Objectives

- Getting familiarized with the Quanser QUBE-Servo 2 hardware.
- Using Arduino to interact with QUBE-Servo 2 system.
- Model linearization
- Model validation

1 Pre-lab: Integration

In this Pre-lab, we will make a program using Arduino-Software to drive the DC motor and then measure it's corresponding angle.

1.1 Reading the Encoder

Follow these steps to read the encoder:

1. Load the Arduino software.
2. Open the template file "QuanserQUBEServo2_Arduino.ino"
3. We propose the read the encoder using the serial port. So, it is necessary to define which program we are going to use for serial plot. We recommend SerialPlot (https://hackaday.io/project/5334-serialplot-realtime-plotting-software).
4. Based on the input format of serial plot program, add a serial print in the display section of the code.
5. Run the code on the Arduino.
6. Run the serial plot program. Rotate the disc back and forth. The program must shows the changes in the angle.
7. What happens to the encoder reading every time the Arduino is started? What do you notice about the encoder measurement when the Arduino is re-started?

1.2 Driving the DC Motor

1. Apply a constant signal of 0.5 V to the DC motor in the QUBE-Servo 2.
2. Confirm that we are obtaining a *positive measurement when a positive signal is applied*. This convention is important, especially in control systems when the design assumes the measurement goes up positively when a positive input is applied. Finally, in what direction does the disc rotate (clockwise or counter-clockwise) when a positive input is applied?

[1]This laboratory take some material from the Quanser Inc workbooks. And this laboratory represent the 2% of the student's final grade.

2 Lab: State Space Modeling

The rotary pendulum model is shown in Figure 1. The rotary arm pivot is attached to the QUBE-Servo 2 system and is actuated. The arm has a length of L_r, a moment of inertia of J_r, and its angle θ increases positively when it rotates counter-clockwise (CCW). The servo (and thus the arm) should turn in the CCW direction when the control voltage is positive ($V_m > 0$).

The pendulum link is connected to the end of the rotary arm. It has a total length of L_p and it center of mass is at $L_p/2$. The moment of inertia about its center of mass is J_p. The inverted pendulum angle α is zero when it is hanging downward and increases positively when rotated CCW.

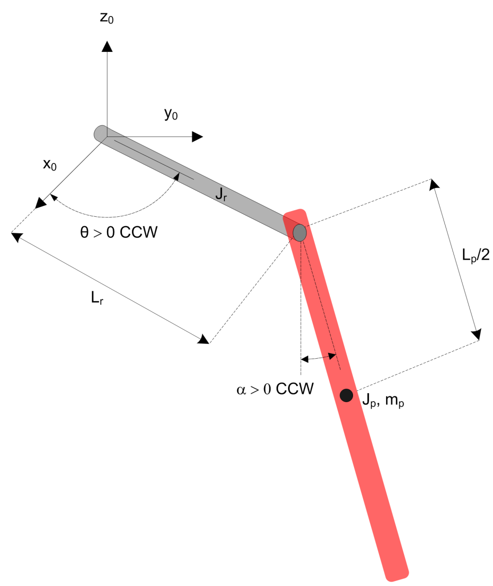

Figure 1: Rotary inverted pendulum model

The equations of motion (EOM) for the pendulum system were developed using the Euler-Lagrange method. This systematic method is often used to model complicated systems such as robot manipulators with multiple joints. The total kinetic and potential energy of the system is obtained, then the Lagrangian can be found. A number of derivatives are then computed to yield the EOMs. The resultant nonlinear EOMs are:

$$\left(m_p L_r^2 + \frac{1}{4} m_p L_p^2 - \frac{1}{4} m_p L_p^2 \cos(\alpha)^2 + J_r\right)\ddot{\theta} - \left(\frac{1}{2} m_p L_p L_r \cos(\alpha)\right)\ddot{\alpha}$$
$$+ \left(\frac{1}{2} m_p L_p^2 \sin(\alpha)\cos(\alpha)\right)\dot{\theta}\dot{\alpha} + \left(\frac{1}{2} m_p L_p L_r \sin(\alpha)\right)\dot{\alpha}^2 = \tau - D_r\dot{\theta}$$
$$\frac{1}{2} m_p L_p L_r \cos(\alpha)\ddot{\theta} + \left(J_p + \frac{1}{4} m_p L_p^2\right)\ddot{\alpha} - \frac{1}{4} m_p L_p^2 \cos(\alpha)\sin(\alpha)\dot{\theta}^2 + \frac{1}{2} m_p L_p g \sin(\alpha) = -D_p\dot{\alpha}.$$

Figure 2: Nonlinear EOMs

with an applied torque τ at the base of the rotary arm generated by the servo motor as described by the equation:

$$\tau = \frac{k_m(V_m - k_m\dot{\theta})}{R_m}$$

When the nonlinear EOM are linearized about the operating point, the resultan linear EOM for the inverted pendulum are defined as:

$$\left(m_p L_r^2 + J_r\right)\ddot{\theta} - \frac{1}{2}m_p L_p L_r \ddot{\alpha} = \tau - D_r\dot{\theta}.$$

$$\frac{1}{2}m_p L_p L_r \ddot{\theta} + \left(J_p + \frac{1}{4}m_p L_p^2\right)\ddot{\alpha} + \frac{1}{2}m_p L_p g\alpha = -D_p\dot{\alpha}.$$

Figure 3: Linear EOMs

2.1 Analysis

1. (0.5 point) Based on the sensors available on the pendulum system find the C and D matrices of the output of the linear system.

2. (1.0 point) Using Arduino-Software apply a 0-1 V, 1 Hz square wave to the pendulum system (QUBE-Servo 2) and capture the response of the system using the serial plotter. (Be sure that you measure the same variables that are the output of the linear system).

3. (1.0 point) Using the resultant linear EOMs, derive the linear state-space model of the pendulum of the system (Matrices A and B).

4. (1.2 point) Based on the state space model derived in Step 1 and 3, create a Mathematica model. The values of the main parameters associated to the QUBE-Servo 2 can be found in Table 2.2 of the User Manual. Moreover, set the rotary arm viscous damping coefficient D_r to $0.0015 N.m.s/rad$, and the pendulum damping coefficient D_p to $0.0005 N.m.s/rad$. These parameters were found experimentally. Apply a 0-1 V, 1 Hz square wave to the state-space model. Run the model. Attach figures of the simulation of your model.

5. (1.0 point) Does your model represent the actual pendulum well? If not, explain why there might be discrepancies.

6. (0.3 point)The viscous damping of each inverted pendulum can vary slightly from system to system. If your model does not accuratly represent your specific pendulum system, try modifying the damping coefficients D_r and D_p to obtain a more accurate model.