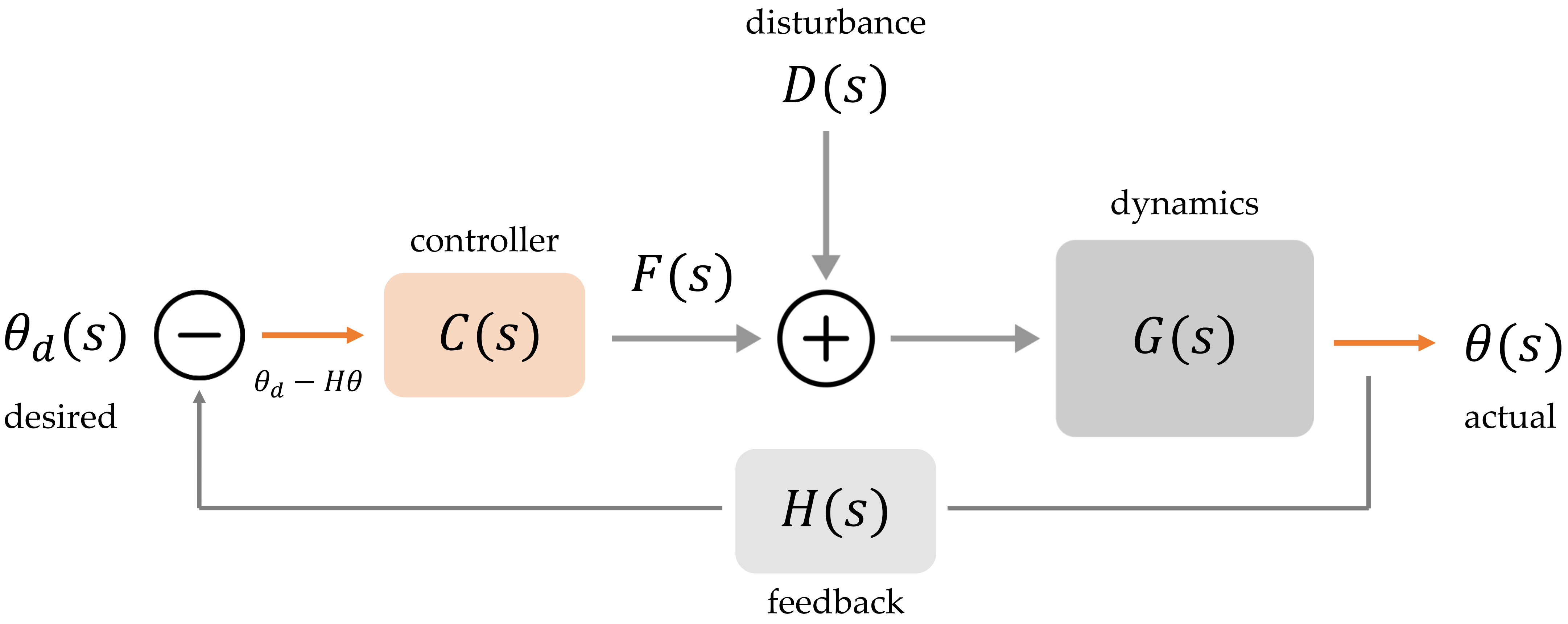
Independent Joint Control

Reading: Robot Modeling and Control 6.2, 6.3, 6.4

This Lecture

- How do we choose a controller?
- How do we know if a controller is stable?
- How do we extend this to multi-DoF robot arms?

Closed-Loop Control



Closed-Loop Control

From the block diagram we obtain the input-output relationship:

$$\theta(s) = \frac{C(s)G(s)}{1 + C(s)H(s)G(s)}\theta_d(s) + \frac{G(s)}{1 + C(s)H(s)G(s)}D(s)$$

$$\frac{\theta(s)}{\theta_d(s)}$$

Closed-loop transfer function

Closed-Loop Control

When controlling an individual robot joint, common choices are:

- (**P**) Proportional control. $C(s) = k_p$
- (**PD**) Proportional-derivative control. $C(s) = k_d s + k_p$
- (**PI**) Proportional-integral control. $C(s) = \frac{k_i}{s} + k_p$
- (**PID**) Proportional-integral-derivative control. $C(s) = k_d s + k_p + \frac{k_i}{s}$

How do we choose the right **controller** C(s)?

Stability

- A linear system is **stable** if *all poles of characteristic equation have negative real values*
- Characteristic equation: set denominator of closed-loop transfer function equal to zero

$$1 + C(s)H(s)G(s) = 0$$

Characteristic equation for the **example** block diagram.

Stability

- A linear system is **stable** if *all poles of characteristic equation have negative real values*
- Characteristic equation: set denominator of closed-loop transfer function equal to zero

$$1 + C(s)H(s)G(s) = 0$$

Imagine the characteristic equation is $s(s-5) = 0$
Then the poles are $s = 0$ and $s = 5$,
and the system is not stable

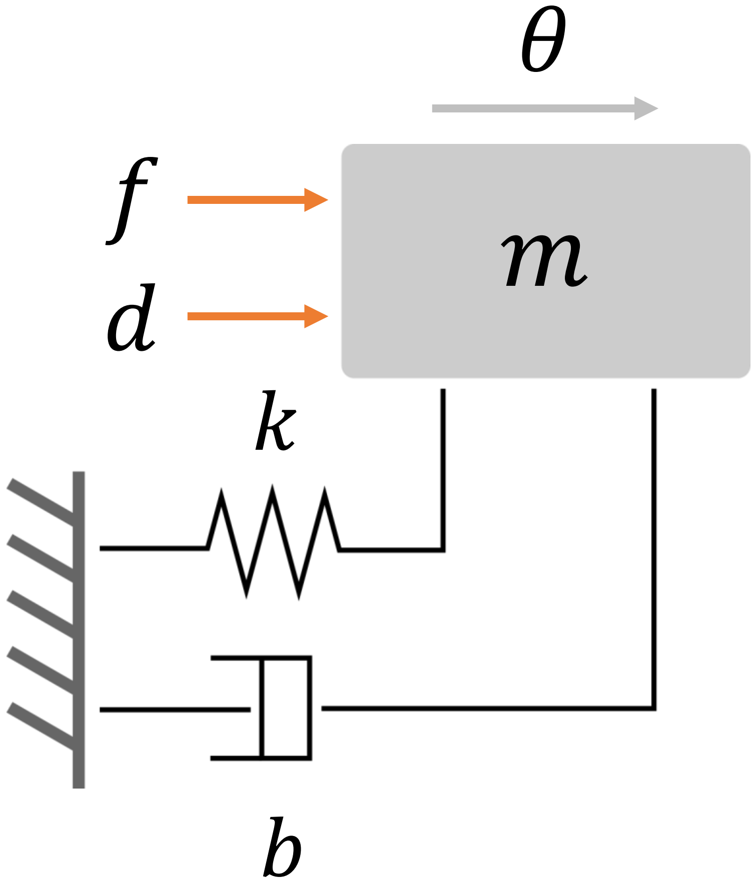
Stability

- A linear system is **stable** if *all poles of characteristic equation have negative real values*
- Characteristic equation: set denominator of closed-loop transfer function equal to zero

Use the **Routh–Hurwitz criterion** to find when polynomials have negative real poles:

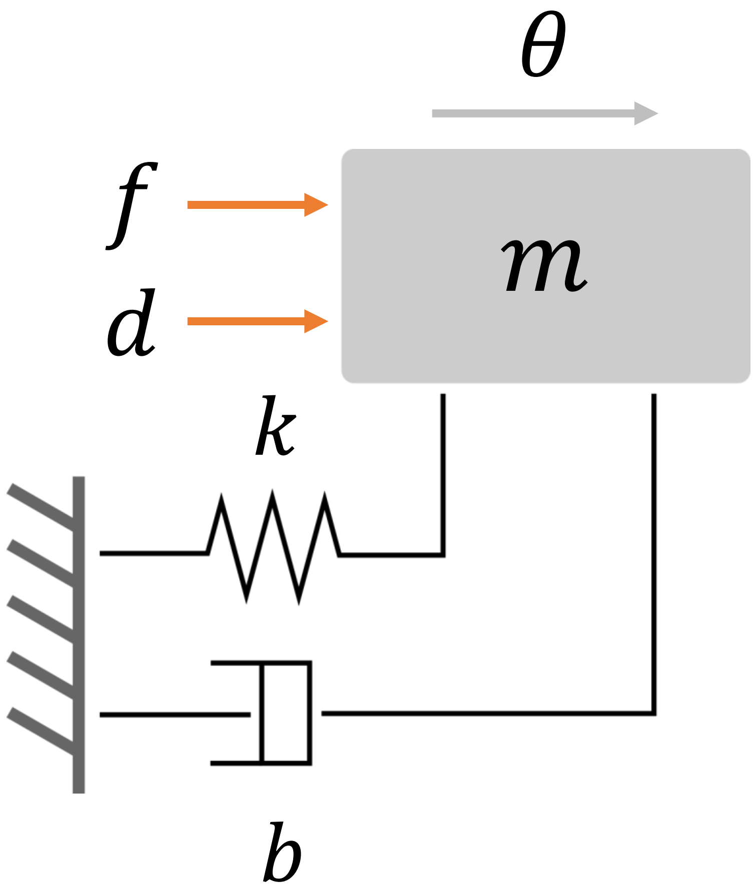
- $s^2 + a_1 s + a_0 = 0$ is stable when $a_1 > 0$ and $a_0 > 0$
- $s^3 + a_2 s^2 + a_1 s + a_0 = 0$ is stable when $a_2 > 0$, $a_1 > 0$, $a_0 > 0$ and $a_2 a_1 > a_0$

Example



Find the gains for a **PD controller** that result in stability. Assume you have a sensor that measures joint position θ.

Example



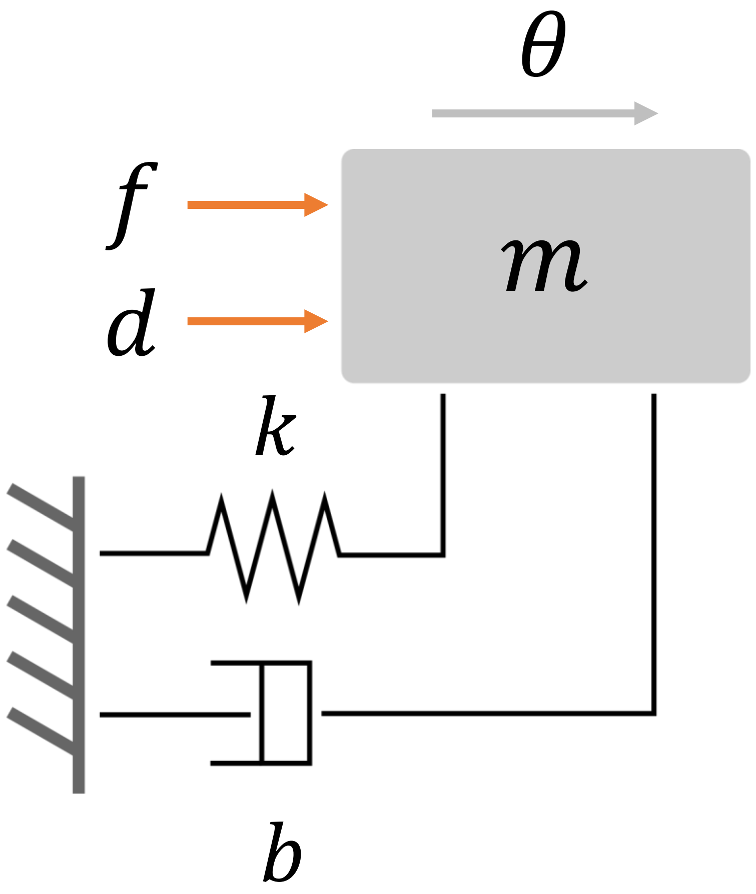
Step 1. Get dynamics $G(s)$

$$f + d = m\ddot{\theta} + b\dot{\theta} + k\theta$$

$$F(s) + D(s) = (ms^2 + bs \ + k)\theta(s)$$

$$G(s) = \frac{\theta(s)}{F(s) + G(s)} = \frac{1}{ms^2 + bs \ + k}$$

Example



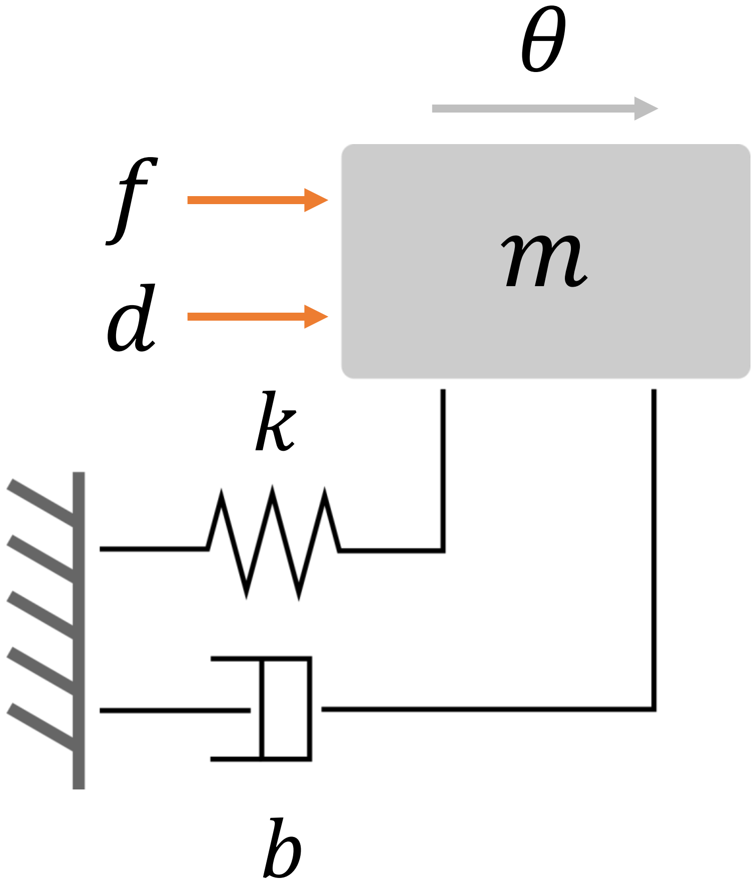
Step 2. Get closed-loop transfer function

$$\theta(s) = \frac{C(s)G(s)}{1 + C(s)H(s)G(s)}\theta_d(s)$$

We observe θ, so $H(s) = 1$

$$\theta(s) = \frac{C(s)G(s)}{1 + C(s)G(s)}\theta_d(s)$$

Example



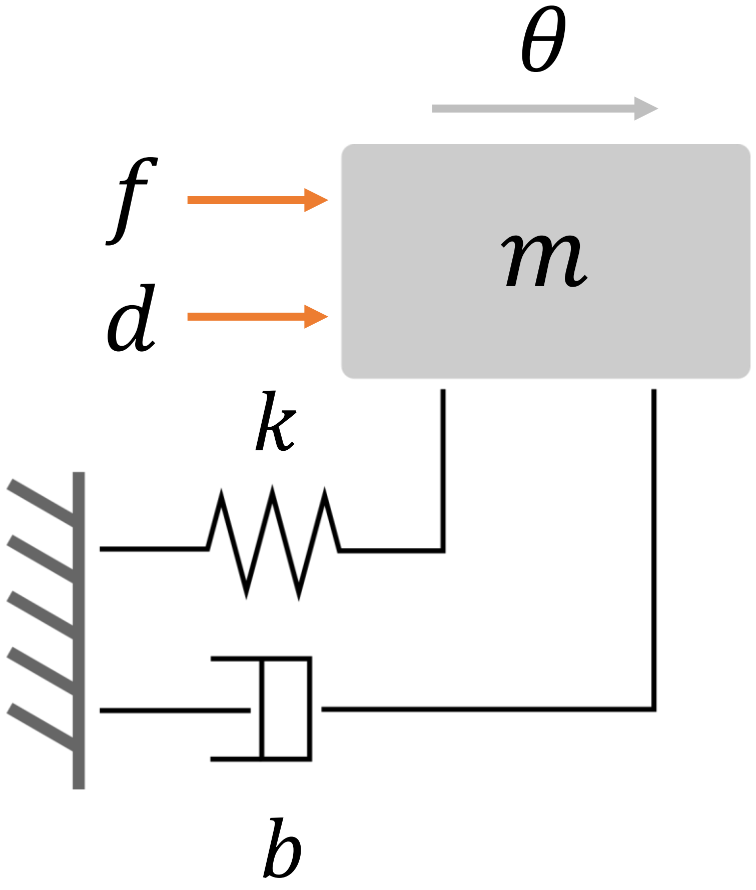
Step 3. Solve characteristic equation

$$1 + C(s)G(s) = 0$$

$$1 + \frac{k_d s + k_p}{ms^2 + bs + k} = 0$$

$$ms^2 + (b + k_d)s + \left(k + k_p\right) = 0$$

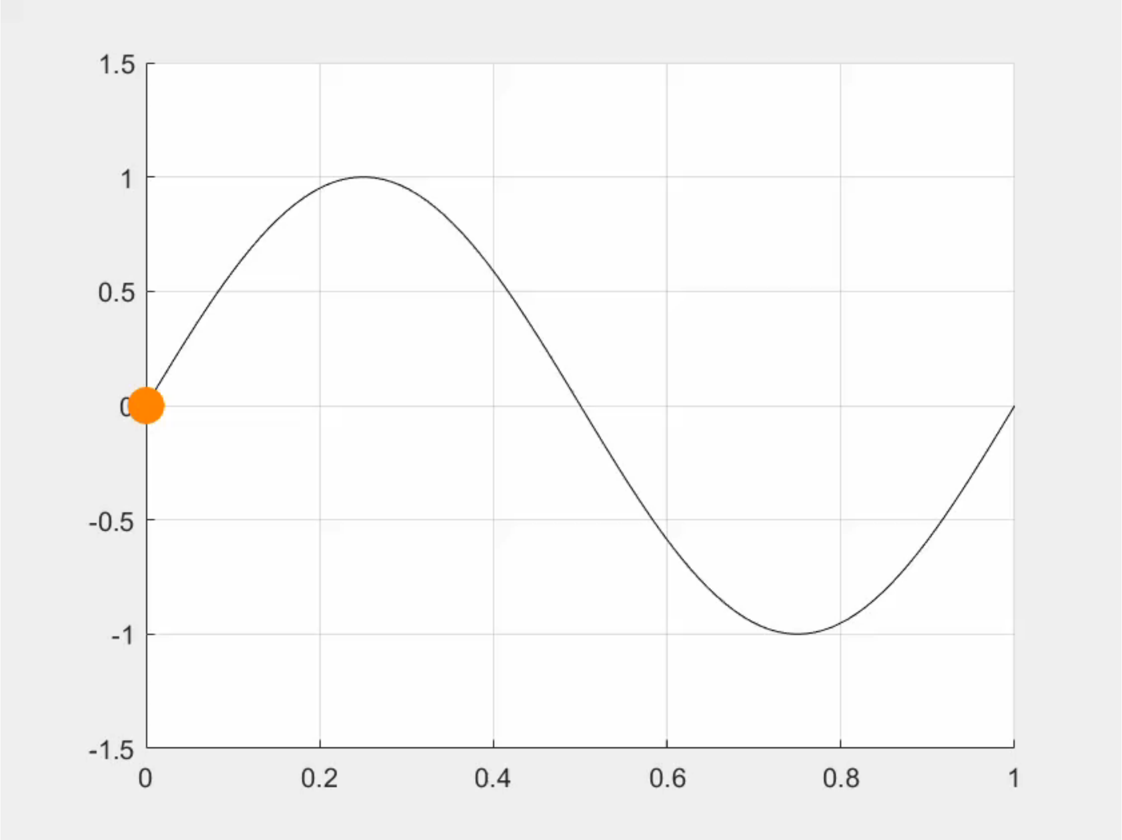
Example

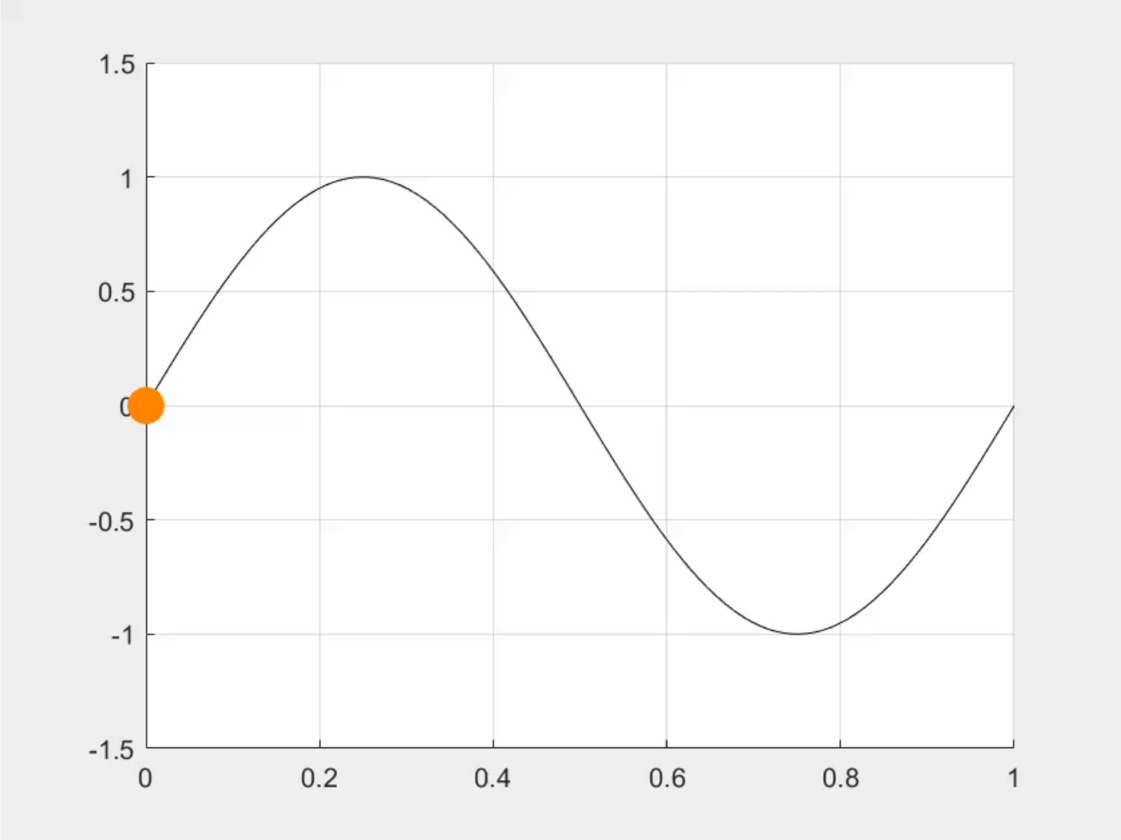


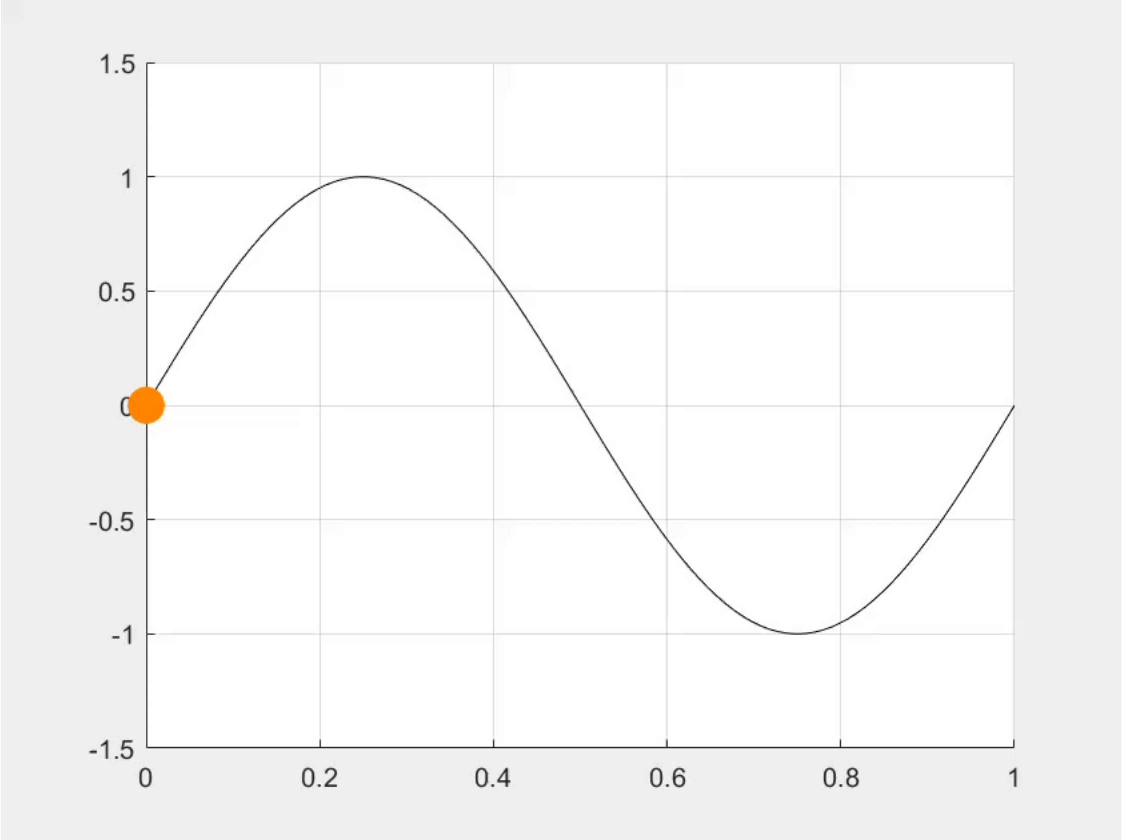
$$ms^2 + (b + k_d)s + \left(k + k_p\right) = 0$$

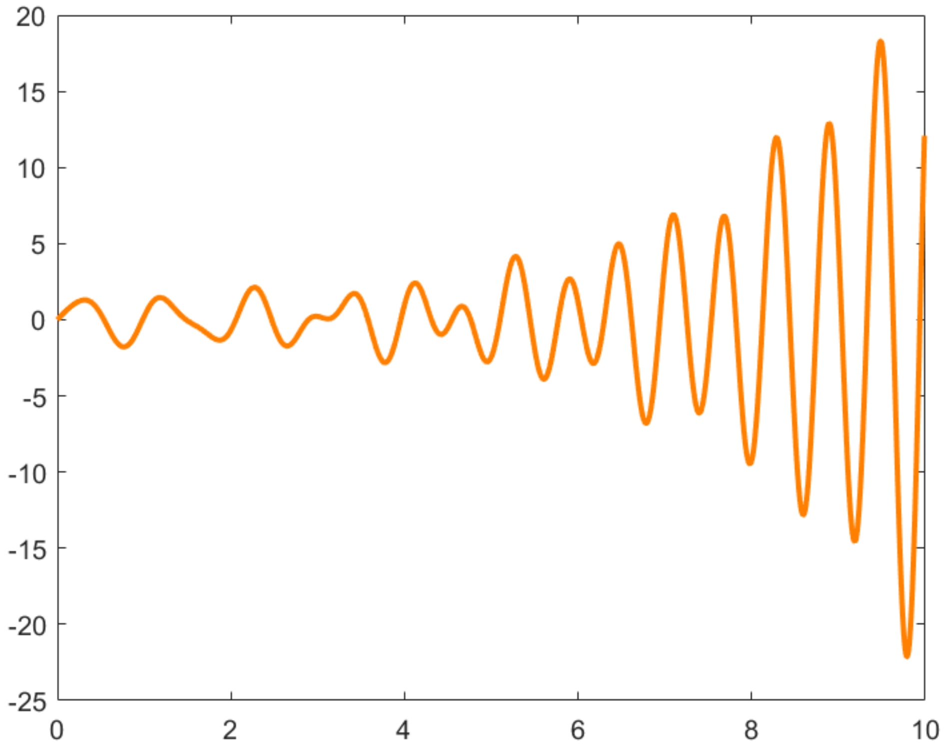
Assuming $m > 0$, the system is stable when:

- $b + k_d > 0$
- $k + k_p > 0$

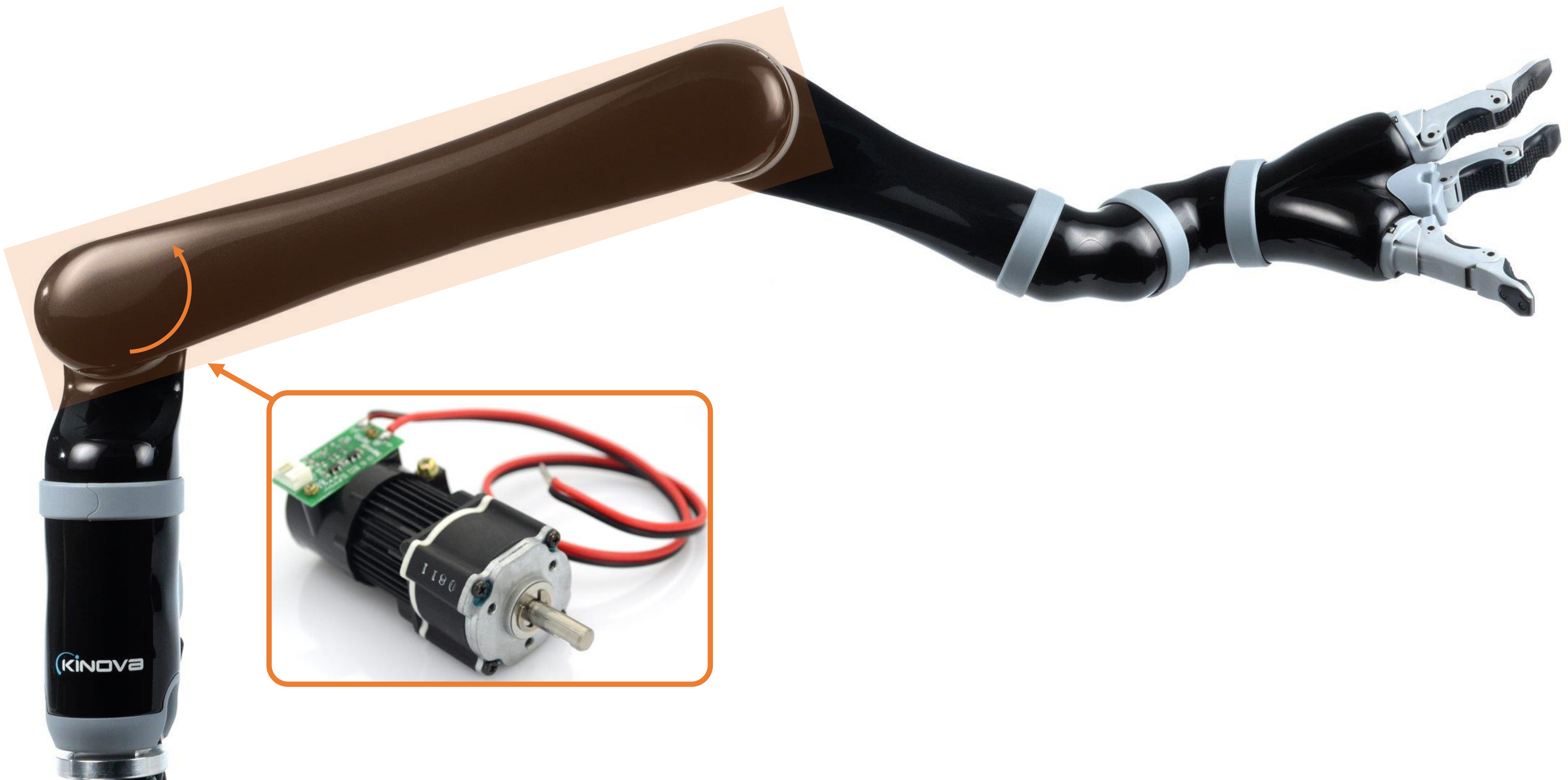

1.5
1
0.5
0
-0.5
-1
-1.5
0
0.2
0.4
0.6
0.8
1

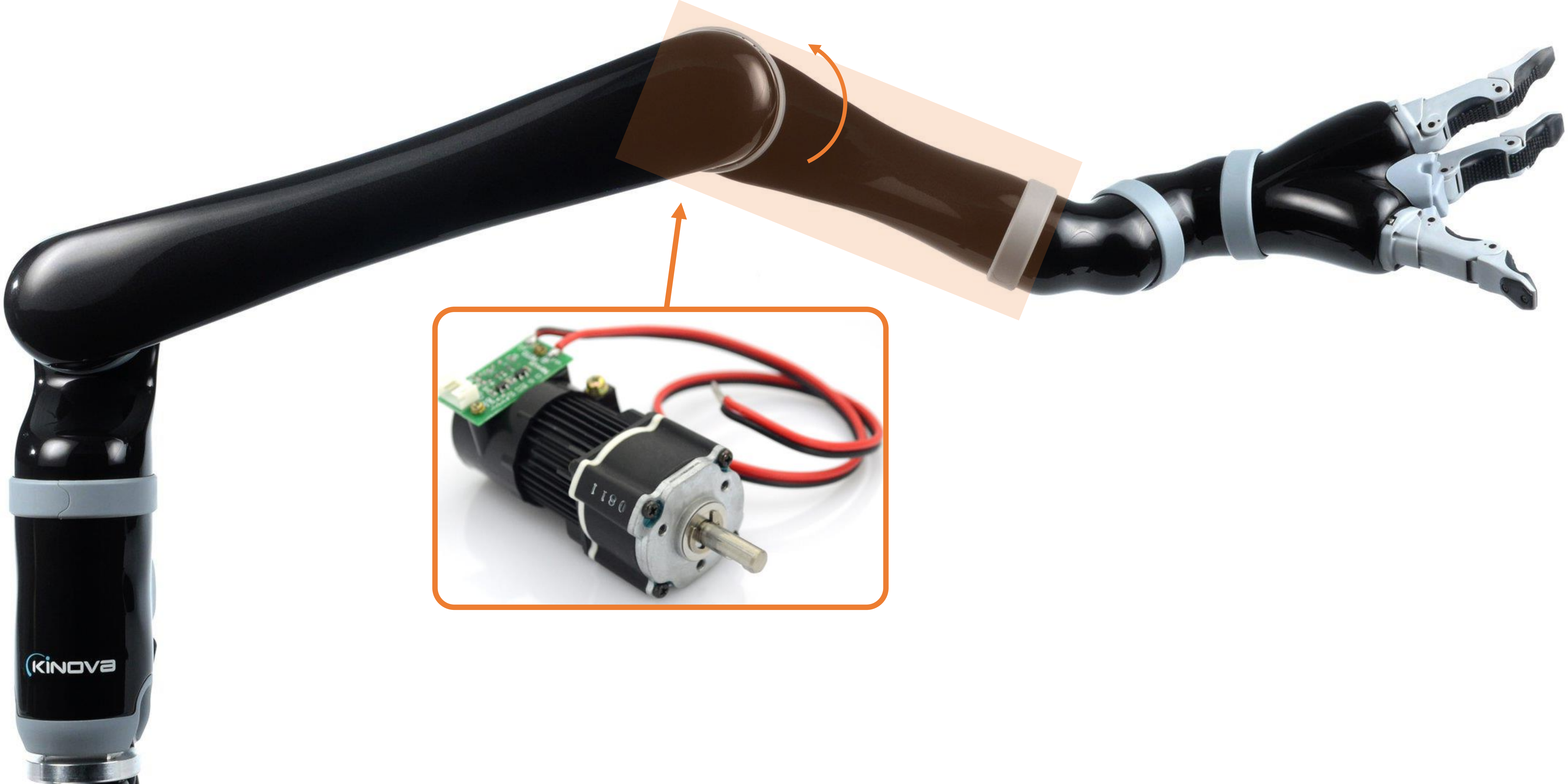

1.5
1
0.5
0
-0.5
-1
-1.5
0
0.2
0.4
0.6
0.8
1


1.5
1
0.5
0
-0.5
-1
-1.5
0
0.2
0.4
0.6
0.8
1


20
15
10
5
0
-5
-10
-15
-20
-25
0
2
4
6
8
10

From one joint to **multiple joints**


KINOVA


KINOVA


KINOVA

Independent Joint Control

In **independent joint control** each joint is treated as a separate single-input single-output system. The interaction forces between joints are **disturbances**.

Independent Joint Control

In **independent joint control** each joint is treated as a separate single-input single-output system. The interaction forces between joints are **disturbances**.

In Practice. Design separate controllers for each 1-DoF joint so that the joint is stable and tracks the desired trajectory.

$$f_1 + d_1 = m_1\ddot{\theta}_1 + b_1\dot{\theta}_1$$
$$f_2 + d_2 = m_2\ddot{\theta}_1 + k_2(\theta_2 - \theta_1)$$
$$\vdots$$

This Lecture

- How do we choose a controller?
- How do we know if a controller is stable?
- How do we extend this to multi-DoF robot arms?

Next Lecture

- Multivariable control using the robot’s dynamics