PENDULUM STATE SPACE MODELING

Topics Covered

- First-principles modeling
- State-space representation
- Model validation

Prerequisites

- Hardware Interfacing laboratory experiment.
- Rotary Pendulum Modeling laboratory experiment.

1 Background

1.1 Rotary Pendulum Model

The rotary pendulum model is shown in Figure 1.1. The rotary arm pivot is attached to the QUBE-Servo 2 system and is actuated. The arm has a length of 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 $l = L_p/2$. The moment of inertia about its center of mass is J_p. The rotary pendulum angle α is zero when it is hanging downward and increases positively when rotated CCW.

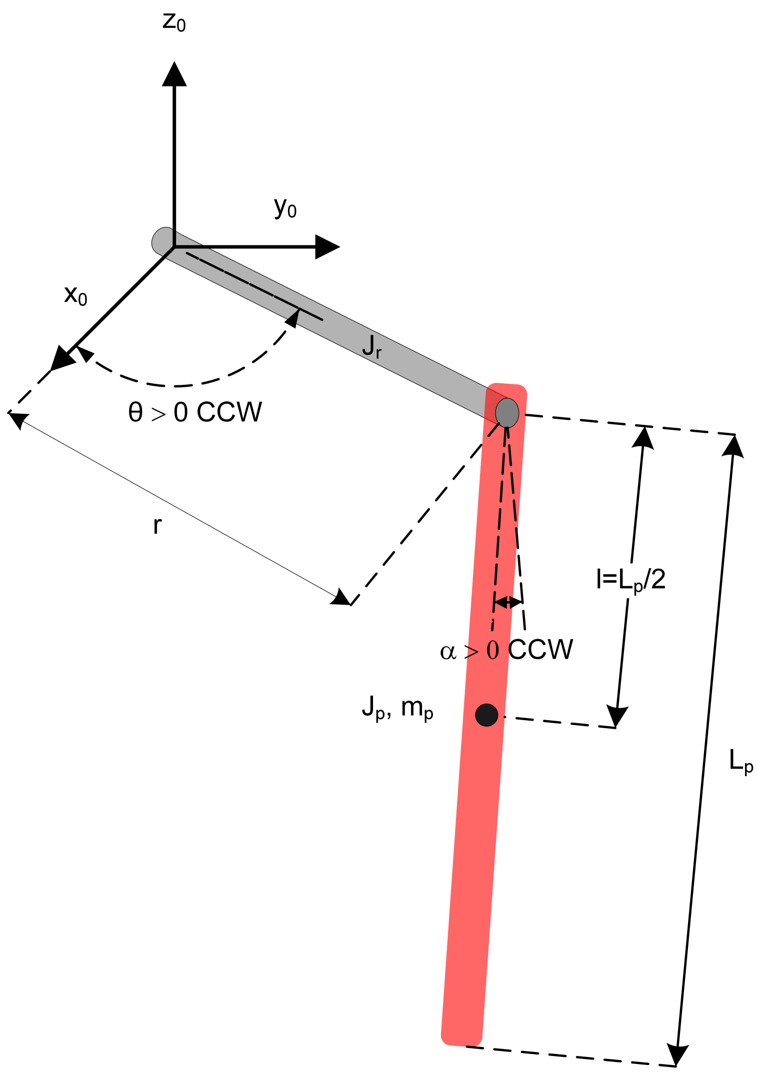


Figure 1.1: Rotary pendulum model

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 EOM are:

$$\left(J_r + J_p \sin\alpha^2\right)\ddot{\theta} + m_p l r \cos\alpha\ddot{\alpha} + 2J_p \sin\alpha\cos\alpha\dot{\theta}\dot{\alpha} - m_p l r \sin\alpha\dot{\alpha}^2 = \tau - b_r\dot{\theta} \tag{1.1}$$

and

$$J_p\ddot{\alpha} + m_p l r \cos\alpha\ddot{\theta} - J_p \sin\alpha\cos\alpha\dot{\theta}^2 + m_p g l \sin\alpha = -b_p\dot{\alpha}. \tag{1.2}$$

where $J_r = m_r r^2/3$ is the moment of inertia of the rotary arm with respect to the pivot (i.e. rotary arm axis of rotation) and $J_p = m_p L_p^2/3$ is the moment of inertia of the pendulum link relative to the pendulum pivot (i.e. axis of rotation of pendulum). The viscous damping acting on the rotary arm and the pendulum link are b_r and b_p, respectively. The applied torque at the base of the rotary arm generated by the servo motor is

$$\tau = \frac{k_m}{R_m}\left(v_m - k_m\dot{\theta}\right) \tag{1.3}$$

These equations of motion are based on the Furuta Pendulum model in the *Pendulum Equations* document by Dr. K. J. Åström, located in the *System Models* folder. Note that the equations in that document are for the rotary *inverted* pendulum, i.e. the equation above are for the pendulum hanging downwards, and use different modeling symbols. The complete derivation of the EOMs for a similar rotary pendulum system is presented in the *Rotary Pendulum HTML* in the *System Models* folder as well.

1.2 Linear Model

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

$$J_r\ddot{\theta} + m_p l r\ddot{\alpha} = \tau - b_r\dot{\theta} \tag{1.4}$$

and

$$J_p\ddot{\alpha} + m_p l r\ddot{\theta} + m_p g l\alpha = -b_p\dot{\alpha}. \tag{1.5}$$

Solving for the acceleration terms yields:

$$\ddot{\theta} = \frac{1}{J_t}\left(m_p^2 l^2 r g\alpha - J_p b_r\dot{\theta} + m_p l r b_p\dot{\alpha} + J_p\tau\right) \tag{1.6}$$

and

$$\ddot{\alpha} = \frac{1}{J_t}\left(-m_p g l J_r\alpha + m_p l r b_r\dot{\theta} - J_p b_p\dot{\alpha} - m_p r l\tau\right). \tag{1.7}$$

where

$$J_t = J_p J_r - m_p^2 l^2 r^2. \tag{1.8}$$

1.3 Linear State-Space Model

The linear state-space equations are

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{1.9}$$

and

$$y(t) = Cx(t) + Du(t) \tag{1.10}$$

where x is the vector of state variables $(n \times 1)$, u is the control input vector $(r \times 1)$, y is the output vector $(m \times 1)$, A is the system matrix $(n \times n)$, B is the input matrix $(n \times r)$, C is the output matrix $(m \times n)$, and D is the feed-forward matrix $(m \times r)$.

The block diagram representation of state-space equations is shown in Figure 1.2.

For the rotary pendulum system, the state and output are defined

$$x(t) = \begin{bmatrix}\theta(t) & \alpha(t) & \dot{\theta}(t) & \dot{\alpha}(t)\end{bmatrix}^T \tag{1.11}$$

and

$$y(t) = \begin{bmatrix}\theta(t) & \alpha(t)\end{bmatrix}^T. \tag{1.12}$$

The system state x defines the state variables needed to model the system and output state y defines the state variables that are measured directly.

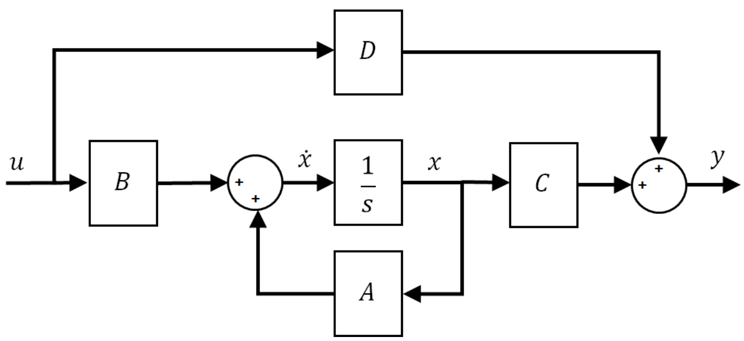


Figure 1.2: Block diagram of state-space system

2 In-Lab Exercises

2.1 Pendulum State-Space Model

1. A-1, A-3 Based on the output state defined above, find the state-space matrices C and D in Equation 1.10. Why are there only two variables defined in the output equation?

Answer 2.1

Outcome	**Solution**
A-1	Based on output equation defined in Equation 1.12, the C and D matrices are

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{Ans.2.1}$$

A-3	The output equation includes only the states that are measured. Only the rotary arm and and pendulum link angles are being measured in the QUBE-Servo 2 (i.e. not the angular rates), therefore only θ and α are defined.

□□□

2. A-1, A-2 Using Equation 1.6 and Equation 1.7 and the defined state in Equation 1.11, derive the linear state-space model of the pendulum system.

Answer 2.2

Outcome	**Solution**
A-1	From the state defined in Equation 1.11, it is given that $\dot{x}_1 = x_3$ and $\dot{x}_2 = x_4$. Substitute state x into the linear equations of motion given in Equation 1.6 and Equation 1.7, where we have $\theta = x_1, \alpha = x_2, \dot{\theta} = x_3, \dot{\alpha} = x_4$, and $u = \tau$. The A and B matrices for $\dot{x} = Ax + Bu$ can then be found.
A-2	Substituting x into Equation 1.6 and Equation 1.7 gives

$$\dot{x}_3 = \frac{1}{J_t}\left(m_p^2 l^2 r g x_2 - J_p b_r x_3 + m_p l r b_p x_4 + J_p u\right) \tag{Ans.2.2}$$

and

$$\dot{x}_4 = \frac{1}{J_t}\left(-m_p g l J_r x_2 + m_p l r b_r x_3 - J_r b_p x_4 - m_p l r u\right). \tag{Ans.2.3}$$

The A and B matrices in the $\dot{x} = Ax + Bu$ equation are

$$A = \frac{1}{J_T}\begin{bmatrix} 0 & 0 & J_T & 0 \\ 0 & 0 & 0 & J_T \\ 0 & m_p^2 l^2 r g & -J_p b_r & m_p l r b_p \\ 0 & -m_p g l J_r & m_p l r b_r & -J_r b_p \end{bmatrix} \quad \text{(Ans.2.4)}$$

and

$$B = \frac{1}{J_T}\begin{bmatrix} 0 \\ 0 \\ J_p \\ -m_p l r \end{bmatrix}. \quad \text{(Ans.2.5)}$$

□□□

3. B-5, K-3 In MATLAB®, write a script that models the rotary pendulum using the state-space representation matrices derived in Step 2. Use the model parameters defined in the `qube2_rotpen_param.m` script supplied (i.e. make sure you use the same variable names) and the `rotpen_ABCD_eqns_down_student.m` as a starting point.

   Note: The A, B, C, and D matrices need to be completed in `rotpen_ABCD_eqns_down_student.m`. Define the matrices as done in Step 2, when the control input is defined as torque, i.e. $u = \tau$. The last few lines of the scripts will convert the model to be with respect to motor voltage, i.e. $u = v_m$.

Answer 2.3

Outcome	Solution
B-5	If a script similarly as shown below was designed, then the student was able to use the software tools to model the system in state-space.
K-3	The Matlab script that models the rotary pendulum when the pendulum is in the downward position is given below. See the `rotpen_ABCD_eqns_down.m` for the full solution.

```
% Find Total Inertia
Jt = Jr*Jp - mp^2*l^2*r^2;
%
% State Space Representation
A = [0 0 1 0;
     0 0 0 1;
     0  mp^2*l^2*r*g/Jt    -br*Jp/Jt   mp*l*r*bp/Jt
     0  -mp*g*l*Jr/Jt      mp*l*r*br/Jt   -Jr*bp/Jt];

B = [0; 0; Jp/Jt; -mp*l*r/Jt];
C = eye(2,4);
D = zeros(2,1);
%
% Convert torque input to voltage input
B = km * B / Rm;
A(3,3) = A(3,3) - km*km/Rm*B(3);
A(4,3) = A(4,3) - km*km/Rm*B(4);
```

□□□

2.2 Model Validation

The SIMULINK® model below runs for 5 sec and applies a step voltage input to the QUBE-Servo 2 and its state-space model. The output scopes display the responses of the rotary arm and pendulum angle of the QUBE-Servo 2 (yellow) in parallel with the response from the linear model of the system (blue).

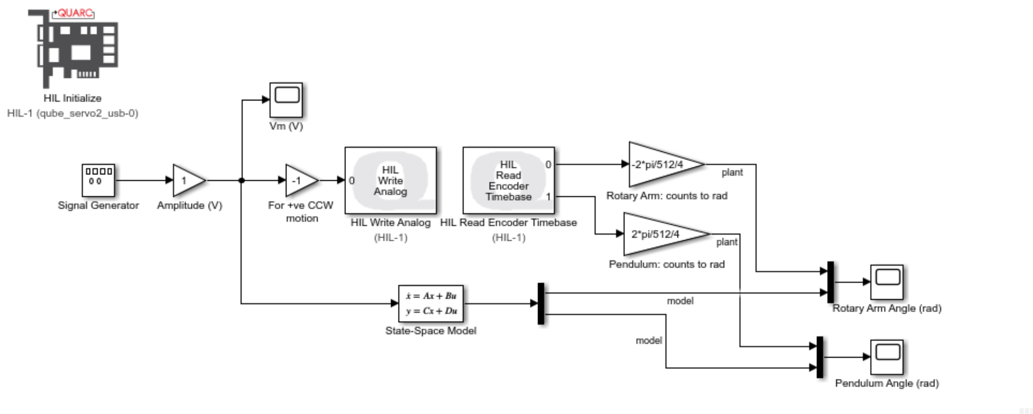


Figure 2.1: Applies a step voltage and displays measured and simulated pendulum response.

1. Open (if supplied) or design the SIMULINK® model shown in Figure 2.1 that applies a 1 V, 1 Hz square wave to the pendulum system and state-space model.

   Hint: If you are designing the Simulink diagram, use the model already developed in the Rotary Pendulum Modeling laboratory experiment.

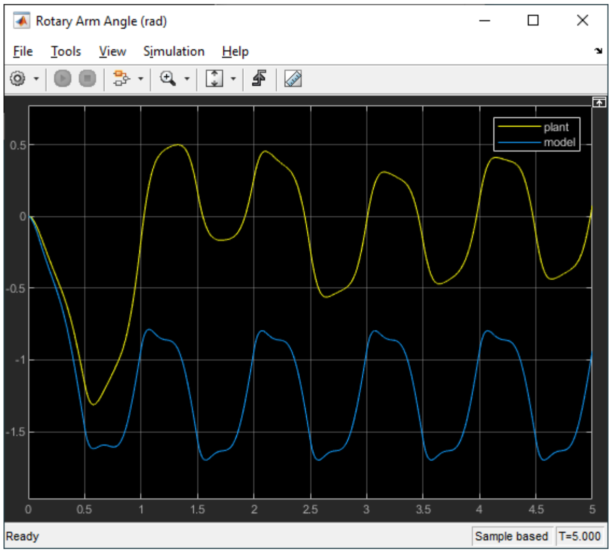
2. Run `setup_rotpen_ss_model.m` to create the state space model parameters in the MATLAB® workspace. Ensure that the generated matrices match your solution in Step 2.

3. In `qube2_rotpen_param.m`, set the rotary arm viscous damping coefficient b_r to 0.001 N.m.s/rad, and the pendulum damping coefficient b_p to 0.00005 N.m.s/rad. These parameters were found experimentally to reasonably accurately reflect the viscous damping of the system due to effects such as friction, when subject to a step response.

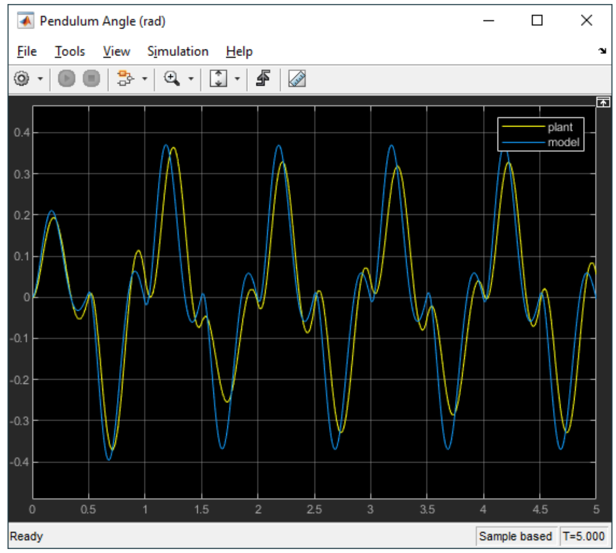
4. Build and run the QUARC controller.

5. B-5, K-2 The scope response should be similar to Figure 2.2. Attach the rotary pendulum response - showing both the measured and simulated (model based) rotary arm and pendulm angles.

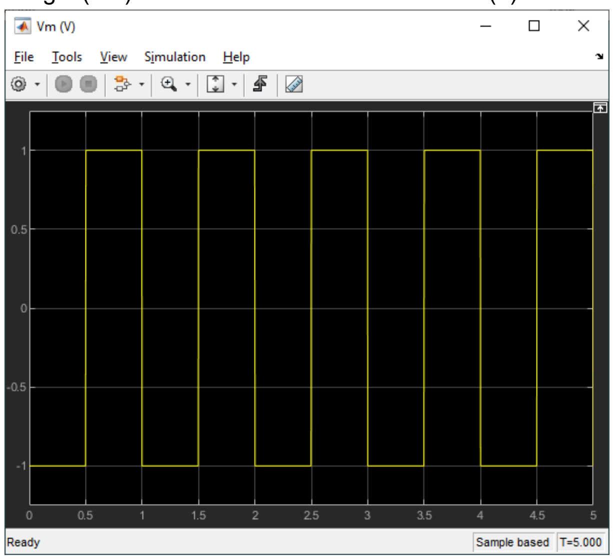
   Hint: For information on saving data to MATLAB® for offline analysis, see the QUARC® help documentation (under *QUARC Targets* | *User's Guide* | *QUARC Basics* | *Data Collection*). You can then use the MATLAB® `plot` command to generate the necessary Matlab figure.



(a) Rotary Arm Angle (rad)



(b) Pendulum Angle (rad)



(c) Motor Voltage (V)

Figure 2.2: Step response of the rotary pendulum system

Answer 2.4

Outcome	Solution
B-5	If the experimental procedure was followed correctly, the user should be able to run the model and obtain a response similar to Figure Ans.2.1.
K-2	The rotary pendulum response is shown in Figure Ans.2.1. This can be generated using the MATLAB® script commands:

```
% Load from variables set in workspace after running a Simulink model or
% from the previously saved response saved in the MAT files above.
t = data_vm(:,1);
u = data_vm(:,2);
arm_meas = data_theta(:,2);
arm_sim = data_theta(:,3);
pend_meas = data_alpha(:,2);
pend_sim = data_alpha(:,3);
%
%% Plot response
subplot(2,1,1);
plot(t,arm_meas,'r-',t,arm_sim,'b--');
ylabel('Rotary Arm (rad)');
legend('Measured', 'Simulated');
%
subplot(2,1,2);
plot(t,pend_meas,'r-',t,pend_sim,'b--');
ylabel('Pendulum (rad)');
xlabel('Time (s)');
legend('Measured', 'Simulated');
```

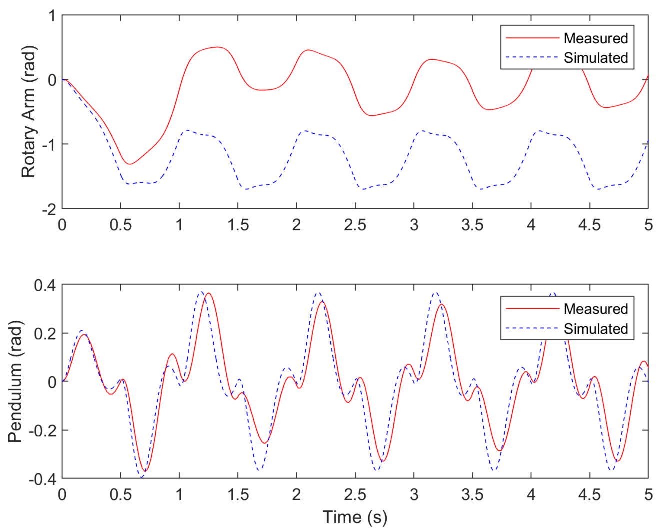


Figure Ans.2.1: Step response of the pendulum system.

☐☐☐

6. B-9 Does your model represent the actual pendulum well? If not, explain why there might be discrepancies.

Answer 2.5

Outcome	Solution
B-9	The pendulum response of the model represents the actual pendulum system accurately because in the simulated response (blue) matches the measured response (red) quite well in Figure Ans.2.1. The model of the arm response displays the same characteristics of the measured arm response, but an offset is observed. This is due to unmodeled dynamics such as the disturbance introduced by the encoder cable and the Coulomb friction in the motor.

□ □ □

7. The viscous damping of each pendulum can vary slightly from system to system. If your model does not accurately represent your specific pendulum system, try modifying the damping coefficients b_r and b_p to obtain a more accurate model.
8. Stop the QUARC® controller and power off the QUBE-Servo 2 if no more experiments will be conducted.

Quanser Inc.
119 Spy Court
Markham, Ontario
L3R 5H6
Canada
info@quanser.com
Phone: 1-905-940-3575
Fax: 1-905-940-3576

Printed in Markham, Ontario.

For more information on the solutions Quanser Inc. offers, please visit the web site at:
http://www.quanser.com