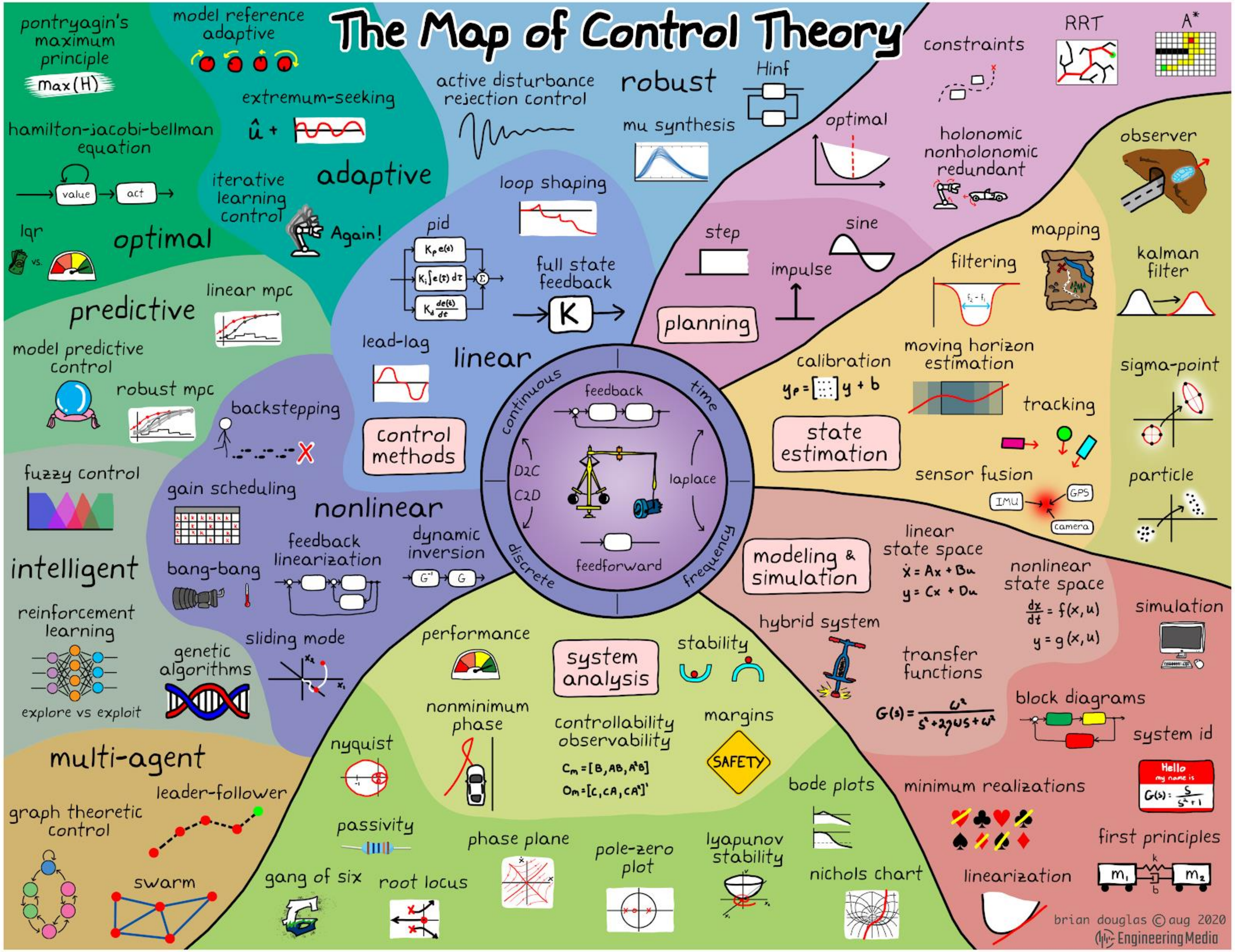
Force & Impedance Control

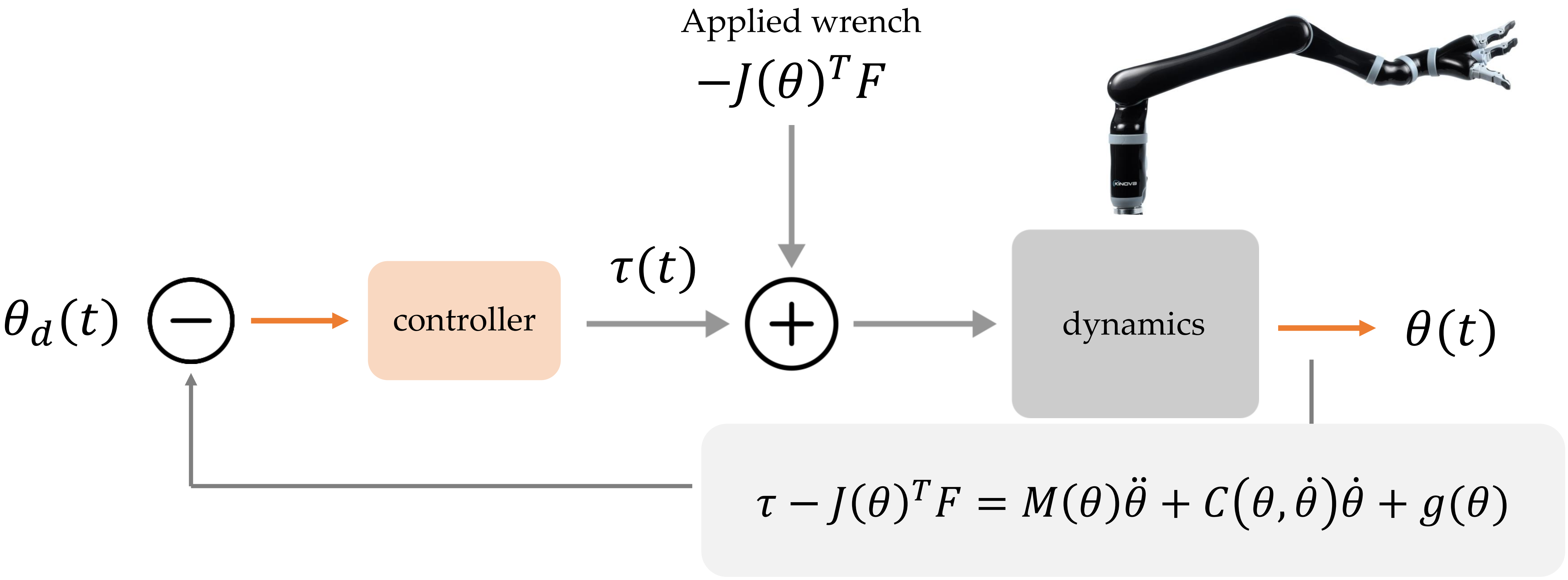
Reading: Modern Robotics 11.5, 11.7


The Map of Control Theory
pontryagin's maximum principle
max(H)
model reference adaptive
extremum-seeking
hamilton-jacobi-bellman equation
value
act
iterative learning control
Again!
adaptive
lqr
vs.
optimal
active disturbance rejection control
robust
Hinf
mu synthesis
loop shaping
pid
full state feedback
K
predictive
linear mpc
model predictive control
robust mpc
lead-lag
linear
control methods
backstepping
fuzzy control
gain scheduling
nonlinear
feedback linearization
dynamic inversion
intelligent
bang-bang
reinforcement learning
explore vs exploit
genetic algorithms
sliding mode
multi-agent
leader-follower
graph theoretic control
swarm
continuous
time
feedback
D2C
C2D
laplace
feedforward
discrete
frequency
constraints
RRT
A*
optimal
holonomic
nonholonomic
redundant
step
sine
impulse
planning
observer
mapping
filtering
kalman filter
calibration
moving horizon estimation
sigma-point
state estimation
tracking
sensor fusion
IMU
GPS
camera
particle
modeling & simulation
linear state space
nonlinear state space
hybrid system
transfer functions
simulation
block diagrams
system id
Hello my name is
minimum realizations
first principles
linearization
performance
system analysis
stability
nonminimum phase
controllability observability
margins
SAFETY
nyquist
passivity
bode plots
phase plane
pole-zero plot
lyapunov stability
nichols chart
gang of six
root locus
brian douglas © aug 2020
EngineeringMedia

This Lecture

- What is force control?
- What is impedance control?
- How do we track a desired trajectory and regulate interaction forces?

Review



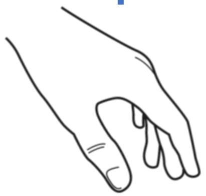
Applied wrench
$-J(\theta)^T F$
$\theta_d(t)$
controller
$\tau(t)$
dynamics
$\theta(t)$
$\tau - J(\theta)^T F = M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + g(\theta)$

Review

The overall equation of motion is:

$$\boxed{\tau} - J(\theta)^T F = \underbrace{M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + g(\theta)}_{\text{Dynamics of the robot arm}}$$

τ is the output of
our chosen controller



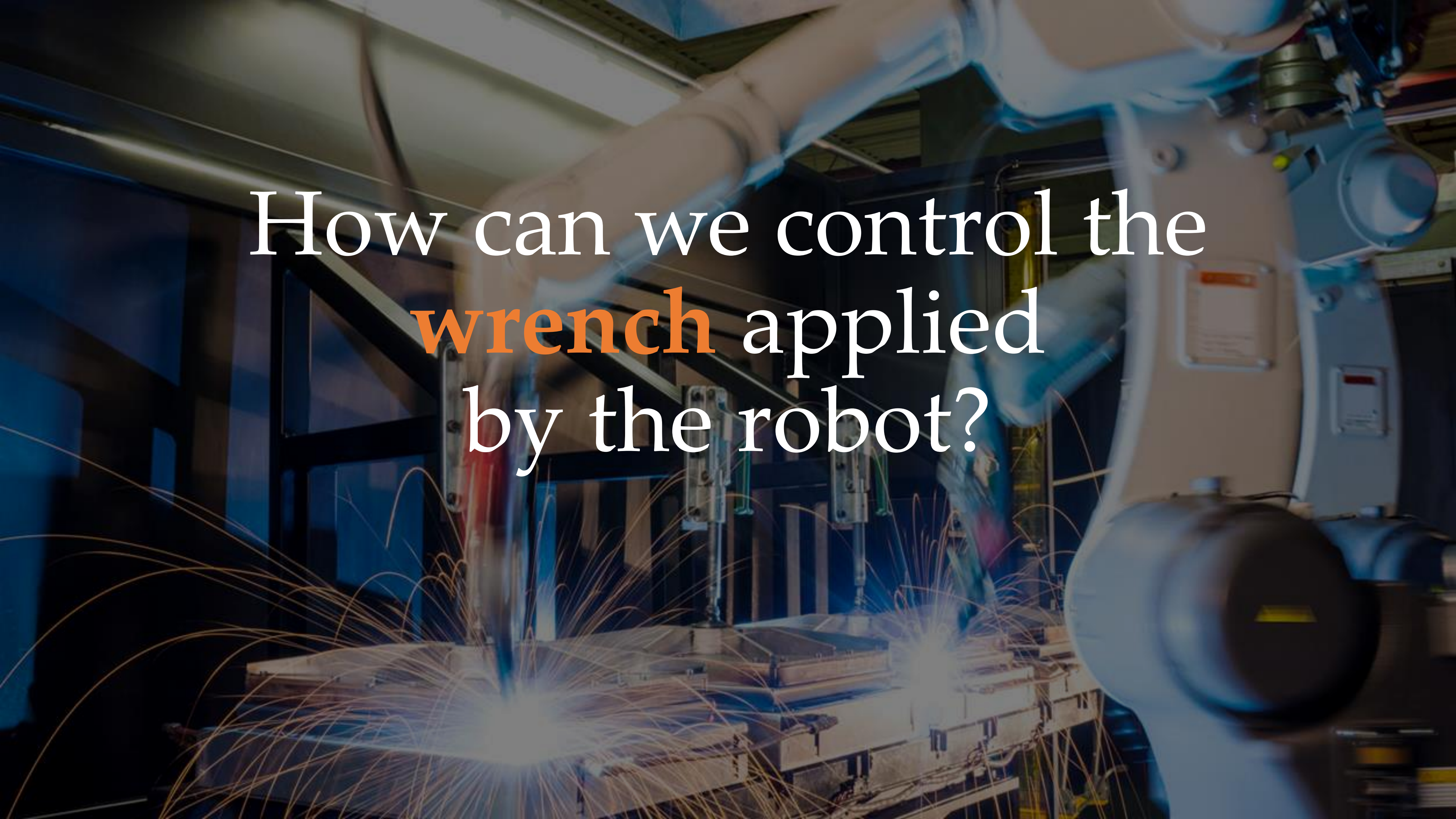
$-F$ is the external wrench
applied by the environment

Review

The overall equation of motion is:

$$\tau - J(\theta)^T F = M(\theta)\ddot{\theta} + C\big(\theta, \dot{\theta}\big)\dot{\theta} + g(\theta)$$

To reach a desired position (or follow desired trajectory),
we can use $\boldsymbol{\tau} = \boldsymbol{K_P}(\boldsymbol{\theta_d} - \boldsymbol{\theta}) - \boldsymbol{K_D}\dot{\boldsymbol{\theta}} + \boldsymbol{g}(\boldsymbol{\theta})$


How can we control the
wrench applied
by the robot?


Speed 5X
gerenga

Force Control

- **Force control** is used when we want to apply forces and torques to the environment
- Intended for settings where the end-effector does is **stationary**

$$\tau - J(\theta)^T F = M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + g(\theta)$$

Since stationary, $\ddot{\theta} = \dot{\theta} = 0$

Force Control

- **Force control** is used when we want to apply forces and torques to the environment
- Intended for settings where the end-effector does is **stationary**

$$\tau = \underbrace{g(\theta)}_{\text{Gravity compensation}} + J(\theta)^T F_d$$

$+F_d$ is the wrench we want to apply to the environment

Force Control

- **Force control** is used when we want to apply forces and torques to the environment
- Intended for settings where the end-effector does is **stationary**

$$\tau = g(\theta) + J(\theta)^T \left(F_d + K_P (F_d - F) + K_I \int (F_d - F)\, dt \right)$$

Force Control

- **Force control** is used when we want to apply forces and torques to the environment
- Intended for settings where the end-effector does is **stationary**

PI force controller

$$\tau = g(\theta) + J(\theta)^T \left(F_d + \boxed{K_P}(F_d - F) + \boxed{K_I} \int (F_d - F)\, dt \right)$$

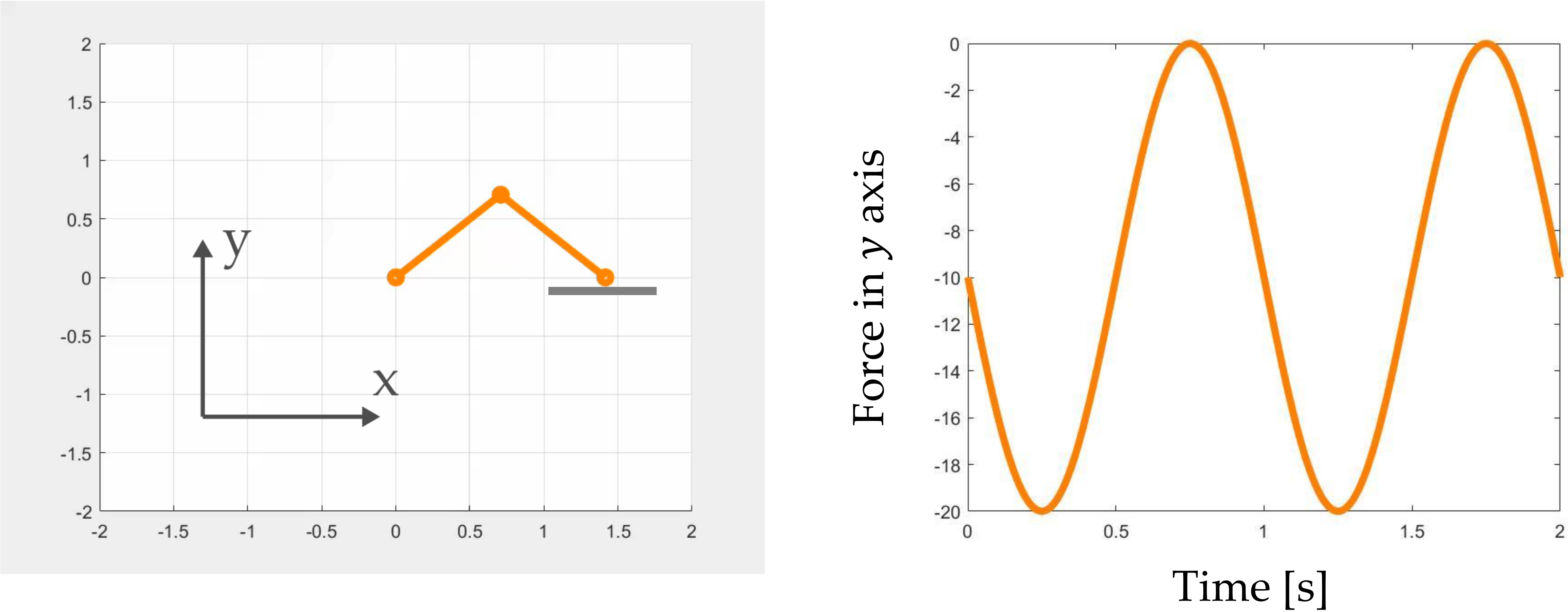
F is the actual wrench measured by the force-torque sensor

Force Control

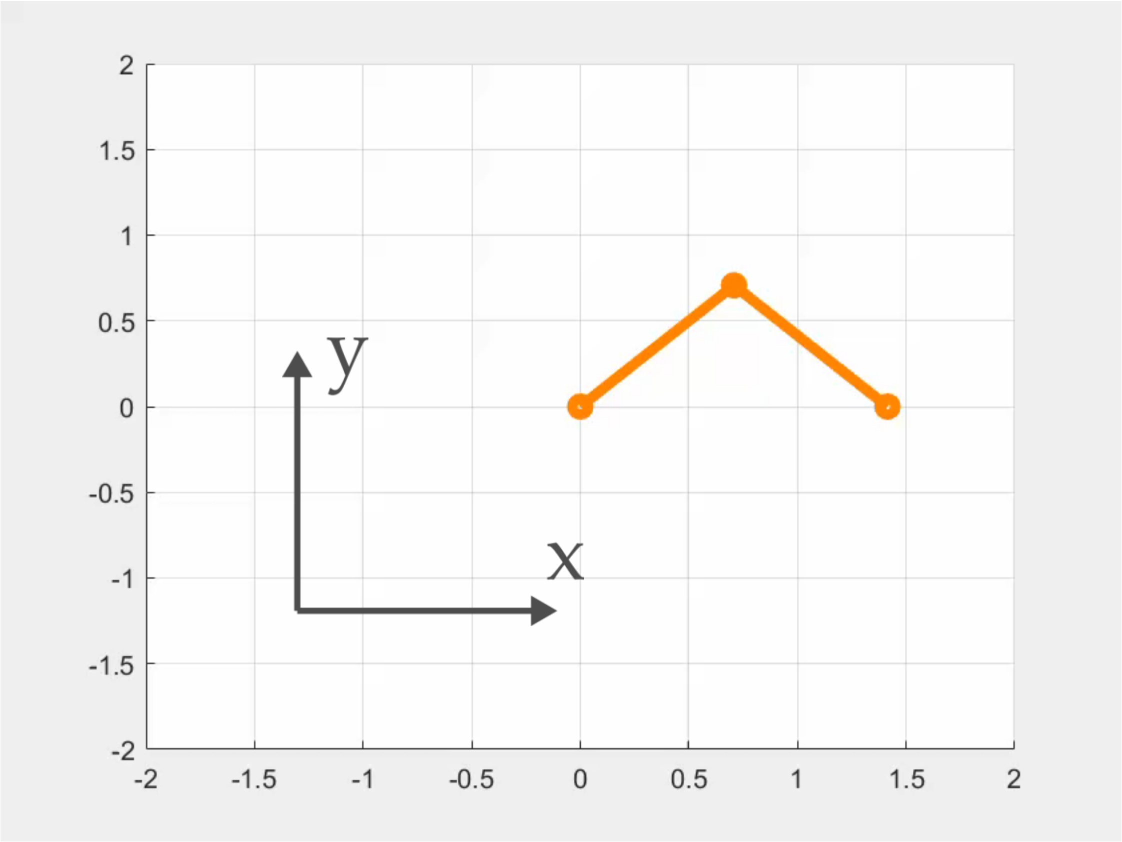
$$\tau = g(\theta) + J(\theta)^T \left(F_d + K_P(F_d - F) + K_I \int (F_d - F)\, dt \right)$$

If nothing to push against, robot accelerates

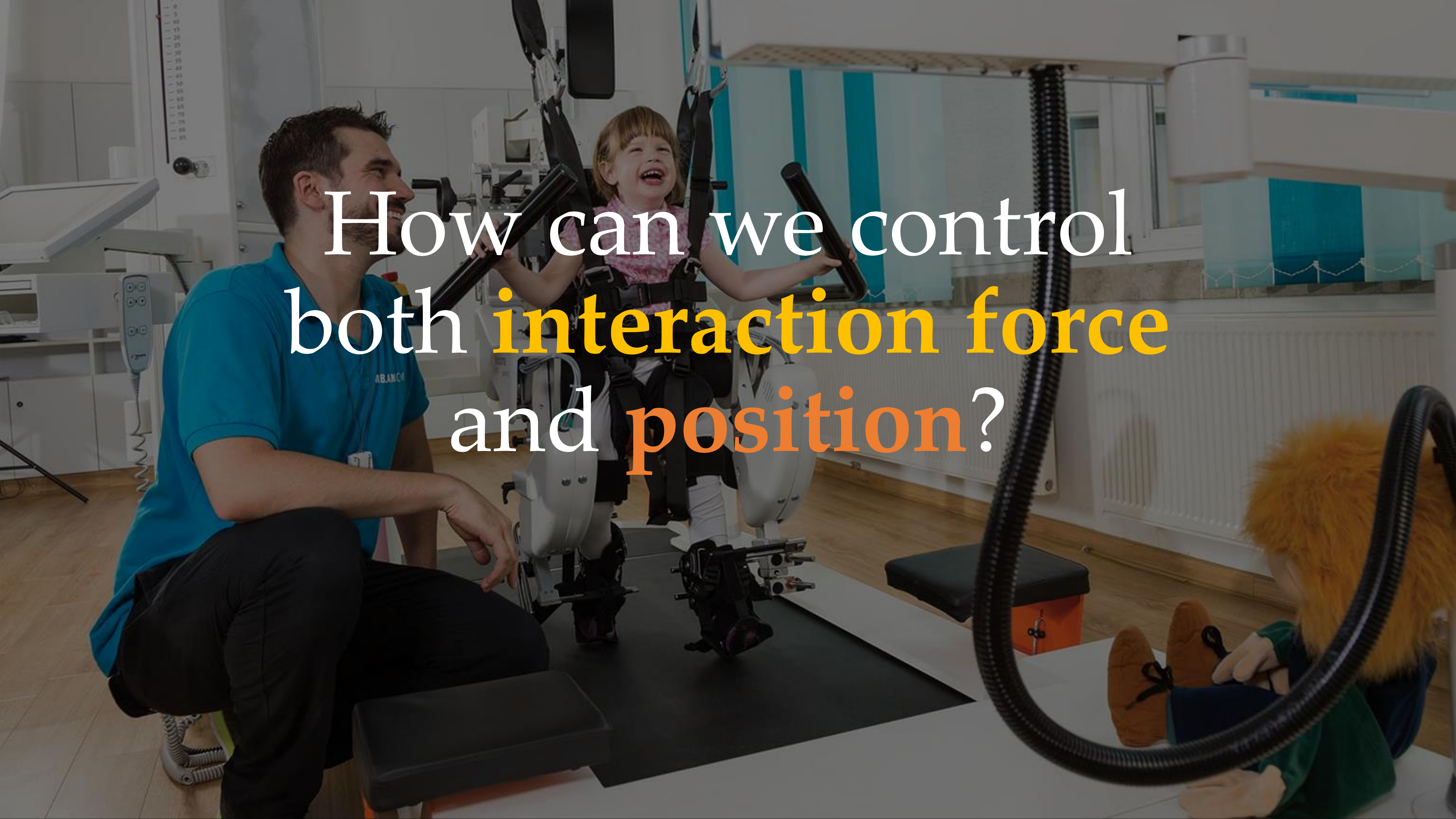
Hint: Include a speed threshold or stop conditions



y
x
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
Force in y axis
Time [s]
0
-2
-4
-6
-8
-10
-12
-14
-16
-18
-20
0
0.5
1
1.5
2



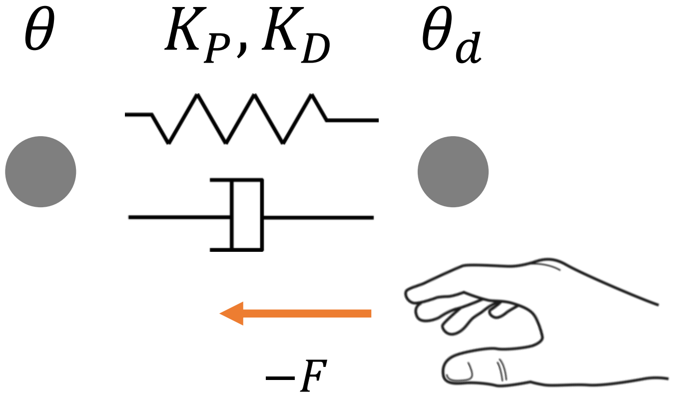
Nothing to resist applied force, robot falls.



How can we control both **interaction force** and **position**?

Impedance Control

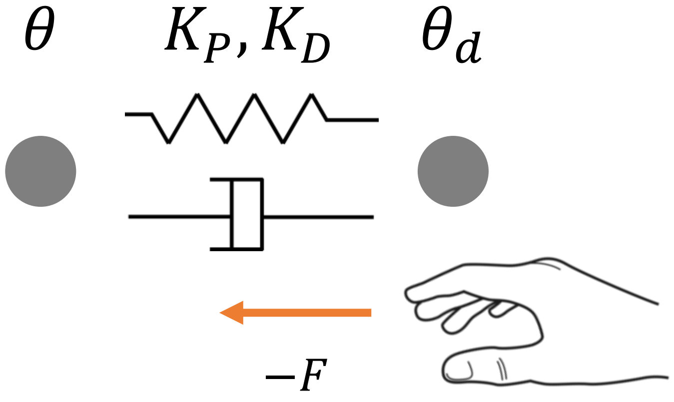
Impedance control makes the robot behave like a **mass-spring-damper** with stiffness, damping, and equilibrium of our choice.



Impedance Control

Impedance control makes the robot behave like a **mass-spring-damper** with stiffness, damping, and equilibrium of our choice.

We choose stiffness matrix K_P, damping matrix K_D, and desired trajectory $\theta_d(t)$, $\dot{\theta}_d(t)$



Impedance Control

$$\tau - J(\theta)^T F = M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + g(\theta)$$

Overall equation of motion

$$\tau = \underbrace{K_D(\dot{\theta}_d - \dot{\theta}) + K_P(\theta_d - \theta)}_{\text{Set the damping and stiffness}} + \underbrace{M(\theta)\ddot{\theta}_d + C(\theta, \dot{\theta})\dot{\theta} + g(\theta)}_{\text{Feedforward dynamics cancellation}}$$

Impedance Control

$$M(\theta)\ddot{\theta}_d - M(\theta)\ddot{\theta} + K_D\left(\dot{\theta}_d - \dot{\theta}\right) + K_P(\theta_d - \theta) = J(\theta)^T F$$

We do not cancel out the mass matrix because measuring acceleration is challenging

$$M(\theta)\ddot{e} + K_D\dot{e} + K_P e = J(\theta)^T F$$

Closed-loop dynamics with
$e = \theta_d - \theta,\ \dot{e} = \dot{\theta}_d - \dot{\theta},\ \ddot{e} = \ddot{\theta}_d - \ddot{\theta}$

Impedance Control

$$M(\theta)\ddot{e} + K_D\dot{e} + K_P e = J(\theta)^T F$$

Increasing K_D and K_P **increases the impedance**:
the human must apply large external forces to move the robot

Decreasing K_D and K_P **decreases the impedance**:
the human can backdrive or guide the robot with small forces

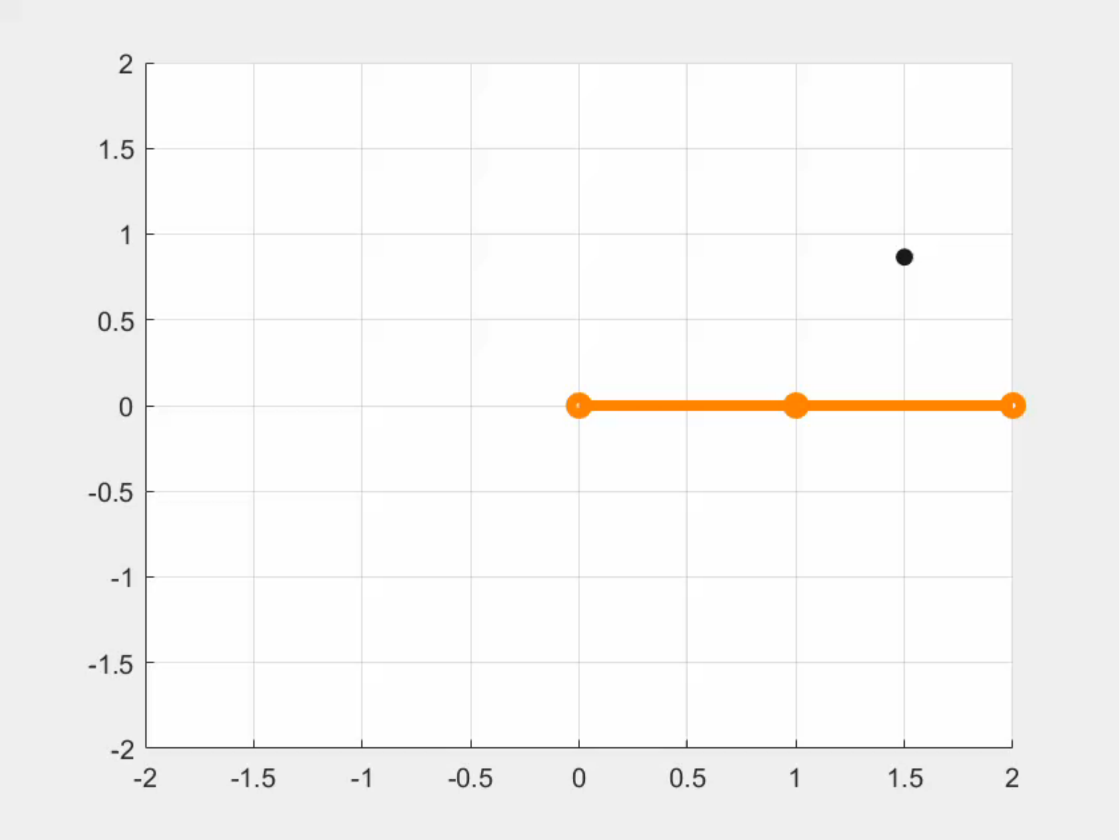
Impedance Control

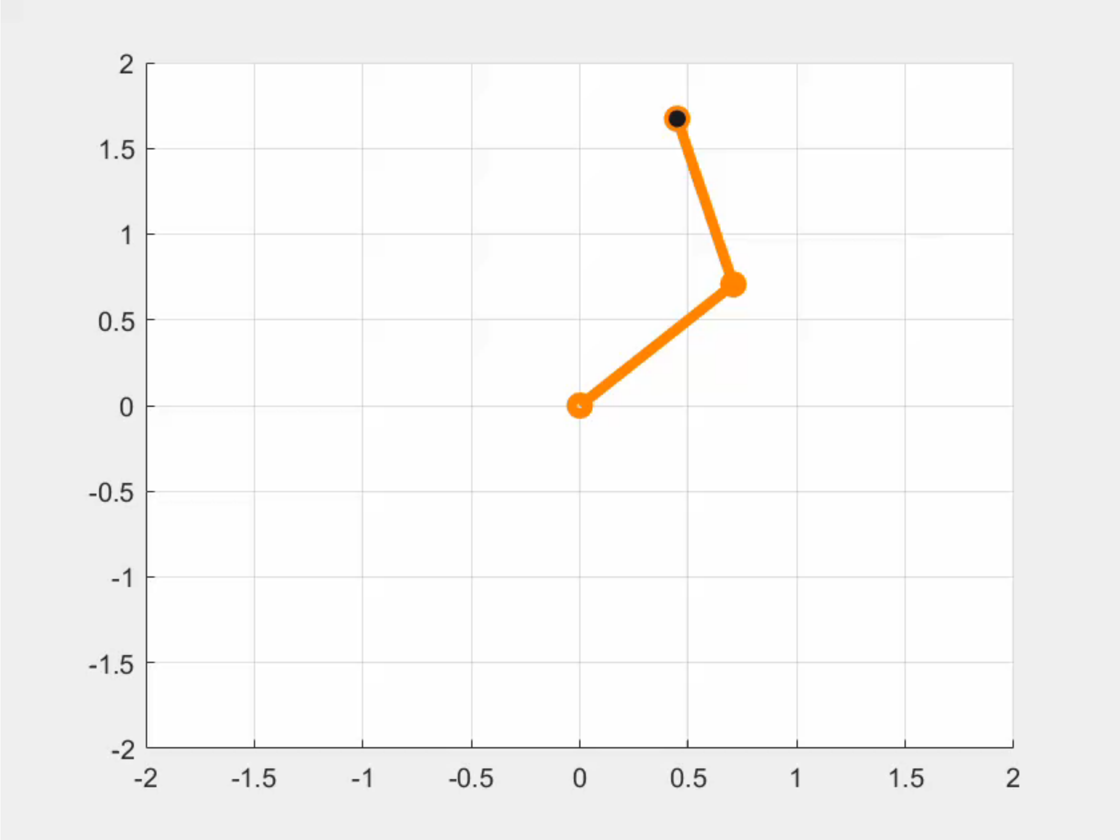
In Practice. Need accurate estimate of M, C, and g to cancel dynamics

$$\tau = K_D(\dot{\theta}_d - \dot{\theta}) + K_P(\theta_d - \theta) + \underline{\widehat{M}(\theta)\ddot{\theta}_d + \hat{C}(\theta, \dot{\theta})\dot{\theta} + \hat{g}(\theta)}$$

We never have exact models.
Use estimates $\widehat{M}$, $\hat{C}$, and $\hat{g}$

Methods like **robust** and **adaptive** control (*not covered here*) address model errors


2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2 -1.5 -1 -0.5 0 0.5 1 1.5 2


2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2 -1.5 -1 -0.5 0 0.5 1 1.5 2

This Lecture

- What is force control?
- What is impedance control?
- How do we track a desired trajectory and regulate interaction forces?

Next Lecture

- How do we get the robot's desired trajectory in the first place?
- Starting motion planning