Aalto University
School of Electrical Engineering

ELEC-E8126: Robotic Manipulation
Motion control

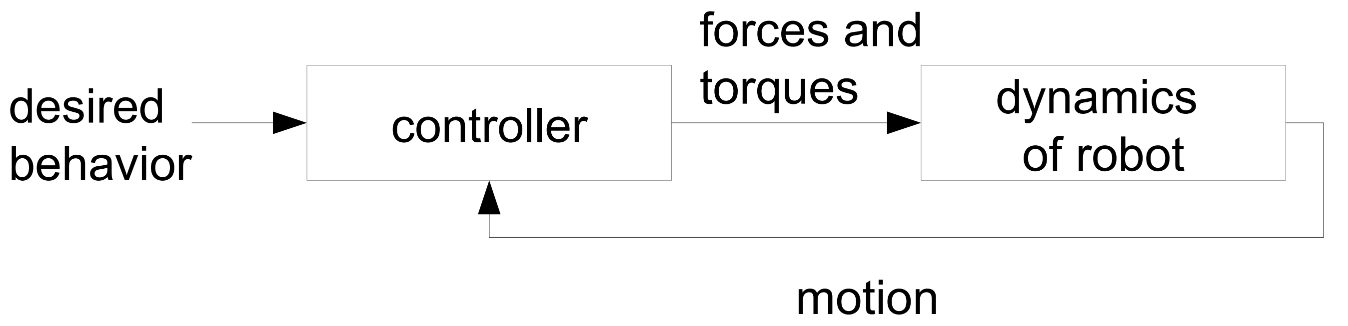
Ville Kyrki

31.1.2022

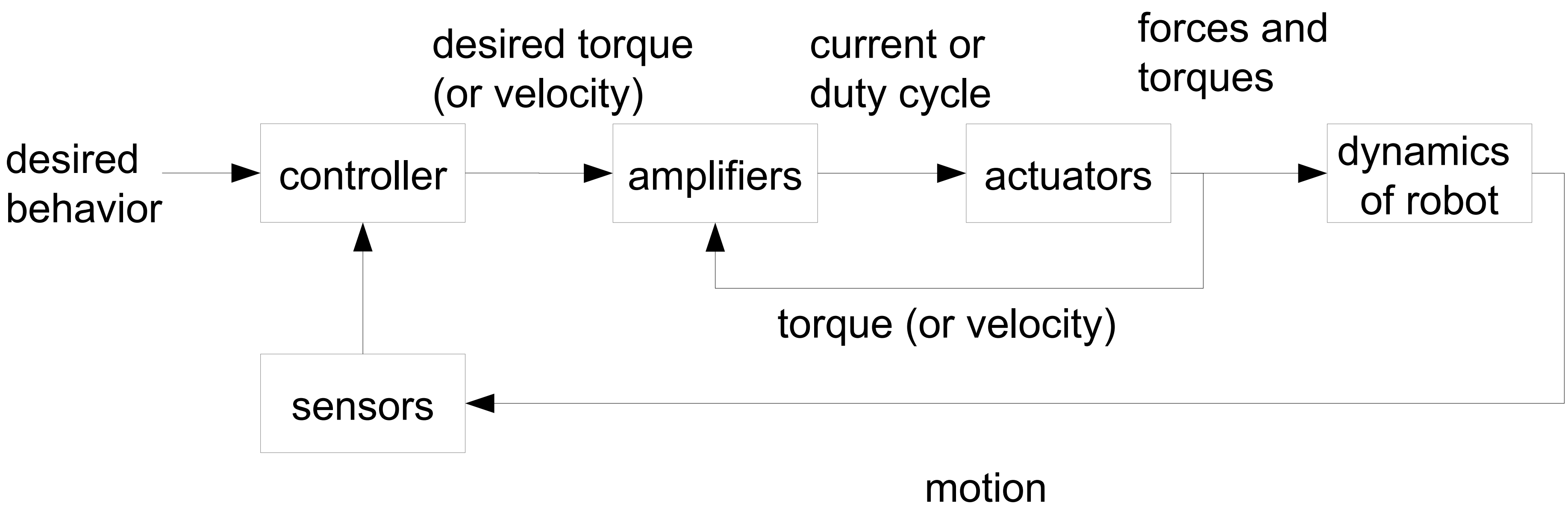
Learning goals

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

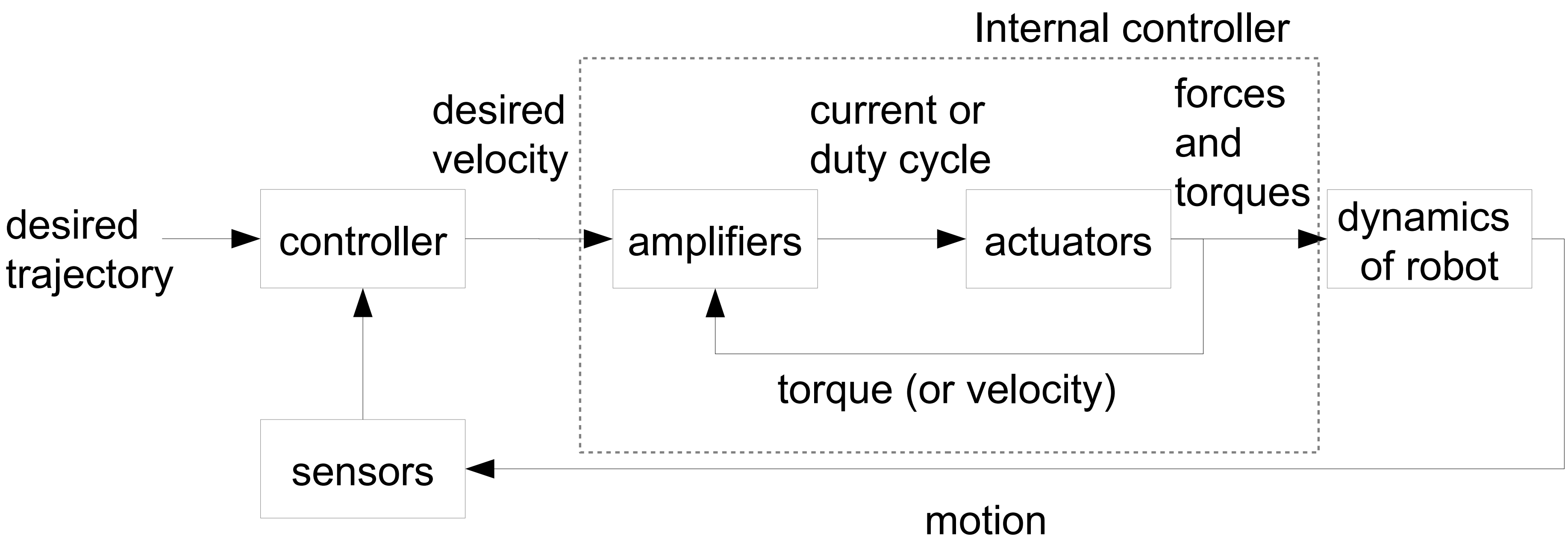
Control – general structure



Control – typical real structure

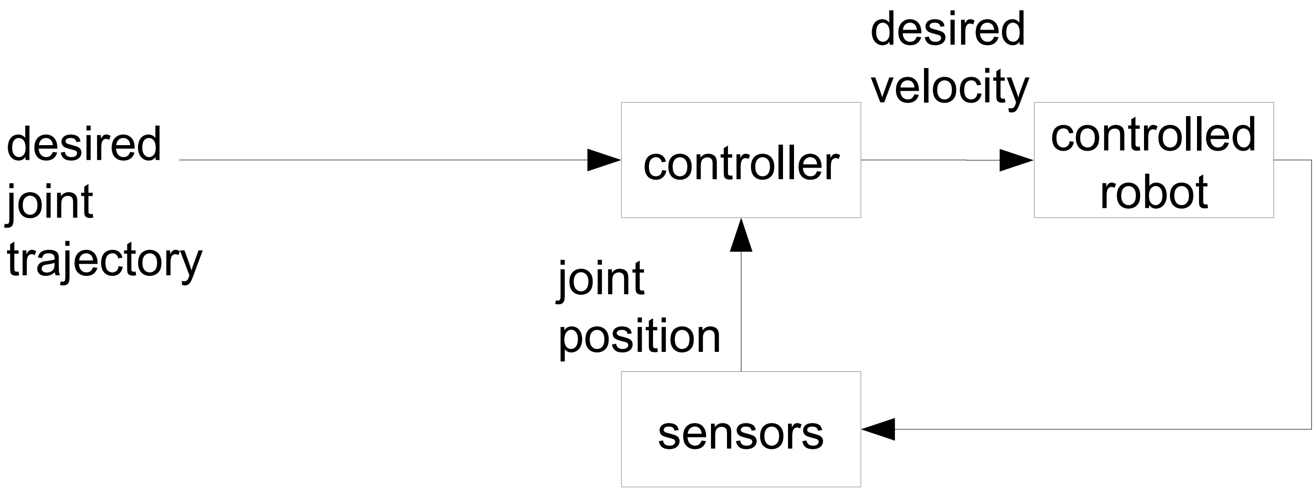


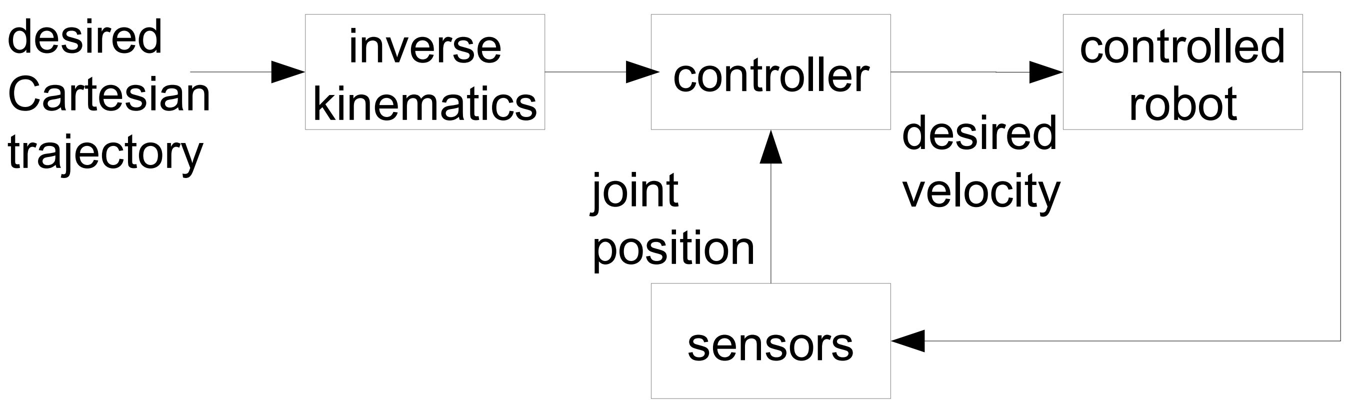
Joint velocity control



Joint velocity control

Assume internal controller tracks velocity accurately.

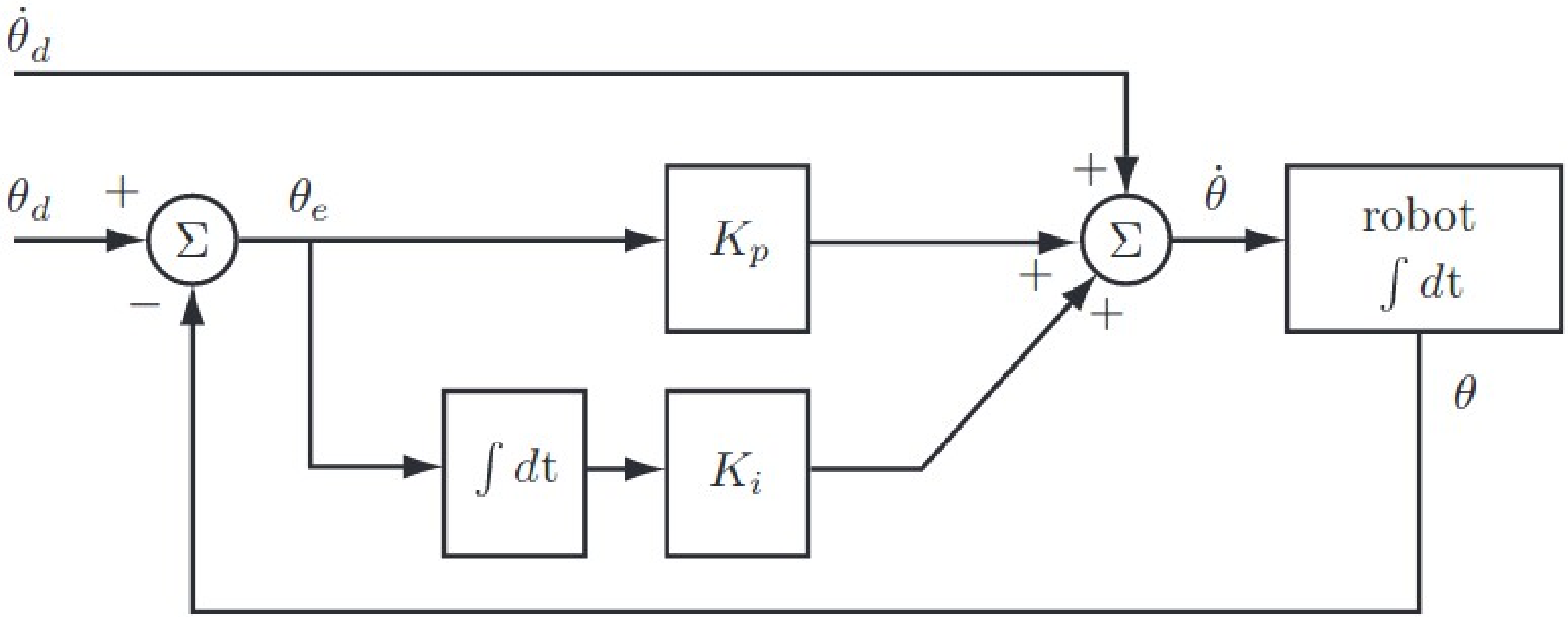




These may be the case with an industrial robot.

Give an example controller!

Practical controller: PI with feedforward



What kind of dynamics does this system have?

Cartesian space control

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

pseudoinverse

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

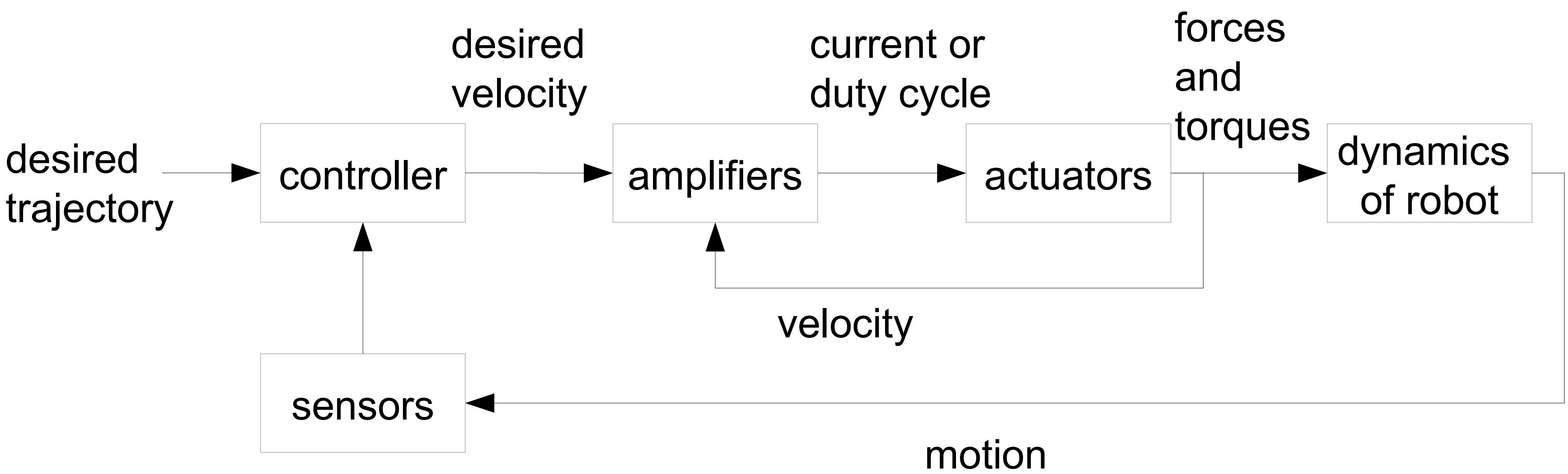
end-effector Jacobian

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

How does the Cartesian trajectory look like?

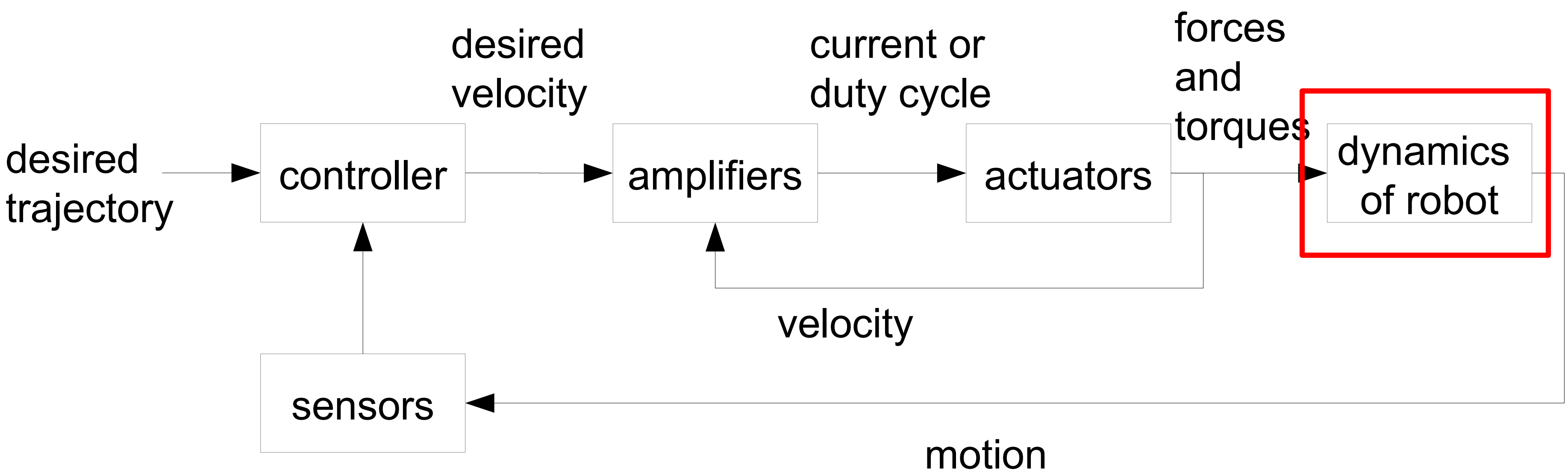
Toward torque control

Can this model control force interactions?

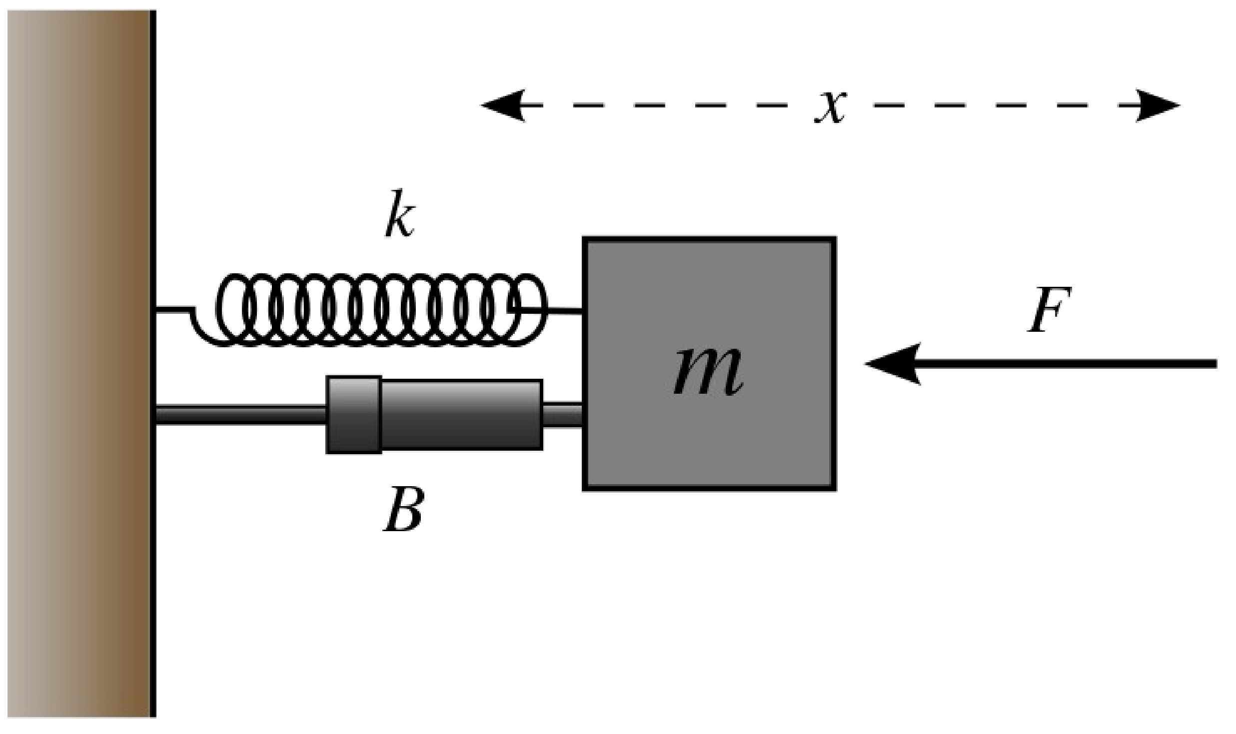


Toward torque control

Can this model control force interactions?



How does the system below behave?



Dynamics

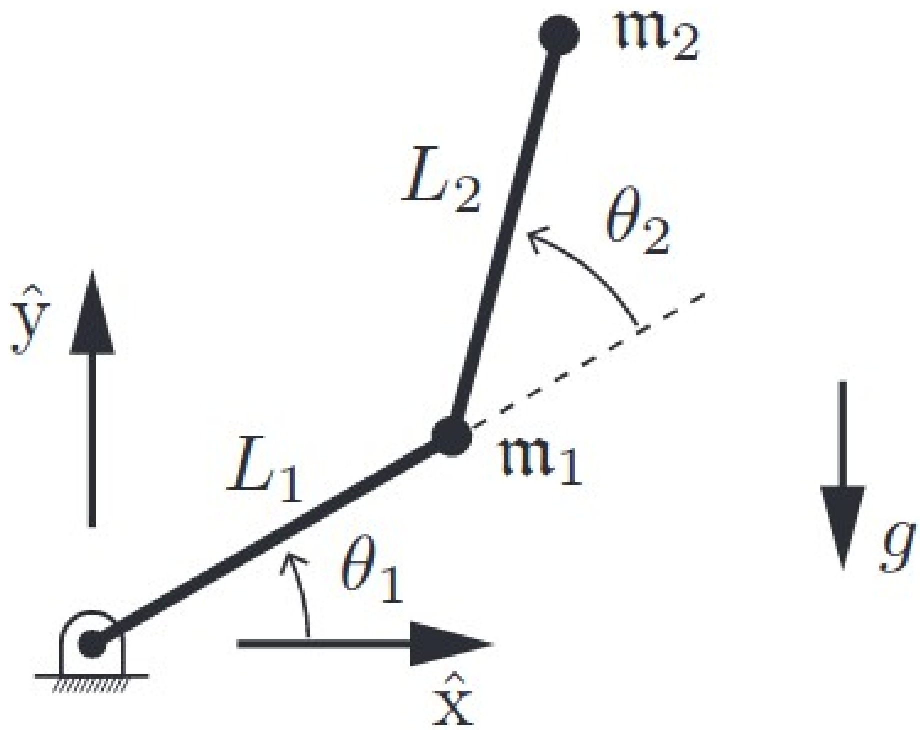
- Represent response to motor/joint torques
- Equation of motion

torques → $$\boldsymbol{\tau} = M(\theta)\ddot{\boldsymbol{\theta}} + h(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$$ ← gravity, friction, centripetal, Coriolis

↑ mass matrix

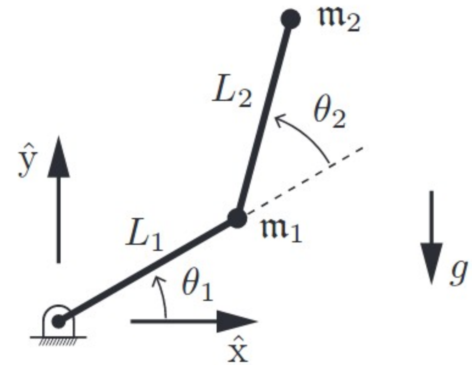
$$\boldsymbol{F} = m\,\boldsymbol{a} + m\,\boldsymbol{g} = m\,\ddot{\mathbf{x}} + m\,\boldsymbol{g}$$

Example: 2R robot under gravity



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

Example: 2R robot



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

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

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

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

What do the terms mean?

Forward dynamics with contact force

What's this? And why?

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

Cartesian contact force

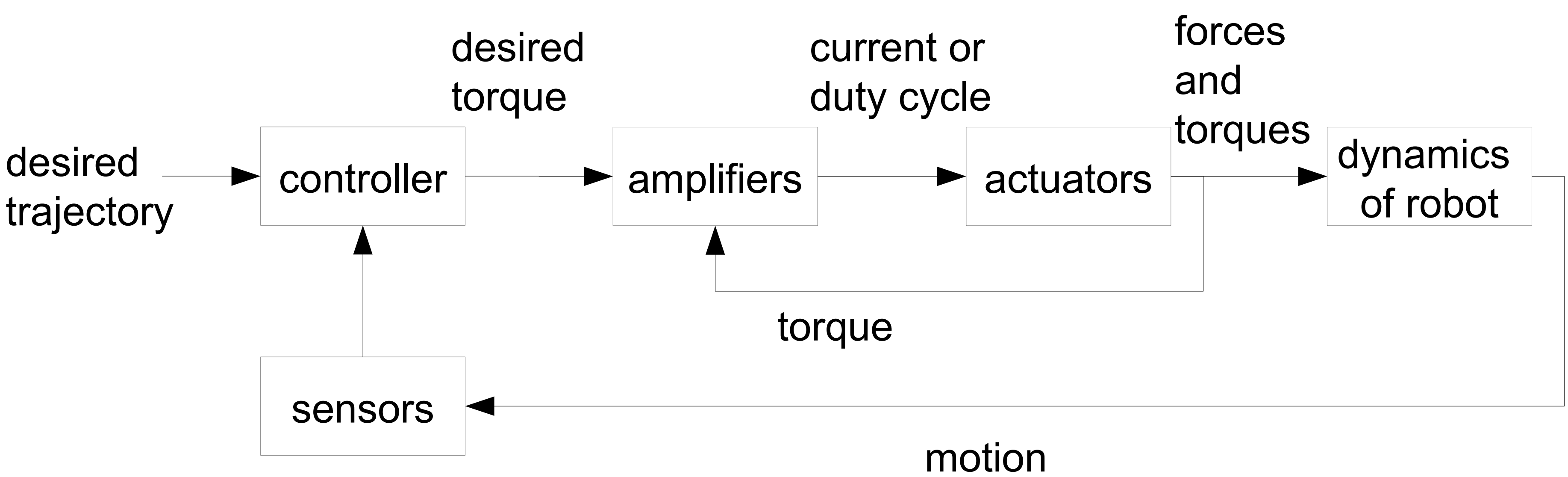
Solve:

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

linear system of equations

Torque control

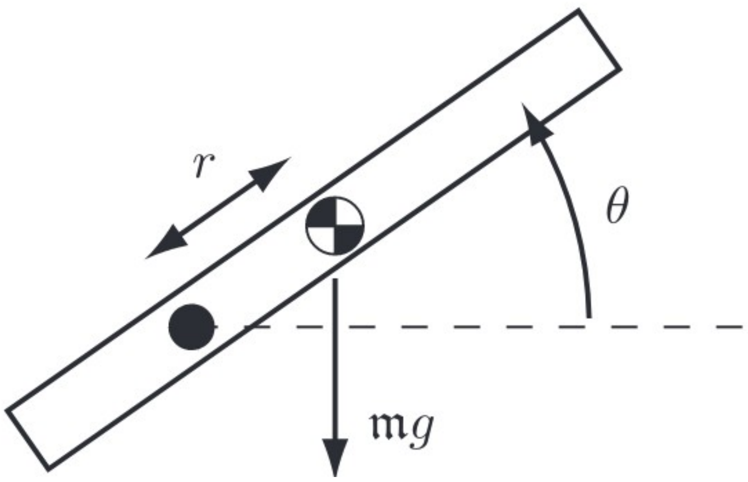
Can this model control force interactions?



Single joint torque control

- Dynamics with (simple) friction

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

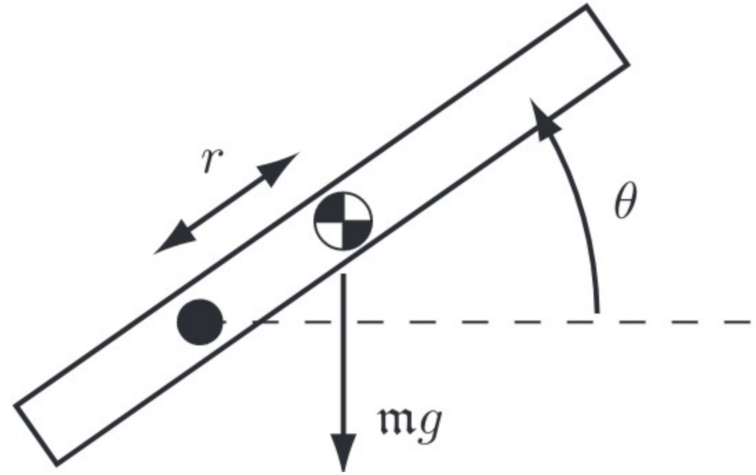


Assume you have a trajectory to follow.
Propose a controller. Or several.

Single joint torque control

- Dynamics with (simple) friction

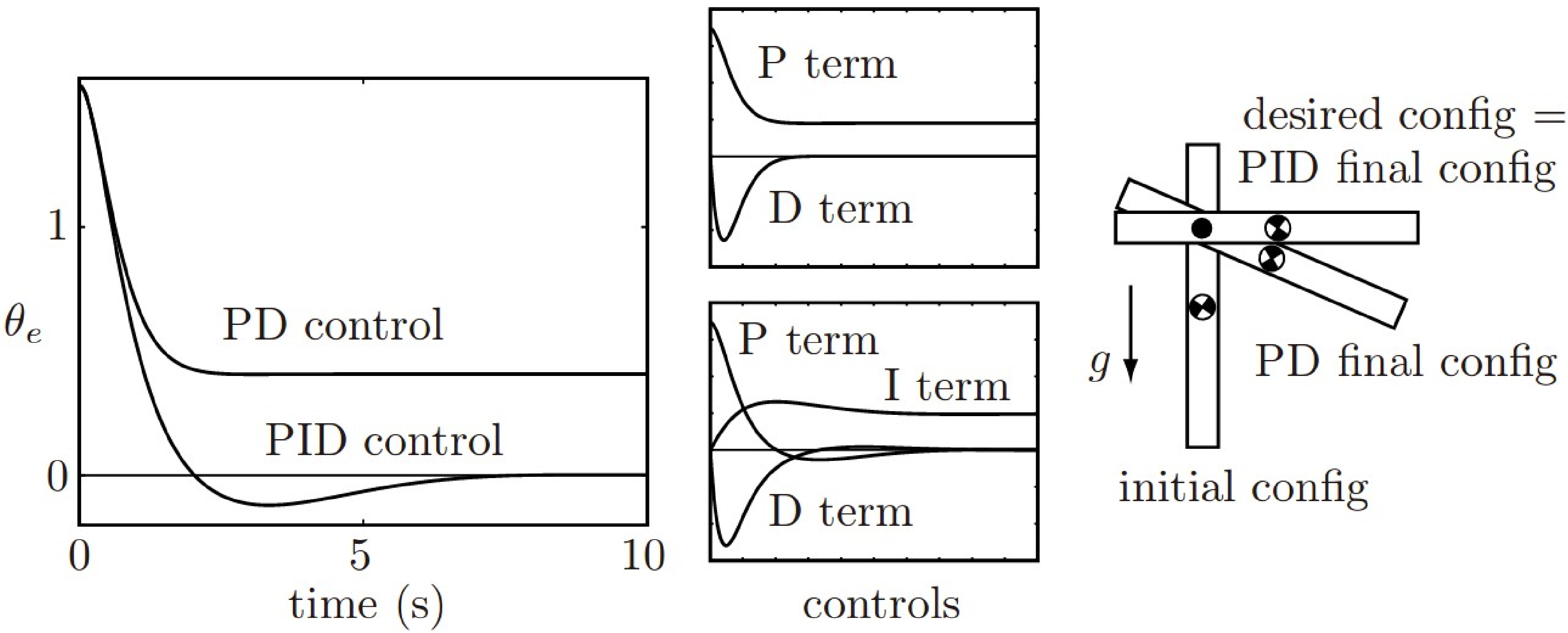
$$\tau = M\ddot{\theta} + mgr\cos\theta + b\dot{\theta}$$



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

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

PID convergence



Would PID eliminate steady state error if desired rotational velocity is constant?

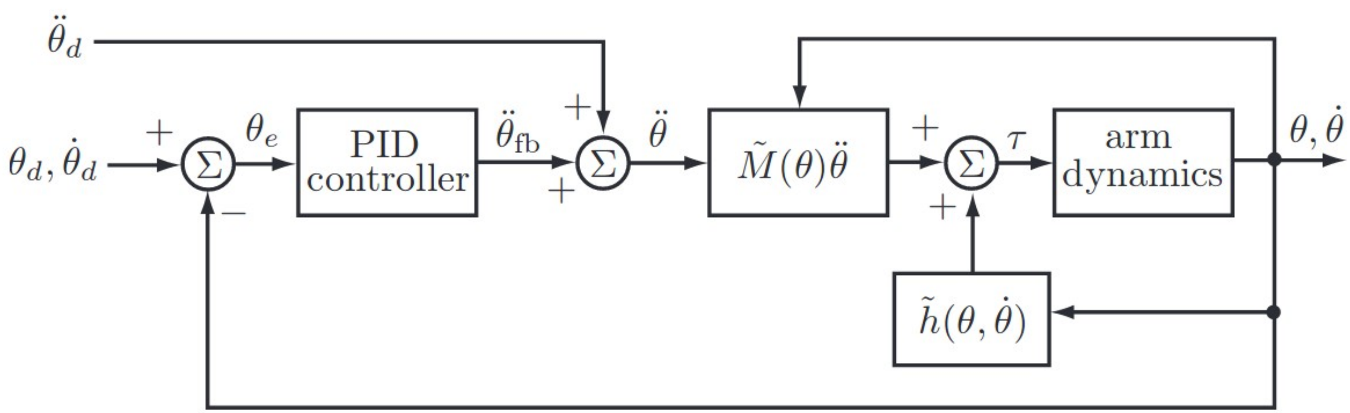
Inverse dynamics / computed torque control

dynamics compensation

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

feedforward

PID feedback



Inverse dynamics

- Problem: Calculate right hand side of

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

- Informally: If I want to follow a certain trajectory, how high torques do I need to apply at joints.

- Solution: Calculate M and **h** by Newton-Euler algorithm.

Cartesian space dynamics

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

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

Cartesian force

Cartesian acceleration

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

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

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

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

Compare to joint space!

Cartesian control

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

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

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

Compare to

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

Represents forces in Cartesian space.
May be useful when looking at force control.

Summary

- Accurate motion control requires knowledge (model) of robot dynamics.

- Good recipe: inverse dynamics + PID + feedforward (computed torque control).

Next time: Towards motion skills (guest lecture)