

Rotary Pendulum Position Control

ME 207 - Mechanical Engineering Laboratory III

March 18, 2024

Contents

1	Introduction and Goals	2
2	Modeling	3
3	Balance Control	7
4	Swing-Up Control	11
4.1	Energy Control	11
4.2	Hybrid Swing-Up Control	13
5	Appendix	16
5.1	Linearized Model for the Hanging (Downward) Pendulum	18
5.2	Linearized Model for the Inverted (Upward) Pendulum	18
5.3	Linearized Output Equation	18

1 Introduction and Goals

In this experiment you will work on the Qube-Servo 3 by Quanser (Fig. 1). The goal of this experiment is to control the position of the rotary pendulum. The successful completion of this experiment requires the execution of the following tasks:

- Obtain the transfer functions for both the hanging and the inverted pendulums.
- Validate the model against the nonlinear model and the actual hanging pendulum.
- Design a proportional-derivative controller for balancing the inverted pendulum.
- Test the balancing controller both in simulations and in experiments.
- Complement the designed balancing controller with a swing-up controller.

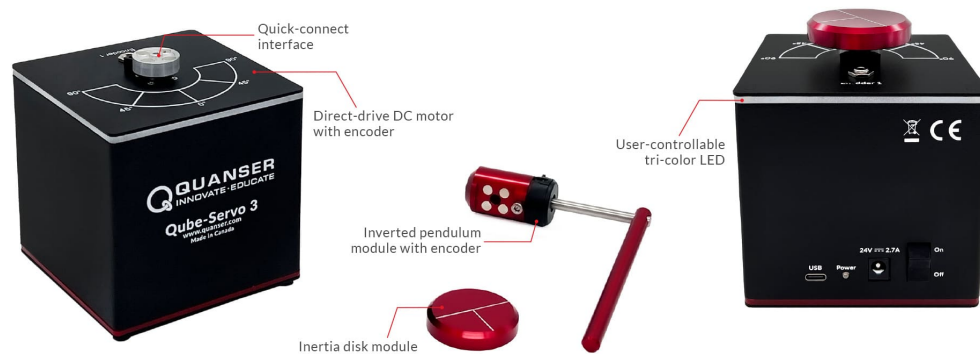


Figure 1: Qube-Servo 3

All the necessary files to carry out the experiment are available in the Google Drive folder shared with you. These files needed to design and test the controllers are in two subfolders:

- Design: These files are intended to be used “off-line” before or between sessions.
- Test: These files are intended to be used to run the experiment (Qube-Servo 3 must be connected to the computer). To run any of these SIMULINK[®] files, navigate to the “Hardware” tab in the top menu bar and select “Monitor and Tune.”

Lab Work 1:

1. Get familiar with Qube-Servo 3 and the provided MATLAB[®] /SIMULINK[®] files.

2 Modeling

The rotary pendulum model is shown in Fig. 2. The rotary arm pivot is attached to the Qube-Servo 3 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 counterclockwise (CCW). The servo (and thus the arm) should turn in the CCW direction when the control voltage is positive, i.e. $v_m > 0$. The pendulum link is connected to the end of the rotary arm. It has a total length of L_p and its 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.

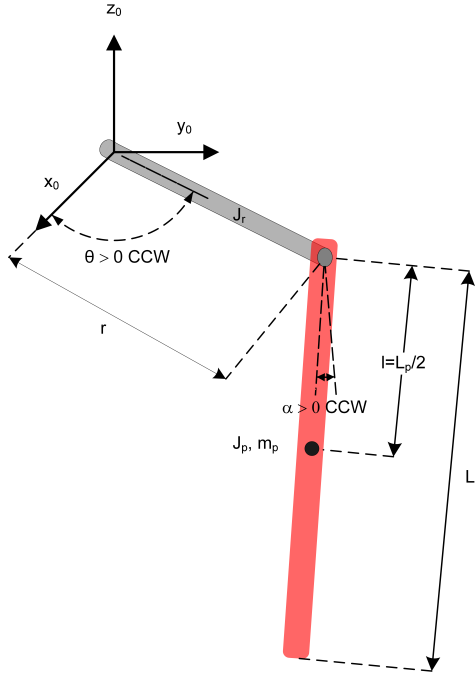


Figure 2: Rotary pendulum schematic.

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:

$$(J_r + J_p \sin^2 \alpha) \ddot{\theta} + m_p l r \cos \alpha \ddot{\alpha} - m_p l r \sin \alpha \dot{\alpha}^2 + 2J_p \sin \alpha \cos \alpha \dot{\theta} \dot{\alpha} = \tau - b_r \dot{\theta}, \quad (1)$$

$$m_p l r \cos \alpha \ddot{\theta} + J_p \ddot{\alpha} - J_p \sin \alpha \cos \alpha \dot{\theta}^2 + m_p g l \sin \alpha = -b_p \dot{\alpha}, \quad (2)$$

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

$$\tau = \frac{k_m}{R_m}(v_m - k_m\dot{\theta}). \quad (3)$$

At equilibrium, i.e. $\ddot{\theta} = \dot{\theta} = \ddot{\alpha} = \dot{\alpha} = 0$, these equations reduce to

$$0 = \tau_0, \quad (4)$$

$$m_p g l \sin \alpha_0 = 0. \quad (5)$$

An equilibrium point needs both $\tau_0 = 0$ and $m_p g l \sin \alpha_0 = 0 \iff \alpha_0 = 0, \alpha_0 = \pi$. The $\alpha_0 = 0$ solution is associated with the downward position of the pendulum, while the $\alpha_0 = \pi$ solution is associated with the upward position of the pendulum. Note that θ_0 can be arbitrary. We choose $\theta_0 = 0$ in this lab.

The linearized equations around the downward position ($\alpha_0 = 0$) are obtained in Appendix 5. From (54) it is possible to write ($\tilde{\theta} = \theta - \theta_0 = \theta$, $\tilde{\alpha} = \alpha - \alpha_0 = \alpha$)

$$\ddot{\tilde{\theta}} = \frac{1}{J_t}(m_p^2 l^2 r g \tilde{\alpha} - J_p(b_r + \frac{k_m^2}{R_m})\dot{\tilde{\theta}} + m_p l r b_p \dot{\tilde{\alpha}} + J_p \frac{k_m}{R_m} v_m), \quad (6)$$

$$\ddot{\tilde{\alpha}} = \frac{1}{J_t}(-m_p g l J_r \tilde{\alpha} + m_p l r(b_r + \frac{k_m^2}{R_m})\dot{\tilde{\theta}} - J_r b_p \dot{\tilde{\alpha}} - m_p r l \frac{k_m}{R_m} v_m). \quad (7)$$

The linearized equations around the upward position ($\alpha = \pi$) are obtained in Appendix 5. From (55) it is possible to write ($\tilde{\theta} = \theta - \theta_0 = \theta$, $\tilde{\alpha} = \alpha - \alpha_0 = \alpha - \pi$)

$$\ddot{\tilde{\theta}} = \frac{1}{J_t}(m_p^2 l^2 r g \tilde{\alpha} - J_p(b_r + \frac{k_m^2}{R_m})\dot{\tilde{\theta}} - m_p l r b_p \dot{\tilde{\alpha}} + J_p \frac{k_m}{R_m} v_m), \quad (8)$$

$$\ddot{\tilde{\alpha}} = \frac{1}{J_t}(m_p g l J_r \tilde{\alpha} - m_p l r(b_r + \frac{k_m^2}{R_m})\dot{\tilde{\theta}} - J_p b_p \dot{\tilde{\alpha}} + m_p r l \frac{k_m}{R_m} v_m). \quad (9)$$

In both cases,

$$J_t \triangleq J_p J_r - m_p^2 l^2 r^2. \quad (10)$$

The electrical and mechanical parameters are given in Table 1.

Lab Work 2:

1. Define $A(s) \triangleq \mathcal{L}\{\tilde{\alpha}(t)\}$, $\Theta(s) \triangleq \mathcal{L}\{\tilde{\theta}(t)\}$, $V(s) \triangleq \mathcal{L}\{v_m(t)\}$ and show that

$$\frac{\Theta(s)}{V(s)} = \frac{50.6372s^2 + 19.0182s + 5776.2}{s^4 + 13.1244s^3 + 268.9104s^2 + 1397.8s} \quad (11)$$

$$\frac{A(s)}{V(s)} = \frac{-50.0484s^2}{s^4 + 13.1244s^3 + 268.9104s^2 + 1397.8s} \quad (12)$$

for the hanging pendulum, and

$$\frac{\Theta(s)}{V(s)} = \frac{50.6372s^2 + 19.0182s - 5776.2}{s^4 + 13.1244s^3 - 259.7056s^2 - 1397.8s} \quad (13)$$

$$\frac{A(s)}{V(s)} = \frac{50.0484s^2}{s^4 + 13.1244s^3 - 259.7056s^2 - 1397.8s} \quad (14)$$

for the inverted pendulum.

Symbol	Description	Value/Units
R_m	Motor terminal resistance	8.4 Ω (ohms)
L_m	Motor inductance	1.16 mH (assumed approximately zero)
k_t	Motor torque constant	0.042 Nm/A (newton-meters per amp)
k_m	Motor back-electromotive force constant	0.042 V/(rad/s)
m_r	Rotary arm mass	0.095 (kg)
r	Rotary arm total length	0.085 m
J_r	Rotary arm moment of inertia	2.2879×10^{-4} kgm ²
b_r	Rotary arm damping coefficient	10^{-3} Nms/rad
m_p	Pendulum mass	0.024 kg
L_p	Pendulum total length	0.129 m
l	Pendulum center of mass	0.0645 m
J_p	Pendulum moment of inertia	1.3313×10^{-4} kgm ²
b_p	Pendulum damping coefficient	5×10^{-5} Nms/rad
g	Gravity constant	9.81 m/s ²
v_m	Motor terminal voltage	V (Volts)
θ	Rotary arm angle	rad
α	Pendulum angle	rad
$\dot{\theta}$	Rotary arm angular velocity	rad/s
$\dot{\alpha}$	Pendulum angular velocity	rad/s

Table 1: Pendulum parameters.

2. Compute the open-loop poles for the hanging-pendulum transfer function. Is there any pole in the right-hand-side (RHS) of the complex plane? Is the equilibrium at $\alpha_0 = 0$ stable?
3. Compute the open-loop poles for the inverted-pendulum transfer function. Is there any pole in the right-hand-side (RHS) of the complex plane? Is the equilibrium at $\alpha_0 = \pi$ stable?

NOTE: The rest of this lab will focus on the hanging pendulum since the instability associated with the inverted pendulum makes model validation very challenging if not impossible.

First validation step: Comparison of linearized model with nonlinear model

The SIMULINK[®] file MEM06_Qube_Pendulum_OLResponse_NonlinLinComp.slx provided in the “Design” directory will allow you to simulate the response of both nonlinear and linearized models. The nonlinear model results from the implementation of the nonlinear state-space representation (42)–(50). The linear model results from the implementation of the linear state-space representation (54)–(58) (the outputs of the model are the four state variables). Before running this SIMULINK[®] file, run the MATLAB[®] file MEM06_Qube_Pendulum_Model.m also provided in the “Design” directory and make sure to choose “Hanging Pendulum” (option 1) and “Full Output” (option 1). This file calls the MATLAB[®] file MEM06_Qube_Pendulum_Parameters.m, which is also located in the “Design” directory. The provided SIMULINK[®] file generates the input voltage v_m by combining a Signal Generator block in series with a

Gain block. Use a square voltage input with frequency of 1 Hz and amplitude ± 1 . Besides showing the simulated inputs and outputs in Scope blocks, SIMULINK[®] saves all the signals in a structure “simout” in the workspace. The MATLAB[®] file MEM06_Qube_Pendulum_OLResponse_NonlinLinComp_Plots.m in the same directory gives you an example on how to plot the signals from the workspace. All the plots in your report should follow this standard (screen captures of the SIMULINK[®] scopes with black backgrounds should not be used in the report). The default value for the signal generator gain is 0. The default values for the initial conditions of the states are also 0.

4. While keeping the signal generator gain at 0, compare responses by both nonlinear and linear models to an initial condition for α of $\pi/2$ by running the simulation for 2 seconds. This emulates the action of releasing the pendulum from a stationary horizontal position. Examine the “unforced” response of the pendulum. Choose the initial conditions of both linear and nonlinear models by double-clicking “Linear State-Space Model” and “Integrator 2,” respectively, and selecting the appropriate values.
5. Change the signal generator gain to 1 and choose zero initial conditions for both nonlinear and linear models. Compare responses by both nonlinear and linear models to the imposed input by running the simulation for 10 seconds. Examine the “forced” response of the pendulum.
6. Keep the signal generator gain at 1 and choose once again an initial condition for α of $\pi/2$ for both nonlinear and linear models. Compare responses by both nonlinear and linear models to the imposed input and assumed initial conditions by running the simulation for 6 seconds. Examine the “combined” response of the pendulum and identified the transient associated with the “unforced” response and the repetitive “forced” response. Provide plots in your report comparing responses by nonlinear and linear model for θ , α , $\dot{\theta}$ and $\dot{\alpha}$. Provide also a plot for the input v_m .

Second validation step: Comparison of linearized model with actual plant

The SIMULINK[®] file q_qube3_pen_ss_model.slx provided in the “Test” directory will allow you to compare the simulated response by the linear model with the actual response of the Qube-Servo 3. Before running this SIMULINK[®] file, run the MATLAB[®] file MEM06_Qube_Pendulum_Model.m also provided in the “Test” directory and make sure to choose “Hanging Pendulum” (option 1) and “Partial Output” (option 2). This file calls the MATLAB[®] file MEM06_Qube_Pendulum_Parameters.m, which is also located in the “Test” directory. Use a square voltage input with frequency of 1 Hz and amplitude ± 1 , and associated gain of 1. Make sure to select zero initial conditions for the linear model.

7. Build and run the QUARC[®] controller. Make sure that the actual initial conditions of the hanging pendulum are consistent with those selected for the linear model. Provide plots comparing both measured (actual) and simulated (model-based) rotary arm (θ) and pendulum (α) angles. You may want to add a “To Workspace” block in the provided SIMULINK[®] file (similar to how is done in the provided SIMULINK[®] file

MEM06_Qube_Pendulum_OLResponse_NonlinLinComp.slx) so that you can plot from MATLAB[®] the variables available in the workspace.

8. Does your model represent the actual pendulum well? If not, explain why there might be discrepancies.
9. 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 in the MEM06_Qube_Pendulum_Parameters.m file to obtain a more accurate model. Remember to run MEM06_Qube_Pendulum_Model.m again.
10. Stop the QUARC[®] controller. Power off the Qube-Servo 3 if no more experiments will be conducted.

3 Balance Control

Balancing is a common control task. In this experiment we will find control strategies that balance the pendulum in the upright position while maintaining a desired position of the arm. When balancing the system, the pendulum angle $\tilde{\alpha}$ is small and balancing can be accomplished with a simple PD controller, as shown in Fig. 3. If we are further interested in keeping the arm in a desired position, a feedback loop from the arm position will also be introduced.

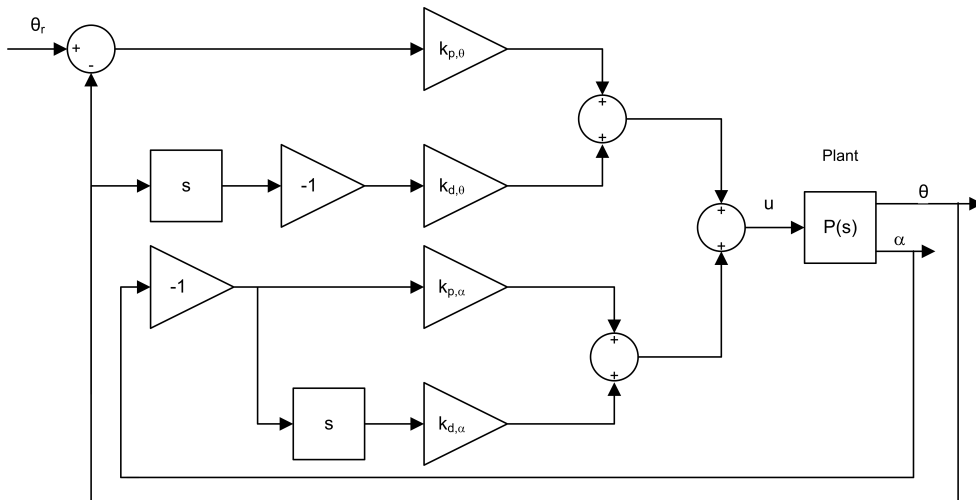


Figure 3: Block diagram of balance PD control for rotary pendulum.

The control law for $u = v_m$ can then be expressed as

$$v_m = -k_{p,\theta}(\tilde{\theta} - \tilde{\theta}_r) - k_{p,\alpha}\tilde{\alpha} - k_{d,\theta}\dot{\tilde{\theta}} - k_{d,\alpha}\dot{\tilde{\alpha}} \quad (15)$$

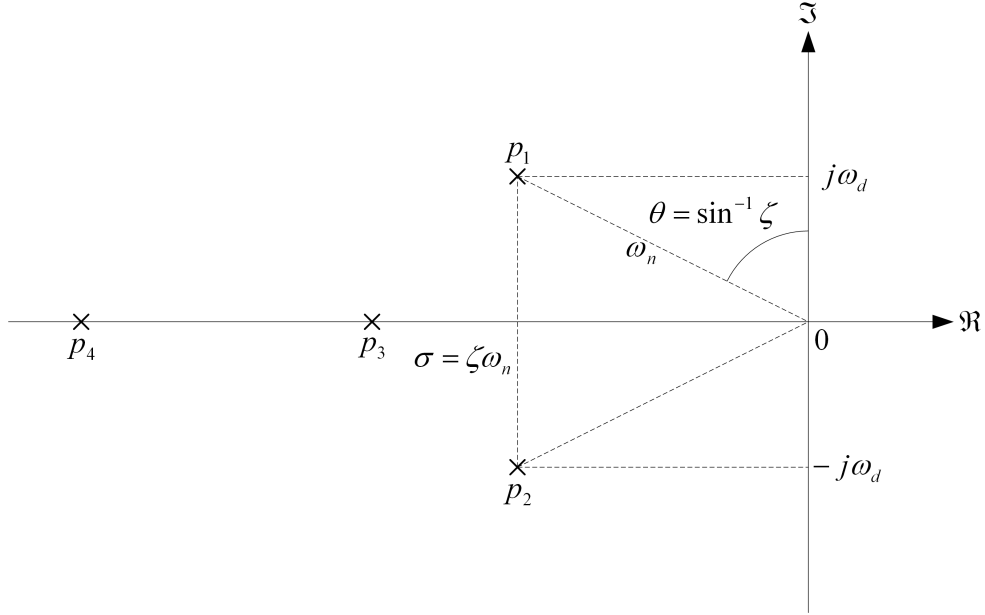


Figure 4: Desired closed-loop pole locations.

where $k_{p,\theta}$ is the arm angle proportional gain, $k_{p,\alpha}$ is the pendulum angle proportional gain, $k_{d,\theta}$ is the arm angle derivative gain, and $k_{d,\alpha}$ is the pendulum angle derivative gain. The desired, or reference, angle of the rotary arm is denoted by $\tilde{\theta}_r$. The reference for the pendulum angle is zero (i.e. upright position). The references for both time derivatives are also zero.

The balance control is to be enabled when the pendulum is within the following range:

$$|\alpha| \leq 10^\circ \quad (16)$$

Given that the pendulum starts in the downward vertical position, it needs to be manually brought up to its upright vertical position. Once the pendulum is within $\pm 10^\circ$, the balance controller is engaged. It remains in balance mode until the pendulum goes beyond $\pm 10^\circ$.

There are many different ways to find the controller parameters. In this lab you will use Pole Placement for the closed-loop transfer function. The question that follows is where to place the closed-loop poles. The rotary inverted pendulum system has four poles as you already noted from the transfer functions (8) and (9). Closing the loop with the PD controller (15) will also lead to a closed-loop transfer function with four poles. However, if two of the closed-loop poles are chosen to be closer to the imaginary axis (typically by a factor equal or greater than four) than the remaining poles, the conjugate poles are considered to be dominant and the system's behavior can be approximated by a second-order system. As depicted in Fig. 3, poles p_1 and p_2 are the complex conjugate dominant poles and are chosen to satisfy the natural frequency, ω_n , and damping ratio, ζ , second-order specifications. Let the conjugate poles be

$$p_1 = -\sigma + j\omega_d \quad (17)$$

$$p_2 = -\sigma - j\omega_d \quad (18)$$

where $\sigma = \zeta\omega_n$ and $\omega_d = \omega_n\sqrt{1-\zeta^2}$. The remaining closed-loop poles, p_3 and p_4 , are placed along the real axis to the left of the dominant poles, as shown in Fig. 3.

Lab Work 3:

1. The time-domain specifications when the pendulum is balanced and tracking a rotary arm setpoint are $M_p \leq 6.81\%$ and $t_s \leq 1.77$ sec. Thus, as the rotary arm goes back and forth to track the reference (while balancing the pendulum) it should have a percent overshoot M_p and settling time t_s matching these requirements. Recalling that

$$M_p[\%] = e^{-\pi\frac{\zeta}{\sqrt{1-\zeta^2}}} \times 100, \quad t_s = \frac{4}{\zeta\omega_n}, \quad (19)$$

calculate the minimum damping ratio ζ and natural frequency ω_n that satisfy these time-domain specifications and provide the resulting locations of the poles $p_1 = -\sigma + j\omega_d$, $p_2 = -\sigma - j\omega_d$.

2. Show that if the other poles are placed at $p_3 = -40$ and $p_4 = -45$, the desired characteristic polynomial is given by

$$s^4 + 90.20s^3 + 2257.80s^2 + 10714.95s + 28779.22 \quad (20)$$

Does this selection of p_3 and p_4 make p_1 and p_2 the dominant poles according to the discussion above around Fig. 3?

3. The control architecture used in this lab and presented in Fig. 15 represents a MIMO (Multi-Input-Multi-Output) closed-loop configuration that can also be represents in a more compact form as shown in Fig 3. Note that in this lab we will work with $\Theta_r(s) \neq 0$ and $A_r(s) \equiv 0$.

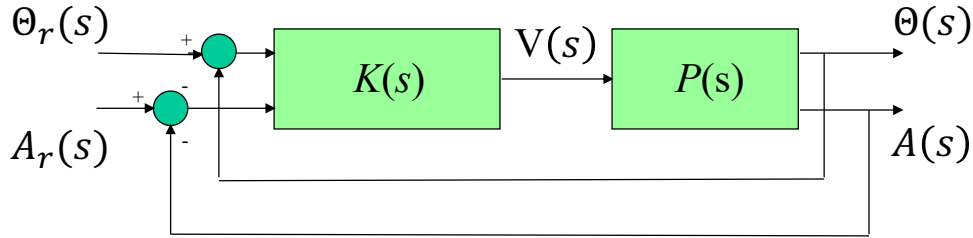


Figure 5: MIMO closed-loop configuration.

The closed-loop input-output relationship can be written in this case as

$$\begin{bmatrix} \Theta(s) \\ A(s) \end{bmatrix} = T(s) \begin{bmatrix} \Theta_r(s) \\ A_r(s) \end{bmatrix}, \quad (21)$$

where $T(s) = (I + P(s)K(s))^{-1}P(s)K(s)$, I is the identity matrix of dimension 2, and

$$P(s) = \begin{bmatrix} P_\theta(s) \\ P_\alpha(s) \end{bmatrix}, \quad K(s) = [k_{p,\theta} + k_{d,\theta}s \quad k_{p,\alpha} + k_{d,\alpha}s], \quad (22)$$

with $P_\theta(s) \triangleq \frac{\Theta(s)}{V(s)} = \frac{b_2s^2+b_1s+b_0}{s^2+a_3s^3+a_2s^2+a_1s}$ as defined in (13) and $P_\alpha(s) \triangleq \frac{A(s)}{V(s)} = \frac{c_2s^2}{s^2+a_3s^3+a_2s^2+a_1s}$ as defined in (14). The poles of the closed-loop transfer function $T(s)$ is given by the roots of the characteristic polynomial, which is in turn given by the determinant of $I + P(s)K(s)$ since $(I + P(s)K(s))^{-1} = \text{adj}(I + P(s)K(s))/\det(I + P(s)K(s))$. Show that the characteristic polynomial is written as

$$s^4 + (a_3 + c_2k_{d,\alpha} + b_2k_{d,\theta})s^3 + (a_2 + c_2k_{p,\alpha} + b_1k_{d,\theta} + b_2k_{p,\theta})s^2 + (a_1 + b_0k_{d,\theta} + b_1k_{p,\theta})s + b_0k_{p,\theta} \quad (23)$$

4. By comparing desired (20) and actual (23), obtain the controller gains $k_{p,\theta}$, $k_{d,\theta}$, $k_{p,\alpha}$, and $k_{d,\alpha}$.

Test of designed controller in Simulations

5. By using the SIMULINK[®] file `MEM06_Qube_Pendulum_CLResponse.slx` provided in the “Design” directory, test your design in linear simulations. Before running this SIMULINK[®] file, run the MATLAB[®] file `MEM06_Qube_Pendulum_Model.m` in the “Design” directory and make sure to make the controller gains `kp_theta_n`, `kd_theta_n`, `kp_alpha_n` and `kd_alpha_n`. This file calls `MEM06_Qube_Pendulum_Parameters.m`, which is also located in the “Design” directory. Use a square voltage input with frequency of 0.25 Hz and amplitude ± 1 , and associated gain of 1. Make sure to select zero initial conditions for the linear model. Provide plots for θ vs θ_r , α , and v_m .
6. By using the SIMULINK[®] file `MEM06_Qube_Pendulum_CLResponse_NL.slx` provided in the “Design” directory, test now your design in nonlinear simulations. Before running this SIMULINK[®] file, run the MATLAB[®] file `MEM06_Qube_Pendulum_Model.m` also provided in the “Design” directory and make sure to make the controller gains `kp_theta_n`, `kd_theta_n`, `kp_alpha_n` and `kd_alpha_n`. This file calls the MATLAB[®] file `MEM06_Qube_Pendulum_Parameters.m`, which is also located in the “Design” directory. Use a square voltage input with frequency of 0.25 Hz and amplitude ± 1 , and associated gain of 1. Make sure to select zero initial conditions for the linear model. Provide plots for θ vs θ_r , α , and v_m .

Test of designed controller in Experiments

7. The SIMULINK[®] file `q_qube3_bal_pp.slx` provided in the “Test” directory will allow you to test the performance of the designed controller on the Qube-Servo 3. Before running this SIMULINK[®] file, define the gain vector

$$K = [k_{p,\theta} \quad k_{p,\alpha} \quad k_{d,\theta} \quad k_{d,\alpha}] \quad (24)$$

and make sure to load it to the MATLAB[®] workspace. Set the “Signal Generator” block as follows:

- Type = Square
- Amplitude = 1
- Frequency = 0.125 Hz

Set the “Gain” block that is connected to the “Signal Generator” to 0.

8. Build and run the QUARC[®] controller.
9. Manually rotate the pendulum in the upright position until the controller engages.
10. Once the pendulum is balanced, set the “Gain” to 30 to make the arm angle go between $\pm 30^\circ$. Provide plots in your report showing the response of the rotary arm, pendulum, and controller voltage.
11. Does the rotary arm and pendulum response match the settling time and percent overshoot specifications the controller was designed for? If not, give one reason why there is a discrepancy.
12. Stop the QUARC[®] controller.
13. Power off the QUBEServo 2 if no more experiments will be conducted.

4 Swing-Up Control

In this lab, a nonlinear control system is developed to swing the pendulum from the downward, hanging down, position to the upright vertical position. In order to do this, an energy-based control will be developed that will calculate the acceleration (i.e. motor voltage) necessary to swing the pendulum up in the inverted position. Once it reaches the upright vertical position, the balance control designed and tested in the previous lab will be engaged to stabilize the pendulum.

4.1 Energy Control

In theory, if the arm angle is kept constant and the pendulum is given an initial perturbation, the pendulum will keep on swinging with constant amplitude. The idea of energy control is based on the preservation of energy in ideal systems: The sum of kinetic and potential energy is constant. However, friction will be damping the oscillation in practice and the overall system energy will not be constant. It is possible to capture the loss of energy with respect to the pivot acceleration, which in turn can be used to find a controller to swing up the pendulum. The nonlinear equation of motion of a single pendulum based on the diagram in Fig 4.1 is given by

$$J_p \ddot{\alpha}(t) + m_p g l \sin \alpha(t) + m_p l u(t) \cos \alpha(t) = 0 \quad (25)$$

where $\alpha(t)$ is the angle of the pendulum defined as positive when rotated counter-clockwise, J_p is the moment of inertia with respect to the pivot point, m_p is the mass of the pendulum link, l is the distance between the pivot and the center of mass, and $u(t)$ is the linear acceleration of the pendulum pivot (positive along the x_0 axis).

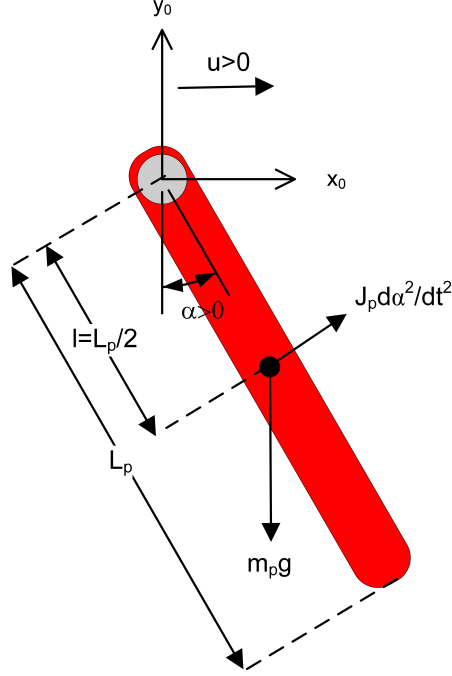


Figure 6: Free-body diagram of pendulum.

The potential energy of the pendulum is

$$E_p = m_p g l (1 - \cos \alpha) \quad (26)$$

and the kinetic energy is

$$E_k = \frac{1}{2} J_p \dot{\alpha}^2. \quad (27)$$

The potential energy is zero when the pendulum is at rest at $\alpha = 0$ and equals $E_p = 2m_p g l$ when the pendulum is upright at $\alpha = \pm\pi$. The sum of the potential and kinetic energy of the pendulum is

$$E = \frac{1}{2} J_p \dot{\alpha}^2 + m_p g l (1 - \cos \alpha). \quad (28)$$

Differentiating (28) yields

$$\dot{E} = \frac{dE}{dt} = J_p \dot{\alpha} \ddot{\alpha} + m_p g l \sin \alpha \dot{\alpha}. \quad (29)$$

Solving for $J_p \ddot{\alpha}$ in (25) yields

$$J_p \ddot{\alpha} = -m_p g l \sin \alpha - m_p l u \cos \alpha \quad (30)$$

and substituting this into (29) gives

$$\dot{E} = -m_p l u \dot{\alpha} \cos \alpha. \quad (31)$$

Since the acceleration of the pivot is proportional to current driving the arm motor and thus also proportional to the motor voltage, it is possible to control the energy of the pendulum with the proportional control law

$$u = (E - E_r)\dot{\alpha} \cos \alpha. \quad (32)$$

This control law will drive the energy of the pendulum towards the reference energy, i.e. $E(t) \rightarrow E_r$. By setting the reference energy to the pendulum potential energy, $E_r = E_p$, the control law will swing the link to its upright position. Notice that the control law is nonlinear because it includes nonlinear terms (e.g. $\cos \alpha$). Further, the control changes sign when $\dot{\alpha}$ changes sign and when the angle is ± 90 degrees. For the system energy to change quickly, the magnitude of the control signal must be large. As a result the following swing-up controller is implemented in the controller as

$$u = sat_{u_{max}}(k_e(E - E_r)sign(\dot{\alpha} \cos \alpha)) \quad (33)$$

where k_e is a tunable control gain and the $sat_{u_{max}}$ function saturates the control signal at the maximum acceleration of the pendulum pivot, u_{max} . The expression $sign(\dot{\alpha} \cos \alpha)$ is used to enable faster control switching. The control law in (33) finds the linear acceleration needed to swing-up the pendulum. Because the control variable in the QUBE-Servo 2 is motor voltage, $v_m(t)$, the acceleration needs to be converted into voltage. This can be done using the expression

$$v_m(t) = \frac{R_m r m_r}{k_t} u(t) \quad (34)$$

where R_m is the motor resistance, k_t is the current-torque constant of the motor, r is the length of the rotary arm, and m_r is the mass of the rotary arm. The block diagram of the swing-up nonlinear control is shown in Fig 4.1. Energy Control Implementation Based on Lyapunov stability, it can be shown that different energy-type quantities can be used for the swing-up, i.e., not only $E = \frac{1}{2}J_p\dot{\alpha}^2 + m_pgl(1 - \cos \alpha)$. In the actual implementation, the following pendulum energy equation is used

$$E = \frac{1}{2}J_{p,cm}\dot{\alpha}^2 + m_pgl(1 - \cos \alpha) \quad (35)$$

where $J_{p,cm}$ is the moment of inertia of the pendulum with respect to the center of mass (as opposed to the pivot J_p). Using this, we can perform the swing-up with a lower tunable gain and reference energy.

4.2 Hybrid Swing-Up Control

The energy swing-up control defined in (33) can be combined with a balancing control, such as the one designed in the previous lab, to obtain a control system that swings up the pendulum and then balances it. Similarly as described in the Balance Control lab, the balance control is to be enabled when the pendulum is within ± 20 degrees. When it is not enabled, the swing-up control is engaged. Thus the switching can be described mathematically by

$$u = \begin{cases} u_{bal} & \text{if } |\alpha - \pi| \leq 0.345rad \\ u_{swing-up} & \text{otherwise} \end{cases} \quad (36)$$

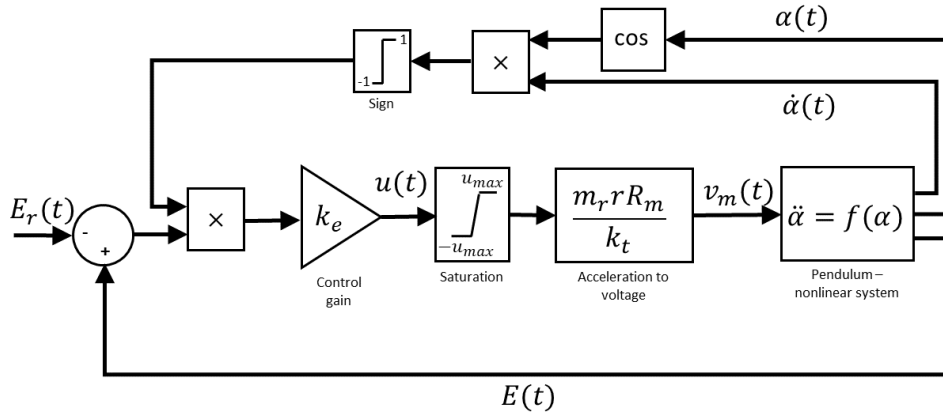


Figure 7: Energy swing-up control of pendulum.

Lab Work 4:

Energy Control

1. Open the SIMULINK[®] file `q_qube3.swingup.slx` provided in the “Test” directory.
2. Run the `setup.swingup.student.m` MATLAB[®] script. This loads the pendulum parameters that is used by the Simulink model.
3. To turn the swing-up control off, set the “Slider Gain” block called k_e to 0.
4. Build and run the QUARC[®] controller.
5. Manually rotate the pendulum at different levels and examine the pendulum angle and energy in the Pendulum (*deg* and *Pendulum Energy (mJ)* scopes).
6. What do you notice about the energy when the pendulum is moved at different positions? Record the energy when the pendulum is being balanced (i.e. fully inverted in the upright vertical position). Does this reading make sense in terms of the equations developed above?
7. Click on the Stop button to bring the pendulum down to the initial, downward position.
8. Set the swing-up control parameters (i.e. the “Constant” and “Gain” blocks connected to the inputs of the “Swing-Up Control” subsystem) to the following:
 - $k_e = 50\text{m/s/J}$
 - $E_r = 10.0\text{mJ}$
 - $u_{max} = 6\text{m/s}^2$
9. If the rotary arm does not start rotating back and forth, gently perturb the pendulum with your finger to get it going.

10. Vary the reference energy, E_r , between 10.0mJ and 20.0mJ. As it is changed, examine the pendulum angle and energy response in *Pendulum (deg)* and the *Pendulum Energy (mJ)* scopes and the control signal in the *Vm (V)* scope. Provide MATLAB[®] versions of these plots in your report, showing how changing the reference energy affects the system.
11. Fix E_r to 20.0mJ and vary the swing-up control gain k_e between 20 and 60m/s²/J. Describe how this changes the performance of the energy control.
12. Stop the QUARC[®] controller.

Hybrid Swing-Up Control

13. Open the SIMULINK[®] file `q_qube3.swingup.slx` provided in the “Test” directory.
14. Run the `setup.swingup.student.m` MATLAB[®] script. This loads the pendulum parameters that is used by the Simulink model.
15. Set the swing-up control parameters to the following:
 - $k_e = 20\text{m/s/J}$
 - $u_{max} = 6\text{m/s}^2$
16. Based on your observations in the previous section of the lab (Energy Control), what should the reference energy be set to?
17. Make sure the pendulum is hanging down motionless and the encoder cable is not interfering with the pendulum.
18. Build and run the QUARC[®] controller.
19. The pendulum should begin going back and forth. If not, manually perturb the pendulum with your hand. **Click on the “Stop” button in the SIMULINK[®] tool bar if the pendulum goes unstable.**
20. Gradually increase the swing-up gain, k_e , denoted as the “ k_e Slider Gain” block, until the pendulum swings up to the vertical position. Capture a response of the swing-up and record the swing-up gain that was required. Show the pendulum angle, pendulum energy, and motor voltage.
21. Stop the QUARC[®] controller.

5 Appendix

The equations of motion (1) and (2) can be written as

$$\begin{bmatrix} (J_r + J_p \sin^2 \alpha) & m_p l r \cos \alpha \\ m_p l r \cos \alpha & J_p \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{\alpha} \end{bmatrix} + \begin{bmatrix} b_r \dot{\theta} + 2J_p \sin \alpha \cos \alpha \dot{\theta} \dot{\alpha} - m_p l r \sin \alpha \dot{\alpha}^2 - \tau \\ -J_p \sin \alpha \cos \alpha \dot{\theta}^2 + m_p g l \sin \alpha + b_p \dot{\alpha} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (37)$$

$$\underbrace{\begin{bmatrix} (J_r + J_p \sin^2 \alpha) & m_p l r \cos \alpha \\ m_p l r \cos \alpha & J_p \end{bmatrix}}_A \begin{bmatrix} \ddot{\theta} \\ \ddot{\alpha} \end{bmatrix} = \begin{bmatrix} -b_r \dot{\theta} - 2J_p \sin \alpha \cos \alpha \dot{\theta} \dot{\alpha} + m_p l r \sin \alpha \dot{\alpha}^2 + \tau \\ J_p \sin \alpha \cos \alpha \dot{\theta}^2 - m_p g l \sin \alpha - b_p \dot{\alpha} \end{bmatrix} \quad (38)$$

We compute the inverse matrix as $A^{-1} = \text{adj}(A) / \det(A)$, where

$$\det(A) = (J_r + J_p \sin^2 \alpha) J_p - m_p l r \cos \alpha m_p l r \cos \alpha = (J_r J_p + J_p^2 \sin^2 \alpha) - m_p^2 l^2 r^2 \cos^2 \alpha \quad (39)$$

and

$$\text{adj}(A) = \begin{bmatrix} J_p & -m_p l r \cos \alpha \\ -m_p l r \cos \alpha & (J_r + J_p \sin^2 \alpha) \end{bmatrix} \quad (40)$$

Then, we can solve for the acceleration vector from (38) as

$$\begin{bmatrix} \ddot{\theta} \\ \ddot{\alpha} \end{bmatrix} = \frac{\begin{bmatrix} J_p & -m_p l r \cos \alpha \\ -m_p l r \cos \alpha & (J_r + J_p \sin^2 \alpha) \end{bmatrix} \begin{bmatrix} -b_r \dot{\theta} - 2J_p \sin \alpha \cos \alpha \dot{\theta} \dot{\alpha} + m_p l r \sin \alpha \dot{\alpha}^2 + \tau \\ J_p \sin \alpha \cos \alpha \dot{\theta}^2 - m_p g l \sin \alpha - b_p \dot{\alpha} \end{bmatrix}}{(J_r J_p + J_p^2 \sin^2 \alpha) - m_p^2 l^2 r^2 \cos^2 \alpha} \quad (41)$$

Therefore, we can finally write

$$\begin{aligned} \ddot{\theta} &= \frac{J_p \left(-b_r \dot{\theta} - 2J_p \sin \alpha \cos \alpha \dot{\theta} \dot{\alpha} + m_p l r \sin \alpha \dot{\alpha}^2 + \tau \right)}{(J_r J_p + J_p^2 \sin^2 \alpha) - m_p^2 l^2 r^2 \cos^2 \alpha} \\ &\quad - \frac{m_p l r \cos \alpha \left(J_p \sin \alpha \cos \alpha \dot{\theta}^2 - m_p g l \sin \alpha - b_p \dot{\alpha} \right)}{(J_r J_p + J_p^2 \sin^2 \alpha) - m_p^2 l^2 r^2 \cos^2 \alpha} \\ \ddot{\alpha} &= \frac{-m_p l r \cos \alpha \left(-b_r \dot{\theta} - 2J_p \sin \alpha \cos \alpha \dot{\theta} \dot{\alpha} + m_p l r \sin \alpha \dot{\alpha}^2 + \tau \right)}{(J_r J_p + J_p^2 \sin^2 \alpha) - m_p^2 l^2 r^2 \cos^2 \alpha} \\ &\quad + \frac{(J_r + J_p \sin^2 \alpha) \left(J_p \sin \alpha \cos \alpha \dot{\theta}^2 - m_p g l \sin \alpha - b_p \dot{\alpha} \right)}{(J_r J_p + J_p^2 \sin^2 \alpha) - m_p^2 l^2 r^2 \cos^2 \alpha} \end{aligned}$$

Defining the state variable as $x^T = [x_1 \ x_2 \ x_3 \ x_4]$ with $x_1 = \theta$, $x_2 = \alpha$, $x_3 = \dot{\theta}$, $x_4 = \dot{\alpha}$, and the input as $u = \tau$, we can write the system dynamics in state-space form as

$$\dot{x} = f(x, u) \quad (42)$$

with

$$f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \\ f_4(x) \end{bmatrix} \quad (43)$$

and

$$f_1(x) = x_3 \quad (44)$$

$$f_2(x) = x_4 \quad (45)$$

$$f_3(x) = \frac{J_p(-b_r x_3 - 2J_p \sin x_2 \cos x_2 x_3 x_4 + m_p l r \sin x_2 x_4^2 + \tau)}{(J_r J_p + J_p^2 \sin^2 x_2) - m_p^2 l^2 r^2 \cos^2 x_2} \quad (46)$$

$$- \frac{m_p l r \cos x_2 (J_p \sin x_2 \cos x_2 x_3^2 - m_p g l \sin x_2 - b_p x_4)}{(J_r J_p + J_p^2 \sin^2 x_2) - m_p^2 l^2 r^2 \cos^2 x_2} \quad (47)$$

$$f_4(x) = \frac{-m_p l r \cos x_2 (-b_r x_3 - 2J_p \sin x_2 \cos x_2 x_3 x_4 + m_p l r \sin x_2 x_4^2 + \tau)}{(J_r J_p + J_p^2 \sin^2 x_2) - m_p^2 l^2 r^2 \cos^2 x_2} \quad (48)$$

$$+ \frac{(J_r + J_p \sin^2 x_2) (J_p \sin x_2 \cos x_2 x_3^2 - m_p g l \sin x_2 - b_p x_4)}{(J_r J_p + J_p^2 \sin^2 x_2) - m_p^2 l^2 r^2 \cos^2 x_2} \quad (49)$$

With this selection of state variable and input, equation (3) can be written as

$$u = -\frac{k_m^2}{R_m} x_3 + \frac{k_m}{R_m} v_m. \quad (50)$$

The model of the Furuta pendulum (42)–(49) is highly nonlinear. To enable control design we are interested in obtaining an approximate linear model. As a first step, the function $f(x)$ is written as a Taylor series expansion around (x_0, u_0) , i.e.

$$f(x) = f(x)|_{(x_0, u_0)} + \frac{\partial f}{\partial x} \Big|_{(x_0, u_0)} (x - x_0) + \frac{\partial f}{\partial u} \Big|_{(x_0, u_0)} (u - u_0) + H.O.T. \quad (51)$$

As a second step, the higher order terms (H.O.T.), that is the nonlinear terms, are neglected to approximate the function $f(x)$ as

$$f(x) \approx f(x, u)|_{(x_0, u_0)} + \underbrace{\frac{\partial f}{\partial x} \Big|_{(x_0, u_0)}}_A \underbrace{(x - x_0)}_{\tilde{x}} + \underbrace{\frac{\partial f}{\partial u} \Big|_{(x_0, u_0)}}_B \underbrace{(u - u_0)}_{\tilde{u}} \quad (52)$$

We are interested in obtaining linearized models for the two equilibria corresponding to the hanging pendulum and the inverted pendulum, i.e. for $x_0^2 = \alpha_0 = 0$ and $x_0^2 = \alpha_0 = \pi$. As it was discussed in Section 2, $x_0^1 = \theta_0$ can be nonzero but both x_0^3 and x_0^4 (time derivatives) must be zero. Note that by using the state-space representation (42), the equilibrium point (x_0, u_0) must make $\dot{x} = 0$, which requires $f(x_0, u_0) \equiv 0$. Making $f_1(x_0, u_0) = 0$ demands $x_0^3 = 0$ and making $f_2(x_0, u_0) = 0$ demands $x_0^4 = 0$. Moreover, making $f_3(x_0, u_0) = 0$ and $f_4(x_0, u_0) = 0$ demands $x_0^2 = \alpha_0 = 0$ or $x_0^2 = \alpha_0 = \pi$. Note that $f(x, u)$ does not depend on x_1 . Therefore, the equilibrium can be achieved for $u_0 = 0$ for any value of $x_0^1 = \theta_0$. We will choose $x_0^1 = \theta_0 = 0$ below for convenience.

5.1 Linearized Model for the Hanging (Downward) Pendulum

In this case, $x_0 = (0, 0, 0, 0)$ (downward position) and $u_0 = 0$. Since this is an equilibrium of the system, $f(x, u)|_{(x_0, u_0)} = 0$. Therefore, we can write the linearized model as

$$\dot{\tilde{x}} = A\tilde{x} + B\tilde{u}, \quad A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{m_p^2 l^2 g r}{J_t} & -\frac{J_p b_r}{J_t} & \frac{m_p l r b_p}{J_t} \\ 0 & -\frac{m_p l g J_r}{J_t} & \frac{m_p l r b_r}{J_t} & -\frac{J_r b_p}{J_t} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \frac{J_p}{J_t} \\ -\frac{m_p l r}{J_t} \end{bmatrix}. \quad (53)$$

where $J_t = J_p J_r - m_p^2 l^2 r^2$. Note that in this case $\tilde{\theta} = \theta - \theta_0 = \theta$, $\tilde{\alpha} = \alpha - \alpha_0 = \alpha$ and $\tilde{u} = u - u_0 = u$. Replacing the expression for $\tilde{u} = u$ from (50) in (53) yields a linear state representation

$$\dot{\tilde{x}} = A\tilde{x} + Bv_m, \quad A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{m_p^2 l^2 g r}{J_t} & -\frac{J_p}{J_t} (b_r + \frac{k_m^2}{R_m}) & \frac{m_p l r b_p}{J_t} \\ 0 & -\frac{m_p l g J_r}{J_t} & \frac{m_p l r}{J_t} (b_r + \frac{k_m^2}{R_m}) & -\frac{J_r b_p}{J_t} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \frac{J_p k_m}{J_t R_m} \\ -\frac{m_p l r k_m}{J_t R_m} \end{bmatrix}. \quad (54)$$

5.2 Linearized Model for the Inverted (Upward) Pendulum

In this case, $x_0 = (\theta_0, \pi, 0, 0)$ (upward position) and $u_0 = 0$. Since this is an equilibrium of the system, $f(x, u)|_{(x_0, u_0)} = 0$. Therefore, we can write the linearized model as

$$\dot{\tilde{x}} = A\tilde{x} + B\tilde{u}, \quad A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{m_p^2 l^2 g r}{J_t} & -\frac{J_p b_r}{J_t} & -\frac{m_p l r b_p}{J_t} \\ 0 & \frac{m_p l g J_r}{J_t} & -\frac{m_p l r b_r}{J_t} & -\frac{J_r b_p}{J_t} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \frac{J_p}{J_t} \\ \frac{m_p l r}{J_t} \end{bmatrix}. \quad (55)$$

where $J_t = J_p J_r - m_p^2 l^2 r^2$. Note that in this case $\tilde{\theta} = \theta - \theta_0 = \theta$, $\tilde{\alpha} = \alpha - \alpha_0 = \alpha - \pi$ and $\tilde{u} = u - u_0 = u$. Replacing the expression for $\tilde{u} = u$ from (50) in (53) yields a linear state representation

$$\dot{\tilde{x}} = A\tilde{x} + Bv_m, \quad A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{m_p^2 l^2 g r}{J_t} & -\frac{J_p}{J_t} (b_r + \frac{k_m^2}{R_m}) & -\frac{m_p l r b_p}{J_t} \\ 0 & \frac{m_p l g J_r}{J_t} & -\frac{m_p l r}{J_t} (b_r + \frac{k_m^2}{R_m}) & -\frac{J_r b_p}{J_t} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \frac{J_p k_m}{J_t R_m} \\ \frac{m_p l r k_m}{J_t R_m} \end{bmatrix}. \quad (56)$$

5.3 Linearized Output Equation

The state-space equation $\dot{\tilde{x}} = A\tilde{x} + Bv_m$ is usually paired with an output equation of the form

$$y = C\tilde{x} + Dv_m. \quad (57)$$

This output equation enables the selection of linear combinations of the states as outputs of the model. For instance, if we are interested in an output identical to the state, these matrices should be chosen as

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad D = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \Rightarrow y = \begin{bmatrix} \theta \\ \alpha \\ \dot{\theta} \\ \dot{\alpha} \end{bmatrix} \quad (58)$$

On the other hand, if we are interested in an output consisting of just the angles, these matrices should be chosen as

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad D = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \Rightarrow y = \begin{bmatrix} \theta \\ \alpha \end{bmatrix} \quad (59)$$