



Aalto University
School of Electrical
Engineering

ELEC-E8126: Robotic Manipulation

Motion control

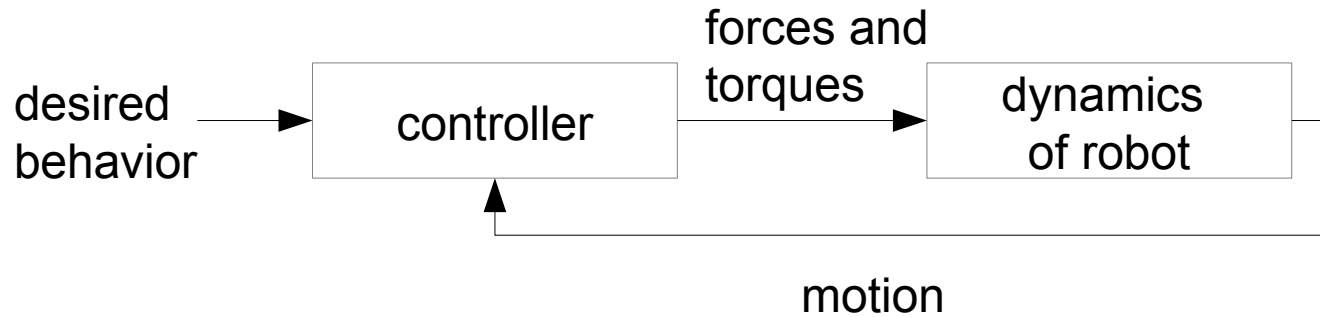
Ville Kyrki

8.2.2021

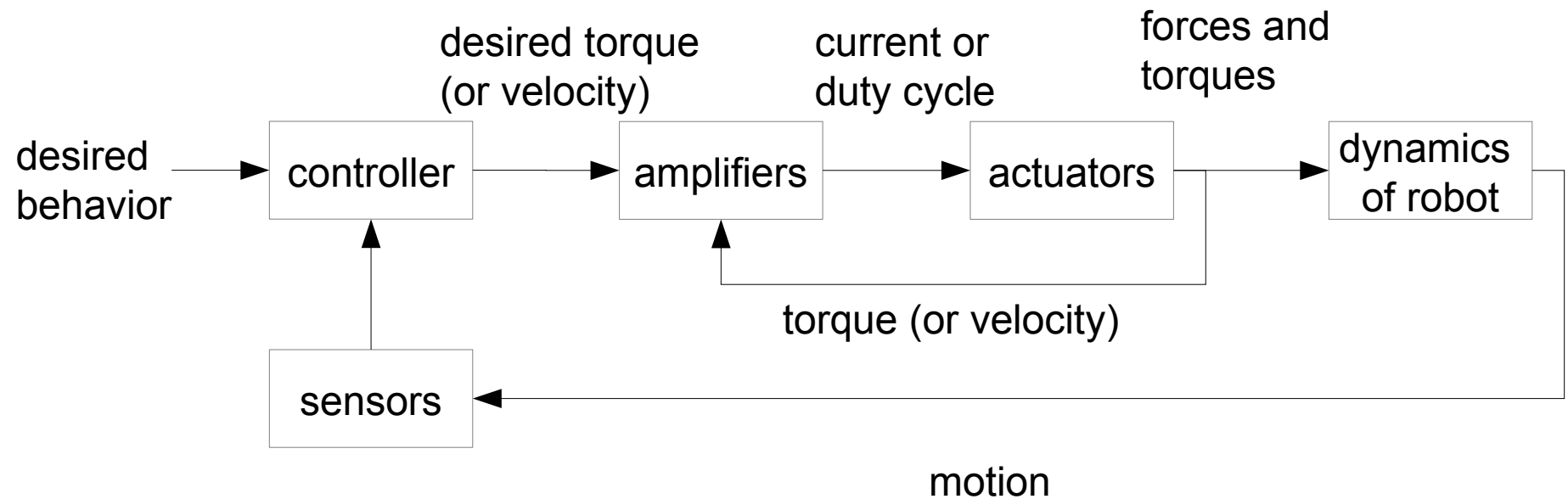
Learning goals

- Understand basic approaches of robot motion control.
- Understand structure of dynamics of serial kinematic chains such as robot arms.

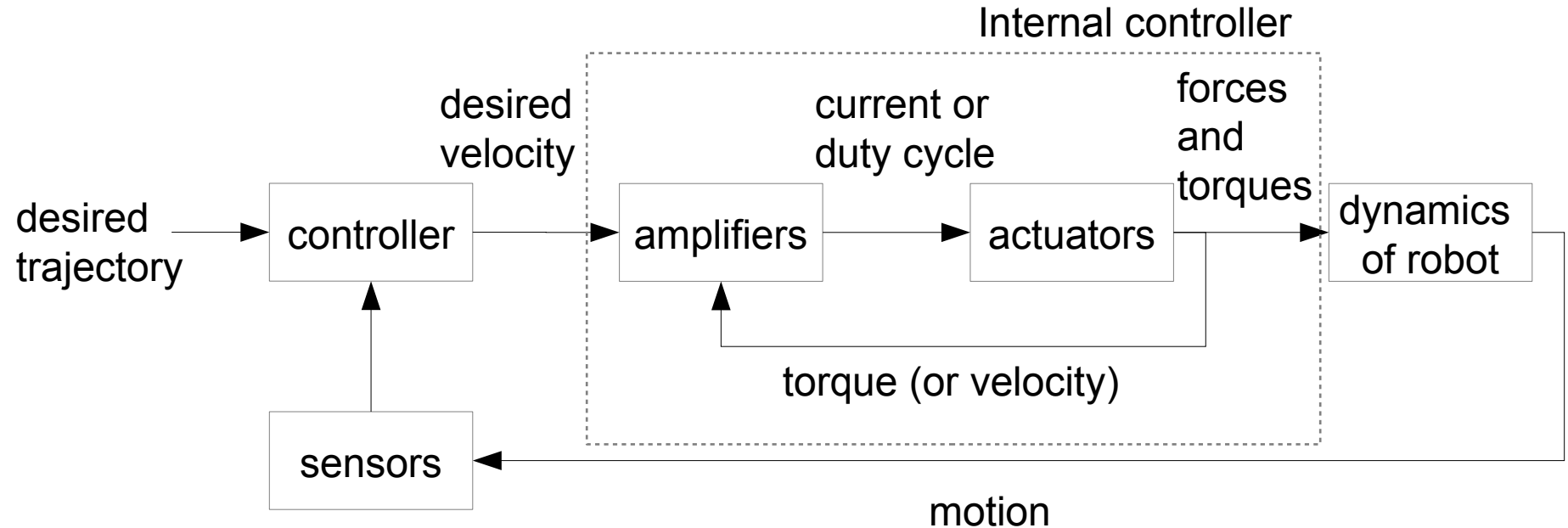
Control – general structure



Control – typical real structure

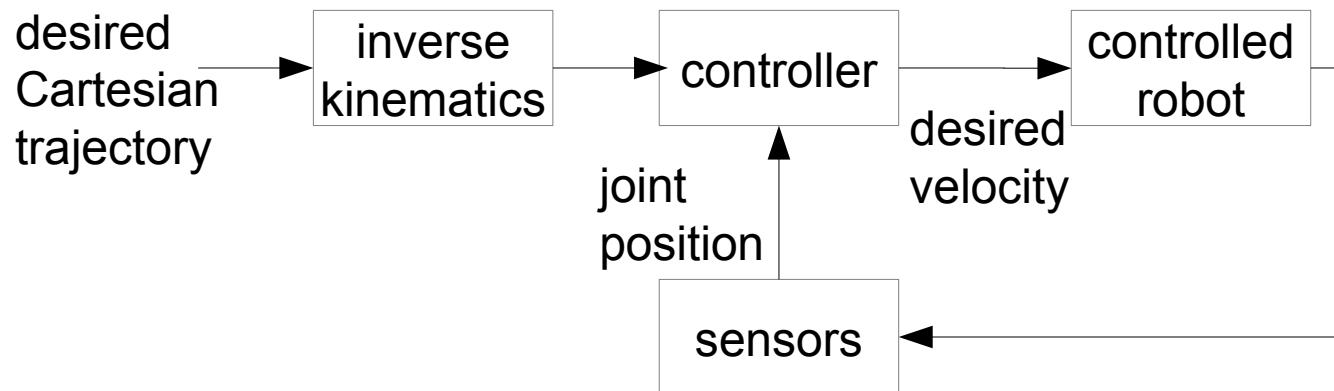
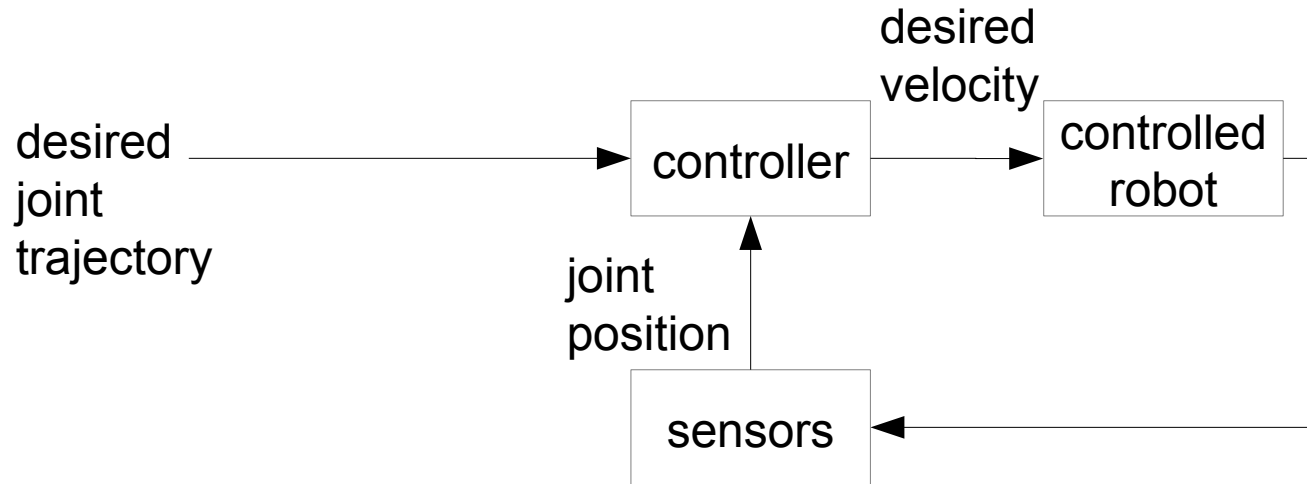


Joint velocity control

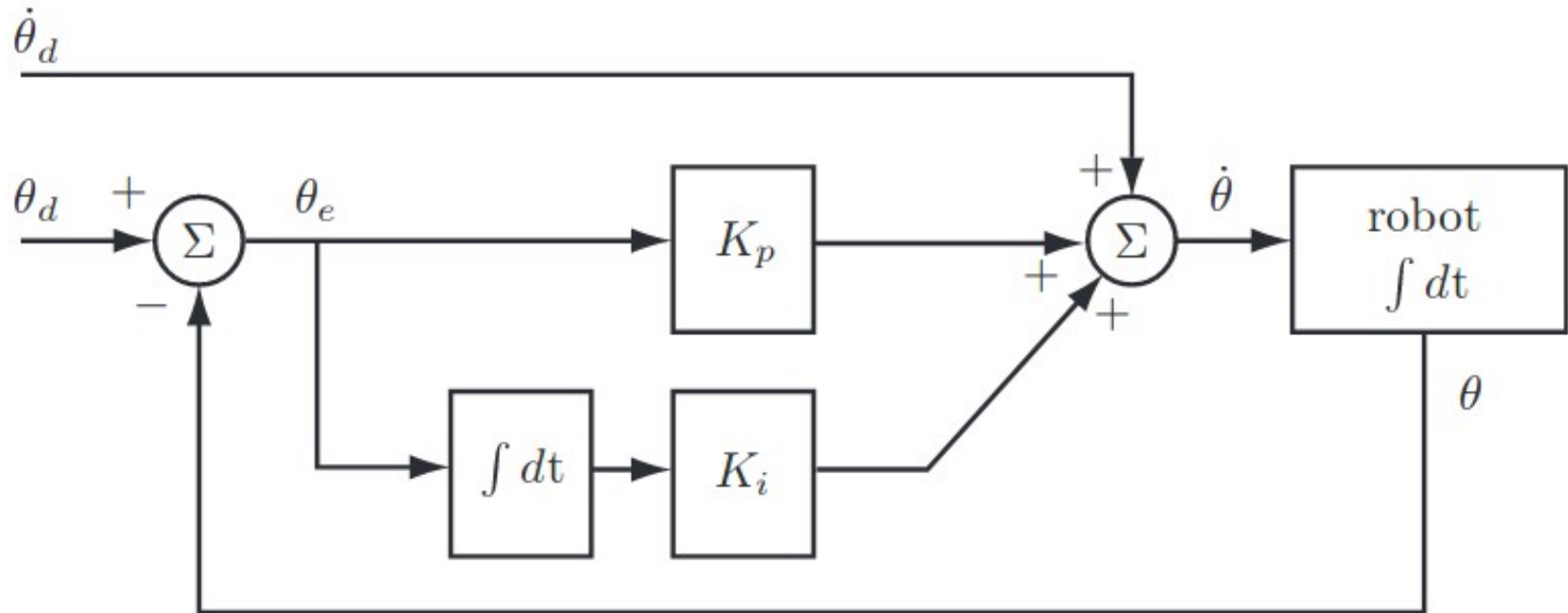


Joint velocity control

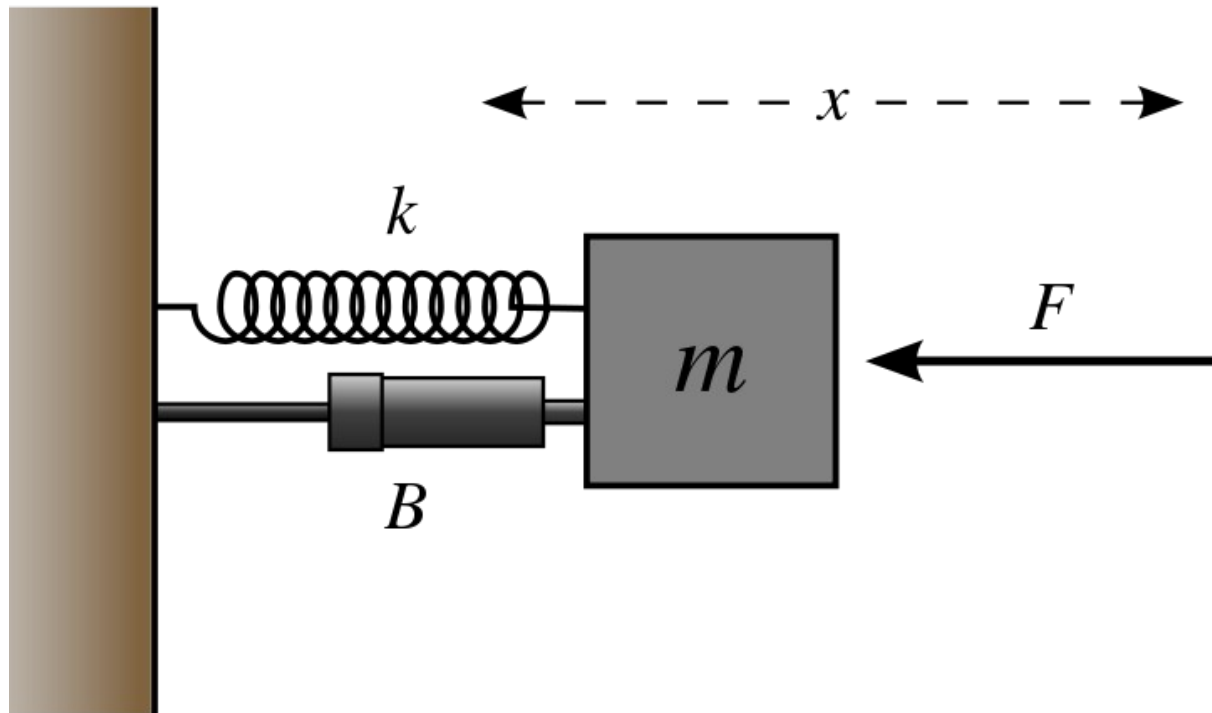
Assume internal controller tracks velocity accurately.



Practical controller: PI with feedforward



How does the system below behave?



Cartesian space control

$$\begin{bmatrix} \omega_b(t) \\ \dot{p}(t) \end{bmatrix} = \begin{bmatrix} R^T(t)R_d(t) & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \omega_d(t) \\ \dot{p}_d(t) \end{bmatrix} + K_p X_e(t) + K_i \int_0^t X_e(t) dt, \dots$$

pseudoinverse

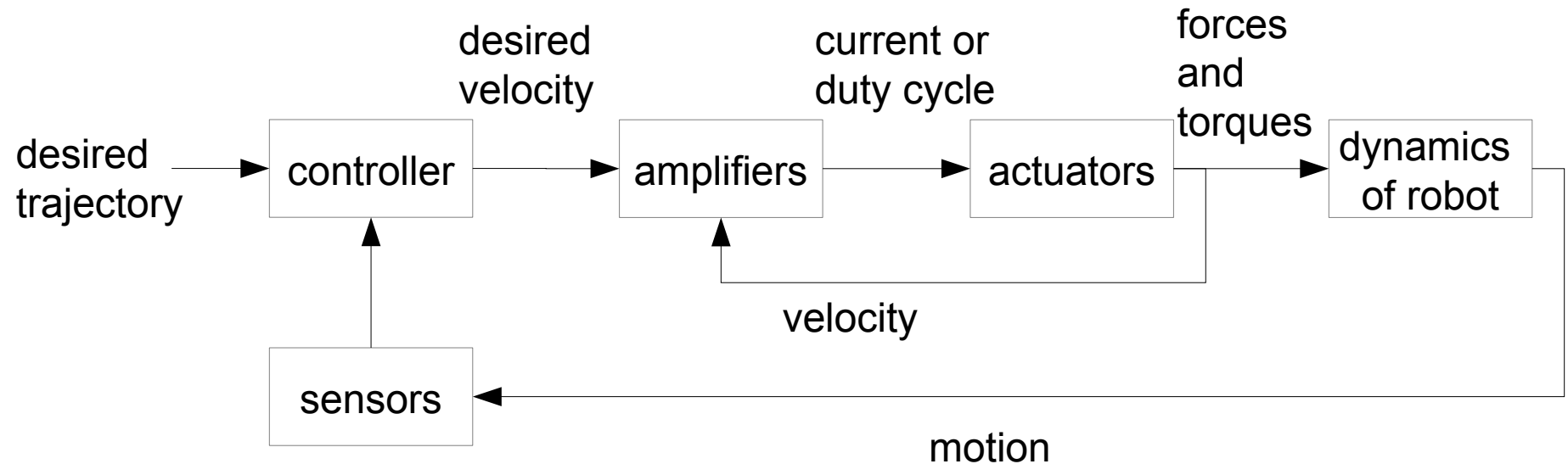
$$\dot{\theta} = J_{ee}^+ \begin{bmatrix} \omega_b(t) \\ \dot{p}(t) \end{bmatrix}$$

end-effector Jacobian

$$X_e(t) = \begin{bmatrix} \log(R^T(d)R_d(t)) \\ p_d(t) - p(t) \end{bmatrix}$$

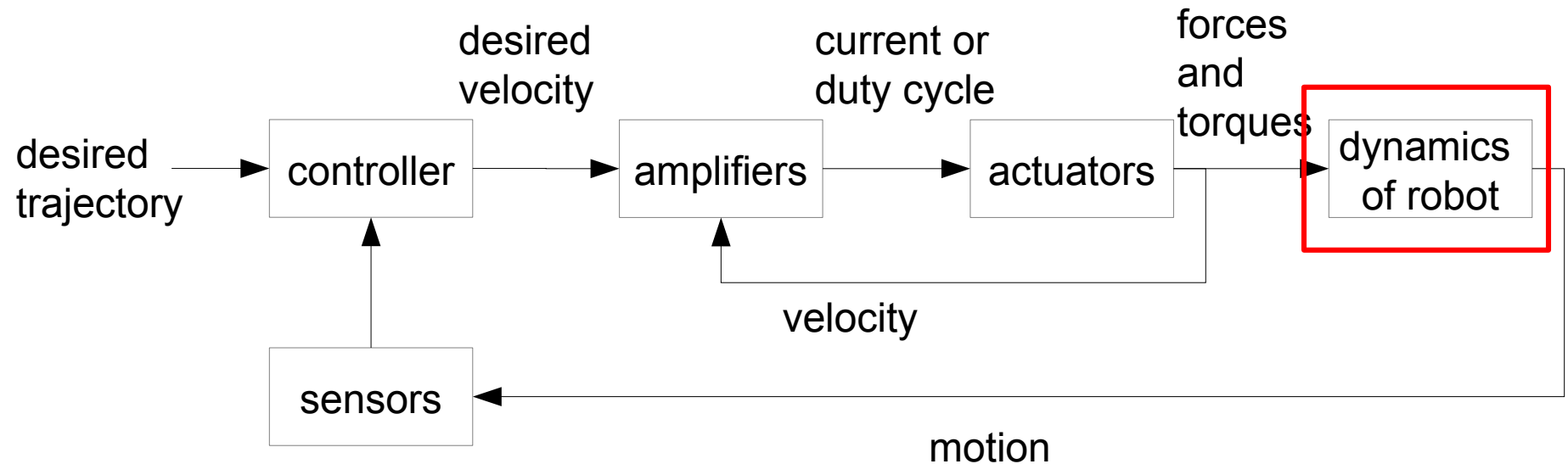
Toward torque control

Can this model control force interactions?



Toward torque control

Can this model control force interactions?



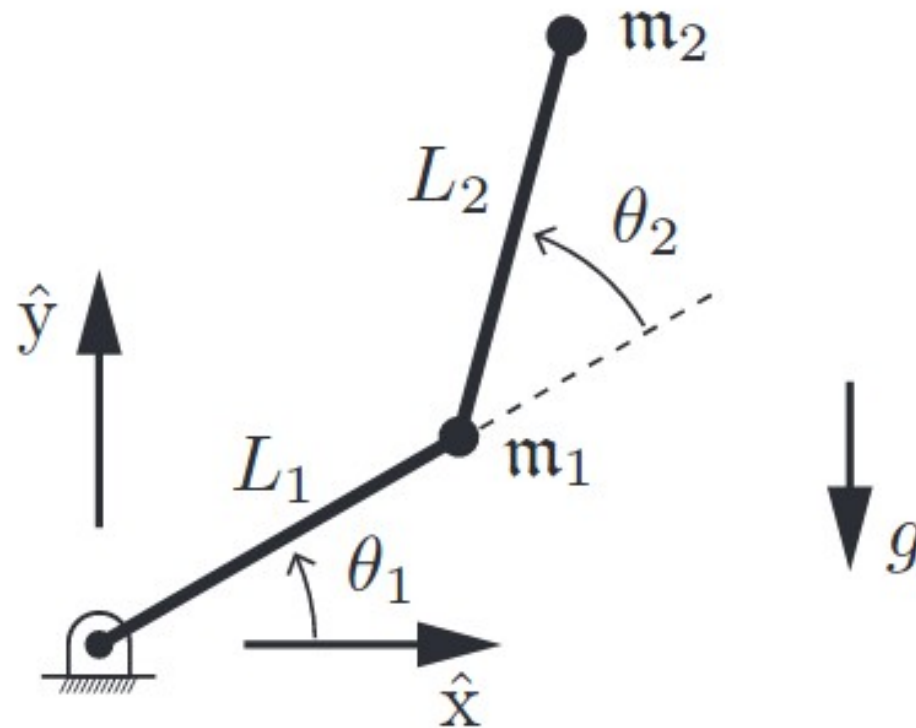
Dynamics

- Represent response to motor/joint torques
- Equation of motion

$$\text{torques} \longrightarrow \tau = M(\theta) \ddot{\theta} + h(\theta, \dot{\theta}) \longleftarrow \begin{array}{l} \text{gravity,} \\ \text{friction,} \\ \text{centripetal,} \\ \text{Coriolis} \end{array}$$

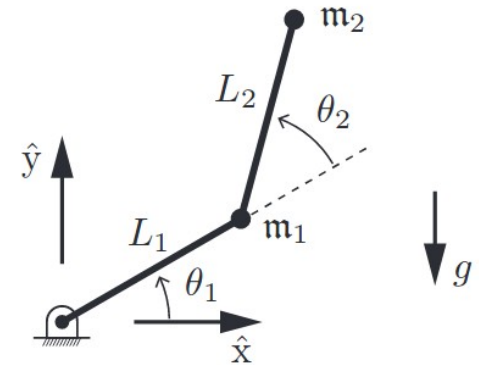
↑
mass matrix

Example: 2R robot under gravity



$$\boldsymbol{\tau} = \mathbf{M}(\boldsymbol{\theta}) \ddot{\boldsymbol{\theta}} + \mathbf{c}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{g}(\boldsymbol{\theta})$$

Example: 2R robot



$$\boldsymbol{\tau} = \mathbf{M}(\boldsymbol{\theta}) \ddot{\boldsymbol{\theta}} + \mathbf{c}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{g}(\boldsymbol{\theta})$$

$$\mathbf{M}(\boldsymbol{\theta}) = \begin{bmatrix} m_1 L_1^2 + m_2 (L_1^2 + 2 L_1 L_2 \cos \theta_2 + L_2^2) & m_2 (L_1 L_2 \cos \theta_2 + L_2^2) \\ m_2 (L_1 L_2 \cos \theta_2 + L_2^2) & m_2 L_2^2 \end{bmatrix}$$

$$\mathbf{c}(\dot{\boldsymbol{\theta}}, \boldsymbol{\theta}) = \begin{bmatrix} -m_2 L_1 L_2 \sin \theta_2 (\dot{\theta}_2^2 + 2 \dot{\theta}_1 \dot{\theta}_2) \\ m_2 L_1 L_2 \dot{\theta}_1^2 \sin \theta_2 \end{bmatrix}$$

$$\mathbf{g}(\boldsymbol{\theta}) = \begin{bmatrix} (m_1 + m_2) g L_1 \cos \theta_1 + m_2 g L_2 \cos(\theta_1 + \theta_2) \\ m_2 g L_2 \cos(\theta_1 + \theta_2) \end{bmatrix}$$

Forward dynamics with contact force

What's this? And why?

$$\boldsymbol{\tau} = \mathbf{M}(\boldsymbol{\theta}) \ddot{\boldsymbol{\theta}} + \mathbf{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \mathbf{J}^T(\boldsymbol{\theta}) \mathbf{F}_{tip}$$

← contact force

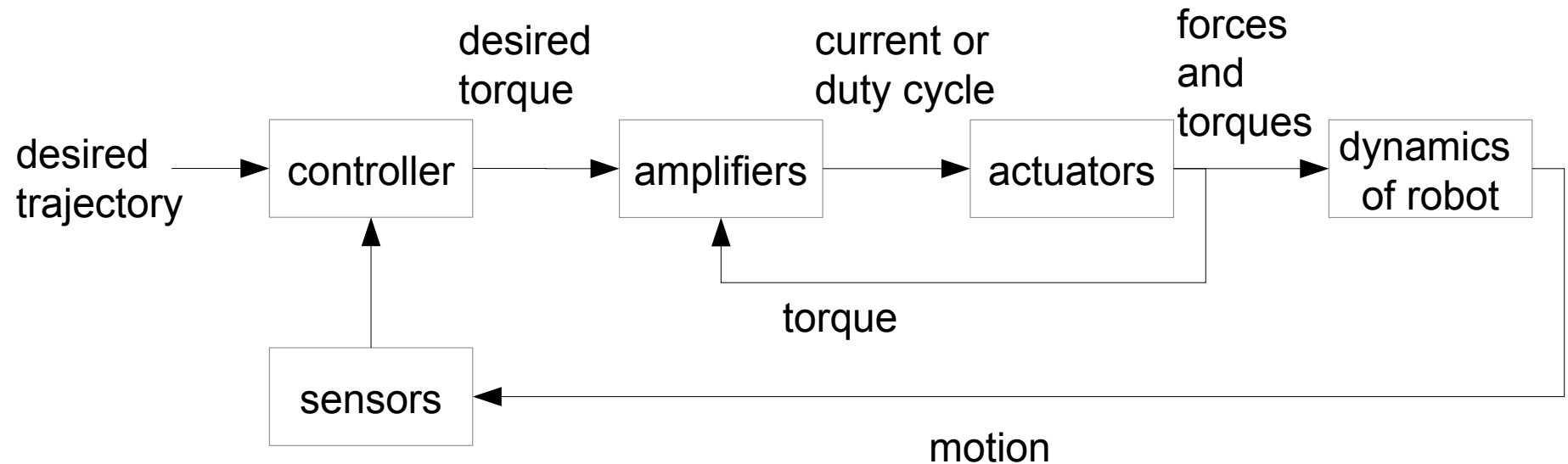
Solve:

$$\mathbf{M}(\boldsymbol{\theta}) \ddot{\boldsymbol{\theta}} = \boldsymbol{\tau} - \mathbf{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) - \mathbf{J}^T(\boldsymbol{\theta}) \mathbf{F}_{tip}$$

linear system of equations

Torque control

Can this model control force interactions?

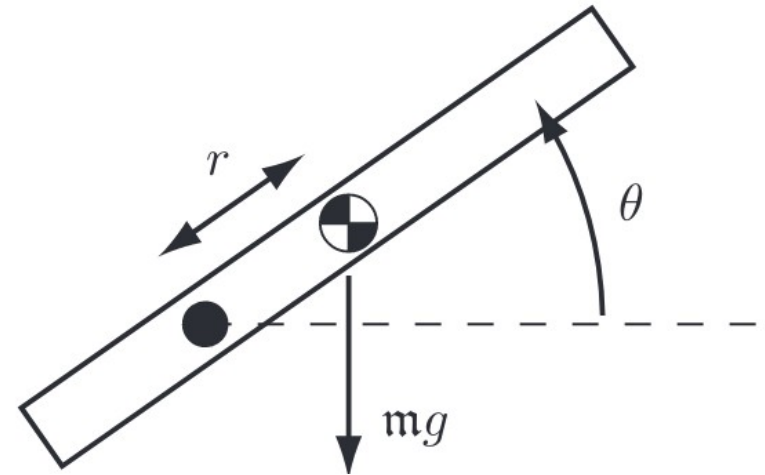


Single joint torque control

- Dynamics with (simple) friction

$$\tau = M \ddot{\theta} + m g r \cos \theta + b \dot{\theta}$$

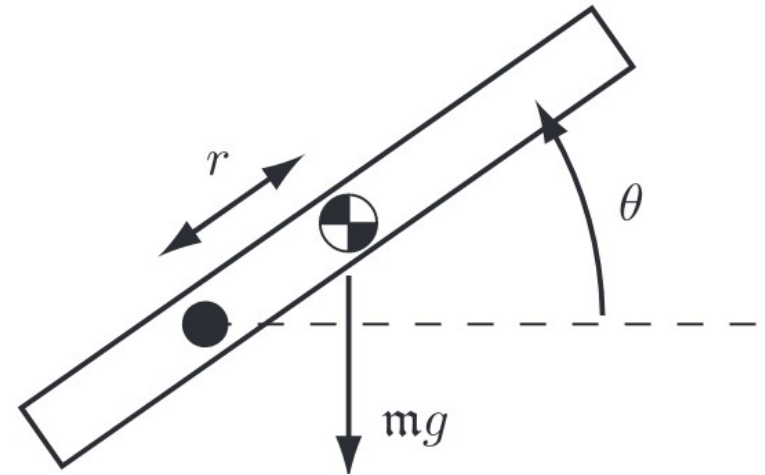
Assume you have a trajectory to follow.
Propose a controller. Or several.



Single joint torque control

- Dynamics with (simple) friction

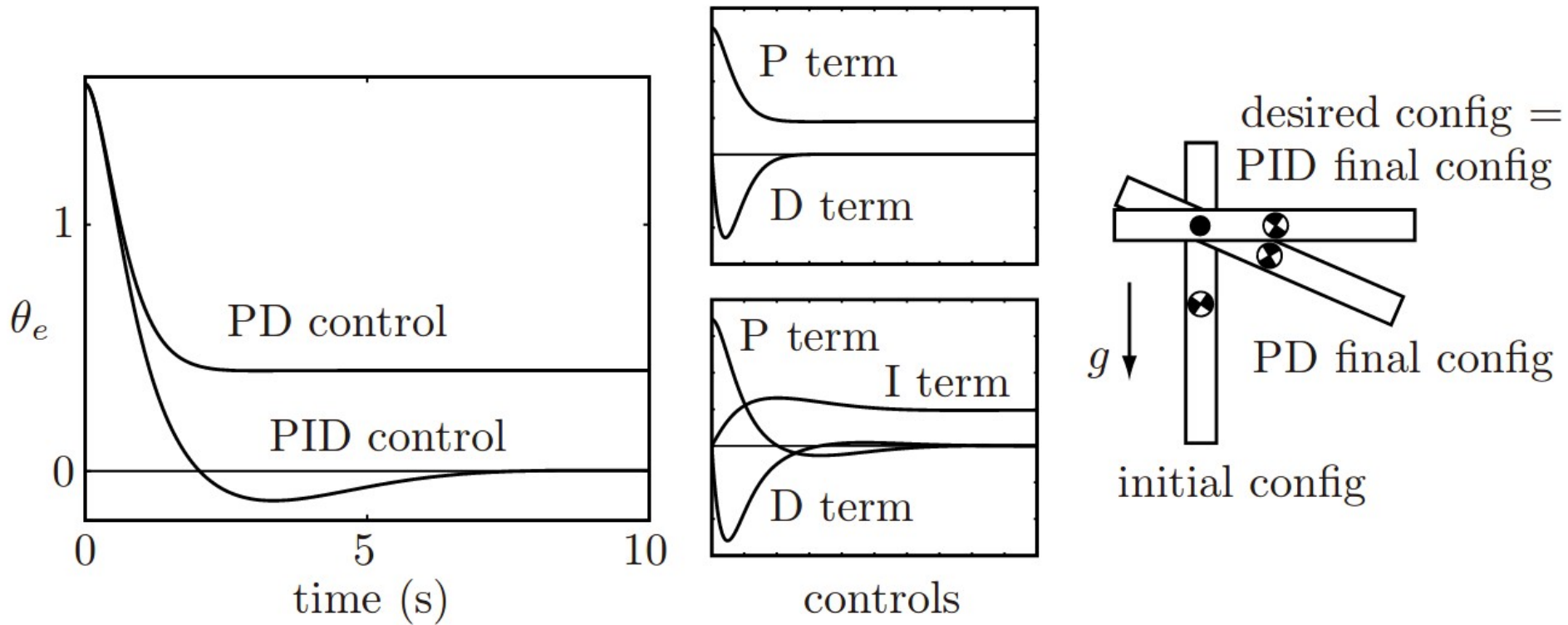
$$\tau = M \ddot{\theta} + m g r \cos \theta + b \dot{\theta}$$



Assume you have a trajectory to follow.
Propose a controller.

Does your controller converge to zero error if desired state is constant?

PID convergence

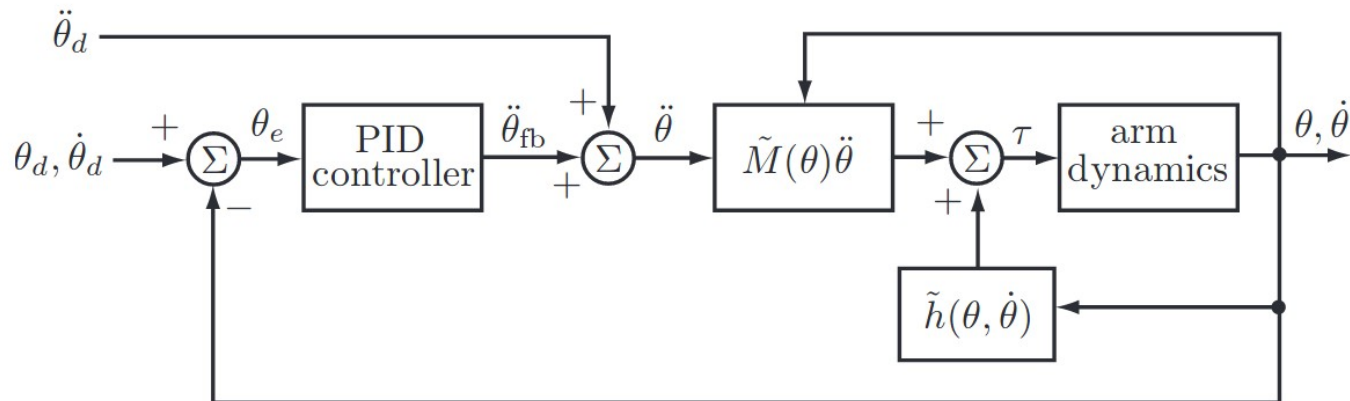


Inverse dynamics / computed torque control

$$\tau = \hat{M}(\theta) \ddot{\theta}_d + \underbrace{K_p \theta_e + K_i \int \theta_e + K_d \dot{\theta}_e}_{\text{PID feedback}} + \hat{h}(\theta, \dot{\theta})$$

dynamics compensation

feedforward PID feedback



Inverse dynamics

- Problem: Calculate right hand side of

$$\boldsymbol{\tau} = \mathbf{M}(\boldsymbol{\theta}) \ddot{\boldsymbol{\theta}} + \mathbf{h}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$$

- Informally: If I want to follow a certain trajectory, how high torques do I need to apply at joints.
- Solution: Calculate \mathbf{M} and \mathbf{h} by Newton-Euler algorithm.

Cartesian space dynamics

- If Jacobian is invertible, dynamics can be expressed in Cartesian space as

$$\mathbf{F} = \mathbf{M}_C(\boldsymbol{\theta}) \ddot{\mathbf{x}} + \mathbf{h}_C(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$$

Cartesian force \rightarrow \mathbf{F}

Cartesian acceleration \rightarrow $\ddot{\mathbf{x}}$

Cartesian dynamics parameters can be calculated using joint space dynamics + Jacobian.
E.g.

$$\mathbf{M}_C(\boldsymbol{\theta}) = \mathbf{J}^{-T} \mathbf{M}(\boldsymbol{\theta}) \mathbf{J}^{-1}$$

- Furthermore, if inverse kinematics is unique, dynamics can be expressed in Cartesian space as

$$\mathbf{F} = \mathbf{M}_C(\mathbf{x}) \ddot{\mathbf{x}} + \mathbf{h}_C(\mathbf{x}, \dot{\mathbf{x}})$$

Cartesian control

$$F = M_C(\mathbf{x}) \ddot{\mathbf{x}} + \mathbf{h}_C(\mathbf{x}, \dot{\mathbf{x}})$$

- Inverse dynamics controller can then be written also in Cartesian space (for a robot with unique inverse dyn.).

$$\boldsymbol{\tau} = J^T(\boldsymbol{\theta}) \underbrace{\left(M_C(\mathbf{x}) \left(\ddot{\mathbf{x}}_d + K_p \mathbf{x}_e + K_i \int \mathbf{x}_e + K_d \dot{\mathbf{x}}_e \right) + \mathbf{h}_C(\mathbf{x}, \dot{\mathbf{x}}) \right)}_{\text{Cartesian force}}$$

Compare to

$$\boldsymbol{\tau} = \hat{M}(\boldsymbol{\theta}) \left(\ddot{\boldsymbol{\theta}}_d + K_p \boldsymbol{\theta}_e + K_i \int \boldsymbol{\theta}_e + K_d \dot{\boldsymbol{\theta}}_e \right) + \hat{\mathbf{h}}(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$$

Summary

- Accurate motion control requires knowledge (model) of robot dynamics.
- Good recipe: inverse dynamics + PID + feedforward (computed torque control).

Next time: Control with external forces

- Readings:
 - Lynch & Park, Chapter 11.5-11.6