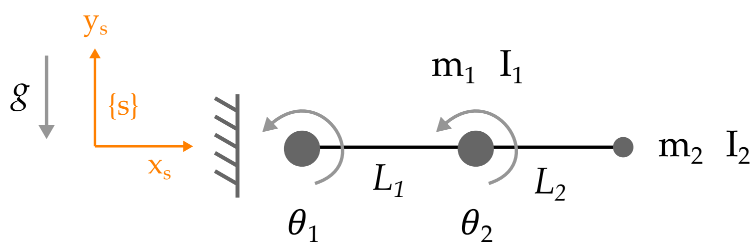
Practice Set 28

Robotics & Automation
Dylan Losey, Virginia Tech

Using your textbook and what we covered in lecture, try solving the following problems. For some problems you may find it convenient to use Matlab (or another programming language of your choice). **Download the starter code provided with this practice set.**



We have derived the dynamics for the robot shown above, and you have previously written code to simulate this robot. Now we will control the robot.

Problem 1

Implement a multivariable PD controller with gravity compensation:

$$\tau = K_P(\theta_d - \theta) - K_D\dot{\theta} + g(\theta) \tag{1}$$

Select $\theta_d = [\pi/4, \pi/4]^T$ as your desired position. Tune the gains in K_P and K_D to achieve performance you find satisfactory.

Problem 2

Implement an impedance controller:

$$\tau = K_P(\theta_d - \theta) + K_D(\dot{\theta}_d - \dot{\theta}) + \hat{M}(\theta)\ddot{\theta}_d + \hat{C}(\theta, \dot{\theta})\dot{\theta} + \hat{g}(\theta) \tag{2}$$

Remember that $\hat{}$ denotes *estimated* dynamics. Start by setting $\hat{M} = M$, $\hat{C} = C$, and $\hat{g} = g$. Track the desired trajectory where t is the simulation time:

$$\theta_d = \begin{bmatrix} t \cdot \pi/2 \\ \pi/3 \end{bmatrix}, \qquad \dot{\theta}_d = \begin{bmatrix} \pi/2 \\ 0 \end{bmatrix}, \qquad \ddot{\theta}_d = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{3}$$

Again you may need to tune the gains K_P and K_D.

Problem 3

Use the same setting as Problem 2, but now test with dynamics that have modeling errors. Set $\hat{M} \neq M$, $\hat{C} \neq C$, and $\hat{g} \neq g$. Based on your experiences in this problem, try to figure out how you should tune K_P and K_D to deal with these errors.