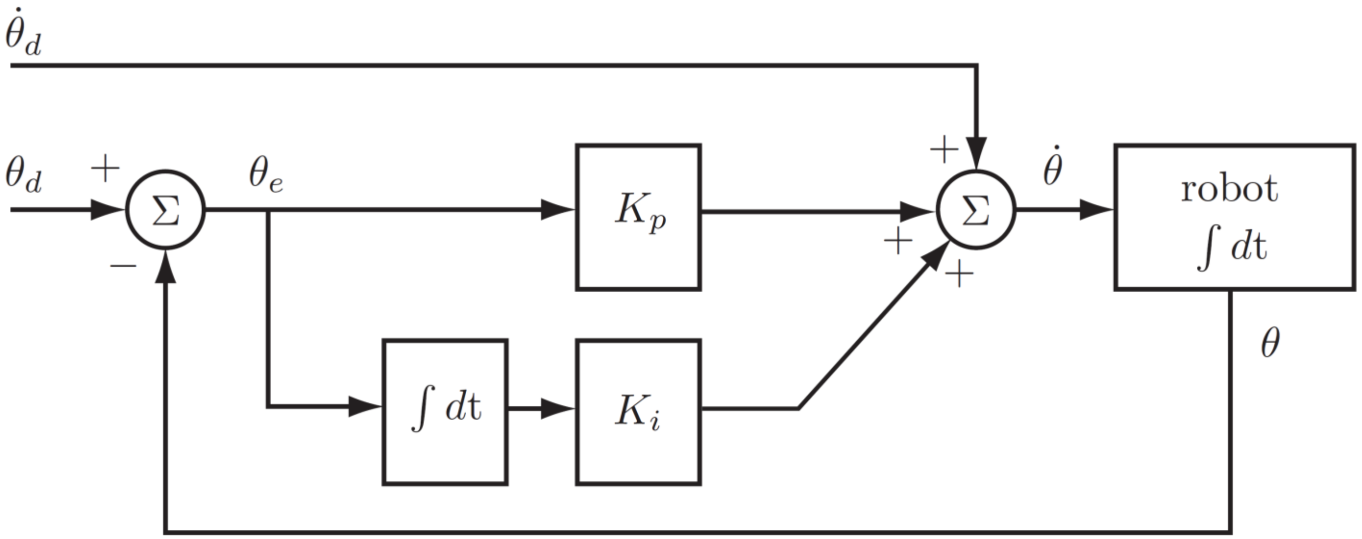
The same error dynamics as the feedback PI controller

Incorporate the reference joint velocity

Feedforward Plus Feedback Control

- Feedforward plus feedback control: Initiate motion before any error accumulates

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



Preferred control law for producing a commanded velocity to the joint

Motion Control of Multi-Joint Robots

- Reference position $\theta_d(t)$ and actual position $\theta(t)$ n dimensional vector
- Gains K_p K_i $n \times n$ matrix

$$k_p I \qquad k_i I$$

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

Task-Space Motion Control

$$[\mathcal{V}_b] = \begin{bmatrix} [\omega_b] & v_b \\ 0 & 0 \end{bmatrix} = T^{-1}\dot{T}$$

- Reference trajectory as end-effecter configuration $X_d(t) \in \mathbb{SE}(3)$

- Reference twist $\mathcal{V}_d(t)$
  - Continuous formula $[\mathcal{V}_d(t)] = X_d^{-1}(t)\dot{X}_d(t)$
  - Discrete formula $[\mathcal{V}_d(t_k)] = \frac{1}{\Delta t}\log(X_d^{-1}(t_k)X_d(t_{k+1}))$

- Similarly, the configuration of the end-effector $X(t) \in SE(3)$

- End-effector twist

$$[\mathcal{V}(t)] = X^{-1}(t)\dot{X}(t) \quad [\mathcal{V}(t_k)] = \frac{1}{\Delta t}\log(X^{-1}(t_k)X(t_{k+1}))$$

Task-Space Motion Control

$$[\mathrm{Ad}_T] = \begin{bmatrix} R & 0 \\ [p]R & R \end{bmatrix} \in \mathbb{R}^{6\times 6}$$

- Joint-space control law

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

$$\mathcal{V}(t) \qquad \mathcal{V}_d(t)$$

$$[X_e] = \log(X^{-1}X_d) \qquad X_{sb} \;\; X_{sd}$$

task-space control law

$$\mathcal{V}_b(t) = [\mathrm{Ad}_{X^{-1}X_d}]\mathcal{V}_d(t) + K_p X_e(t) + K_i \int_0^t X_e(\mathrm{t})\, dt$$

$K_p, K_i \in \mathbb{R}^{6\times 6}$ Commanded joint velocities $\dot{\theta} = J_b^{\dagger}(\theta)\mathcal{V}_b$

Motion Control with Velocity Inputs

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

- Limited to applications with low or predictable force-torque requirements

- Do not make use of a dynamic model of the robot

Summary

- Motion control with velocities
  - P controller
  - PI controller
  - Feedforward plus feedback controller
  - Task-Space Motion Control

Further Reading

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