Rotary Inverted Pendulum

EE615 Control and Computational Laboratory

Tejas Sanjaykumar Pagare | 190070067

Rathour Param Jitendrakumar | 190070049

Spring Semester 2022-23

Contents

1 Introduction

1.1 Aim

The purpose of this experiment is to understand and implement rotary inverted pendulum by designing a swing-up control and a balance controller.

Specifications
Arm Regulation – $|\alpha(t)| < 30°$
Pendulum Regulation – $|\beta(t)| < 1.5°$

1.2 Background

In the classical inverted pendulum problem, a pendulum is attached to a cart and the objective is to move the cart such that the pendulum balances vertically upward. Here, the pendulum is attached to a rotary arm instead of card. This gives rise to a challenging control problem to control a highly unstable, multivariable nonlinear system. The rotary inverted pendulum system models have been used in solving stabilization problems of aircraft, missiles, etc[2].

2 Methodology

For simulation, Simscape and its Simscape Multibody toolboxes are used to model the rigid bodies associated with a rotary pendulum. A rotary pendulum consists of a pendulum pivoted to a arm/shaft which rotates horizontally. The idea is to balance the pendulum in the inverted position by applying appropriate torque to move the shaft.

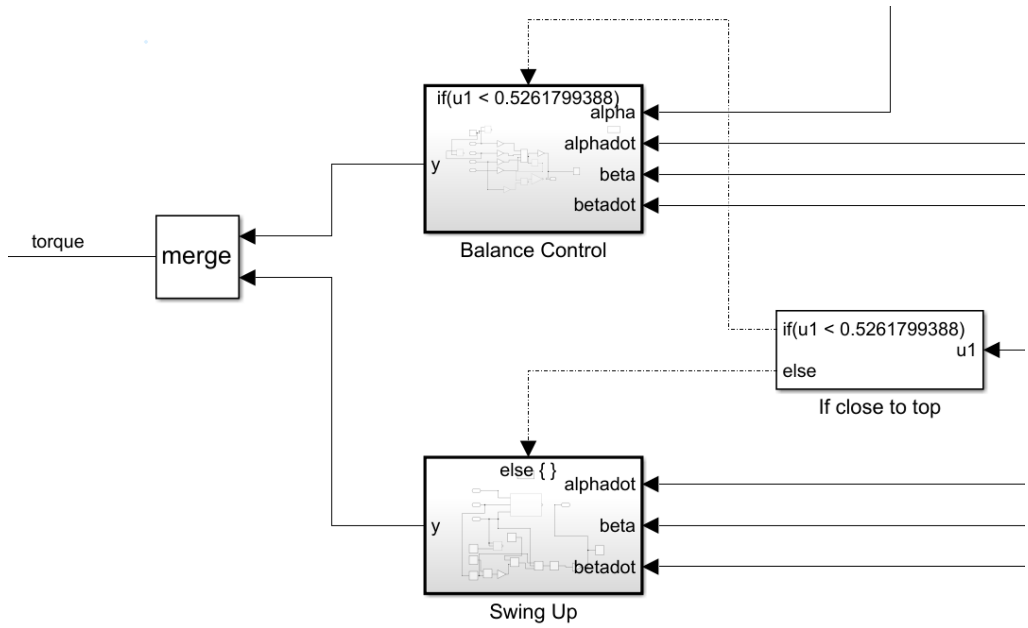

Figure 1: Switching Control

3 Physical Parameters

Name	Description	Value
M_p	Mass of the pendulum assembly (weight and link combined)	3.3e-3
M	Mass of the Rotary Arm	20.3e-3
m_a	Mass of the Rotary Arm without encoder	8.3e-3
r	Length of arm pivot to pendulum pivot	109e-3
l_p	Length of pendulum center of mass from pivot.	183.2e-3
g	Gravity	9.81
J_{eq}	Equivalent moment of inertia about motor shaft pivot axis.	3.2871e-05
J_p	Pendulum moment of inertia about its pivot axis.	1.1076e-04
K_t	Motor torque constant.	0.02797
K_m	Motor back-electromotive force constant	0.02797
R_m	Motor armature resistance.	3.30

where, the moment of inertia is calculated as follows

$$J_{eq} = \frac{m_a r^2}{3}$$
$$f = \frac{1}{T} = \frac{1}{2\pi\sqrt{lp/g}}$$
$$J_p = \frac{M_p g l_p}{4\pi^2 f^2}$$

4 Swing-up Control

$$E(\beta) = \underbrace{\frac{1}{2} J_p \left(\frac{d}{dt}\beta(t)\right)^2}_{\text{Kinetic Energy}} + \underbrace{M_p g l_p (\cos\beta(t) - 1)}_{\text{Potential Energy}} \tag{1}$$

Note the lowest potential energy point is the downward point i.e. $\beta = \pi$ is $-2M_p g l_p$
We observe rapid fluctuation in $\frac{d}{dt}\beta(t)$ hence, we avoid using Kinetic Energy term for further calculations.

$$u = \mu \text{sgn}\left(E(\beta)\frac{d}{dt}\beta(t)\cos\beta(t)\right) \tag{2}$$

$$\tau_{\text{out}} = M_{\text{arm}} u r \tag{3}$$

Input and Output The block takes pendulum angle (radians) $\beta(t)$ and pendulum velocity $\frac{d}{dt}\beta(t)$ as input and outputs Swing-up control torque τ_{out}.

4.1 Implementation

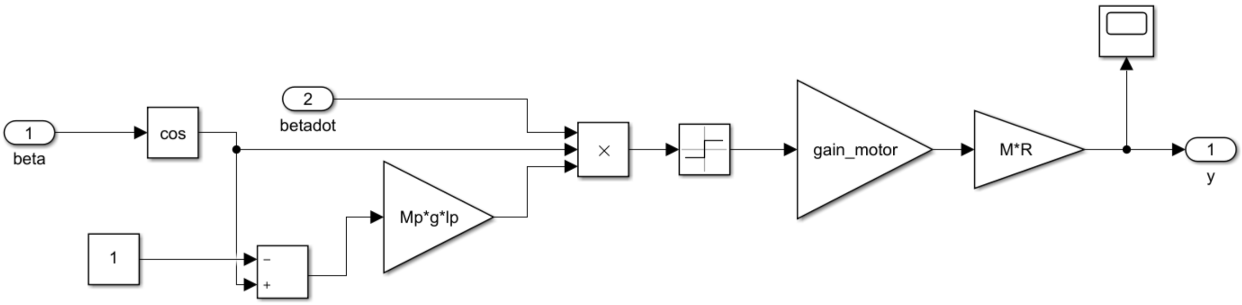

Figure 2: Swing-up Control

5 Balance Control

The linear state-space representation of the Rotary inverted pendulum is

$$\begin{aligned} \frac{d}{dt}x(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \tag{4}$$

State-Space Matrix	Expression
A	$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{rM_p^2 l_p^2 g}{J_p J_{eq} + M_p l_p^2 J_{eq} + J_p M_p r^2} & -\frac{K_t K_m (J_p + M_p l_p^2)}{(J_p J_{eq} + M_p l_p^2 J_{eq} + J_p M_p r^2) R_m} & 0 \\ 0 & -\frac{M_p l_p g (J_{eq} + M_p r^2)}{J_p J_{eq} + M_p l_p^2 J_{eq} + J_p M_p r^2} & \frac{M_p l_p K_t r K_m}{(J_p J_{eq} + M_p l_p^2 J_{eq} + J_p M_p r^2) R_m} & 0 \end{bmatrix}$
B	$\begin{bmatrix} 0 \\ 0 \\ \frac{K_t (J_p + M_p l_p^2)}{(J_p J_{eq} + M_p l_p^2 J_{eq} + J_p M_p r^2) R_m} \\ -\frac{M_p l_p K_t r}{(J_p J_{eq} + M_p l_p^2 J_{eq} + J_p M_p r^2) R_m} \end{bmatrix}$
C	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
D	$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$

Table 3 Linear State-Space Matrices

Figure 3: Source [1]

5.1 LQR

Linear Quadratic Regulator (LQR) problem is for a given plant

$$\frac{d}{dt}x(t) = Ax(t) + Bu(t) \tag{5}$$

find a control input u that minimizes the cost function

$$J = \int_0^\infty x(t)^T Qx(t) + u(t)^T Ru(t)dt \tag{6}$$

where Q is an $n \times n$ positive semidefinite weighting matrix and R is positive scalar. That is, find a control gain K in the state feedback control law

$$u = Kx \tag{7}$$

such that the quadratic cost function J is minimized.

5.2 Implementation

The state of the control system is given by

$$x = \left[\alpha, \beta, \frac{d}{dt}\alpha, \frac{d}{dt}\beta\right]$$

We use `MATLAB` [3], `lqr` function which takes A, B, Q and R as input and outputs the feedback matrix K. We get the state-feedback using LQR as `K = lqr(A,B,Q,R)`. The motor DC Voltage then can be calculated as

$$V_m = Kx \tag{8}$$

Using voltage, we can calculate the torque using following equation

$$\tau_{\text{out}} = \frac{K_t}{R_m}\left(V_m - K_m \frac{d}{dt}\alpha(t)\right) \tag{9}$$

Again, due to rapid fluctuations of the term $\frac{d}{dt}\alpha(t)$, we neglect it from the torque expression and use

$$\tau_{\text{out}} = \frac{K_t}{R_m}V_m \tag{10}$$

```
r = 109e-3;
Mp = 3.3e-3;
M = 20.3e-3;
me = 0.012;
ma = M - me;
lp = 183.2e-3;
g = 9.81;
T = 2*pi*sqrt(lp/g);
f = 1/T;
Jeq = ma*r^2/3;
Jp = 1/4*Mp*g*lp/(pi^2*f^2);
Kt = 0.02797;
Km = 0.02797;
Rm = 3.30;

den = Jp*Jeq + Mp*lp^2*Jeq + Jp*Mp*r^2;
rden = den*Rm;

a32 = r*Mp^2*lp^2*g/den;
```

```
a33 = -Kt*Km*(Jp + Mp*lp^2)/rden;

a42 = -Mp*lp*g*(Jeq + Mp*r^2)/den;

a43 = Mp*lp*Kt*r*Km/rden;

b31 = Kt*(Jp + Mp*lp^2)/rden;

b41 = -Mp*lp*Kt*r/rden;

A = [0, 0, 1, 0;
    0, 0, 0, 1;
    0, a32, a33, 0;
    0, a42, a43, 0];

B = [0; 0; b31; b41];

C = eye(4);

Q = [8, 0, 0, 0;
    0, 0.5, 0, 0;
    0, 0, 0.5, 0;
    0, 0, 0, 0.5];
R = 1;
[k,s,e]=lqr(A, B, Q, R);
k
```

6 Simulation and Modeling

6.1 Simulation Environment and Model structure

We use MATLAB and Simulink version R2020a and the Simscape, Simscape Multibody toolboxes. Here a physical model for the rotary pendulum called Plant was already provided.

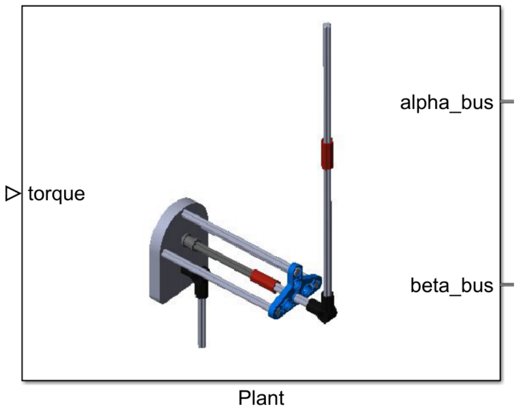

Figure 4: Rotary Pendulum Model

The overall model is divided into joints, axles, arm and pendulum. A world frame and mechanism is provided. The Simscape Solver simulates the interaction between this world and the model using the given mechanisms.

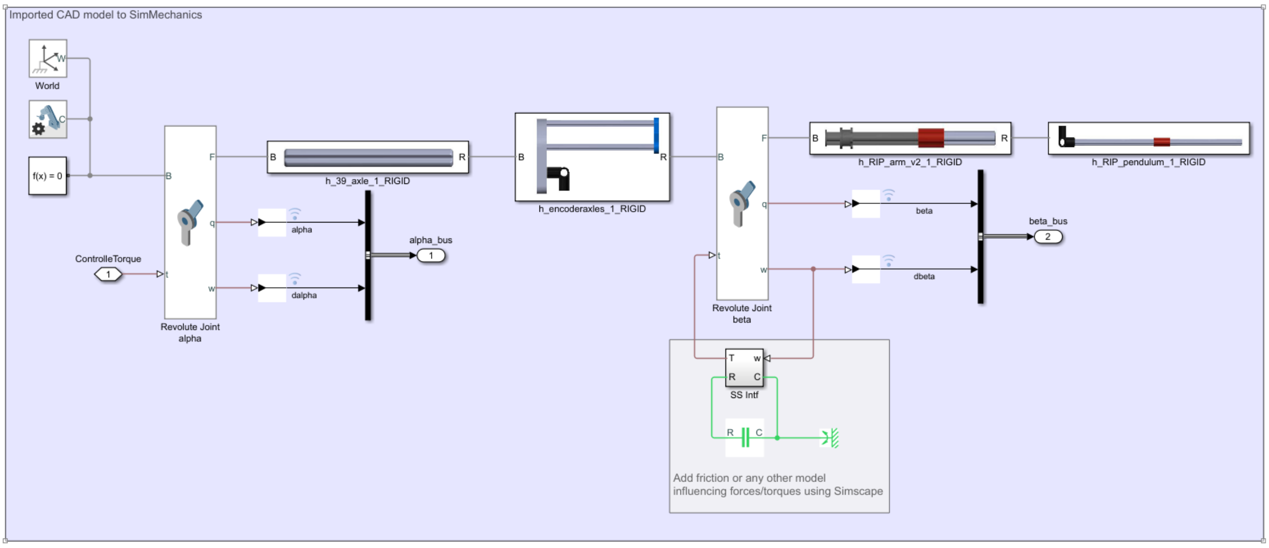

Figure 5: Simscape Multibody Model of the Plant

6.2 Simulation Results

6.2.1 Swing-up Control

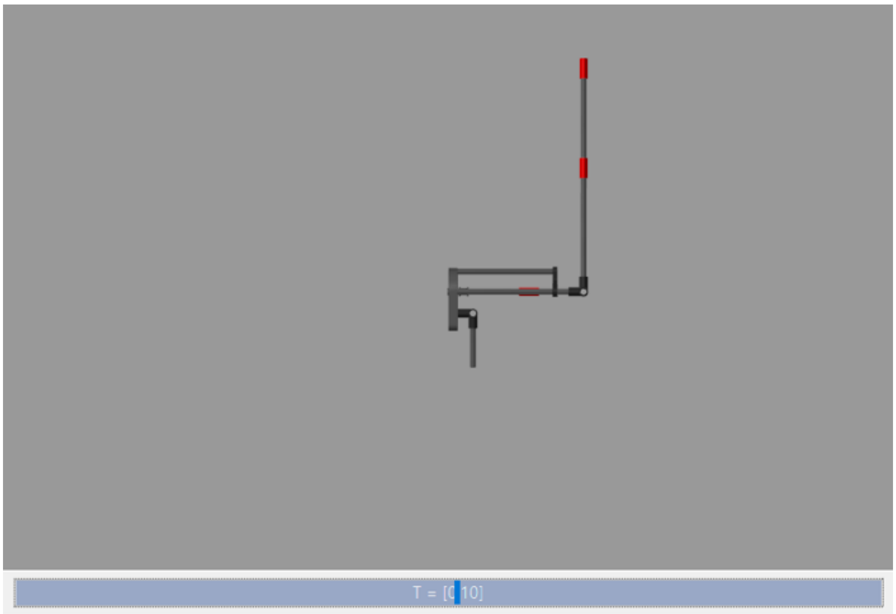

Figure 6: Swing-up

As we can see in 4, the pendulum is able to swing up near the top at around half the given time.

6.2.2 Balance Control

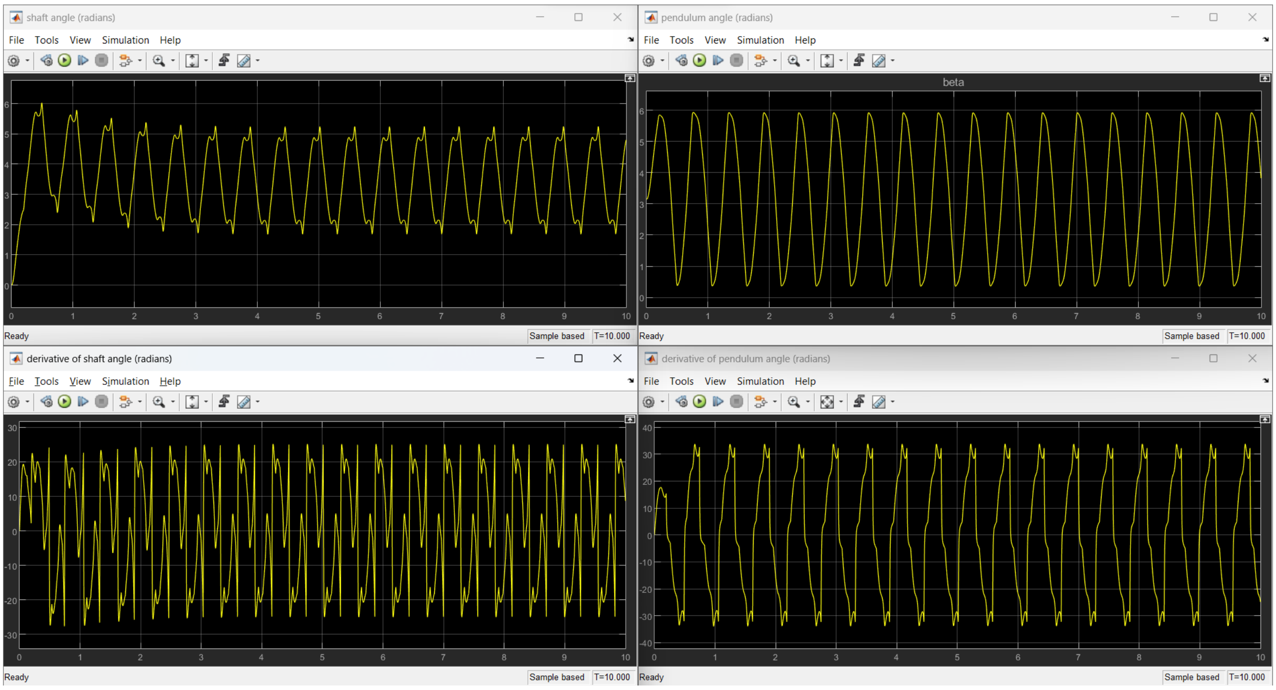

Figure 7: $k_1 = 2.8284, k_2 = -3.1409, k_3 = 0.5, k_4 = -0.5$

Here, we can see the swing up control in action, as the pendulum keeps moving towards angle 0 (or 2π). But the balancing is not strong and the pendulum goes down again.

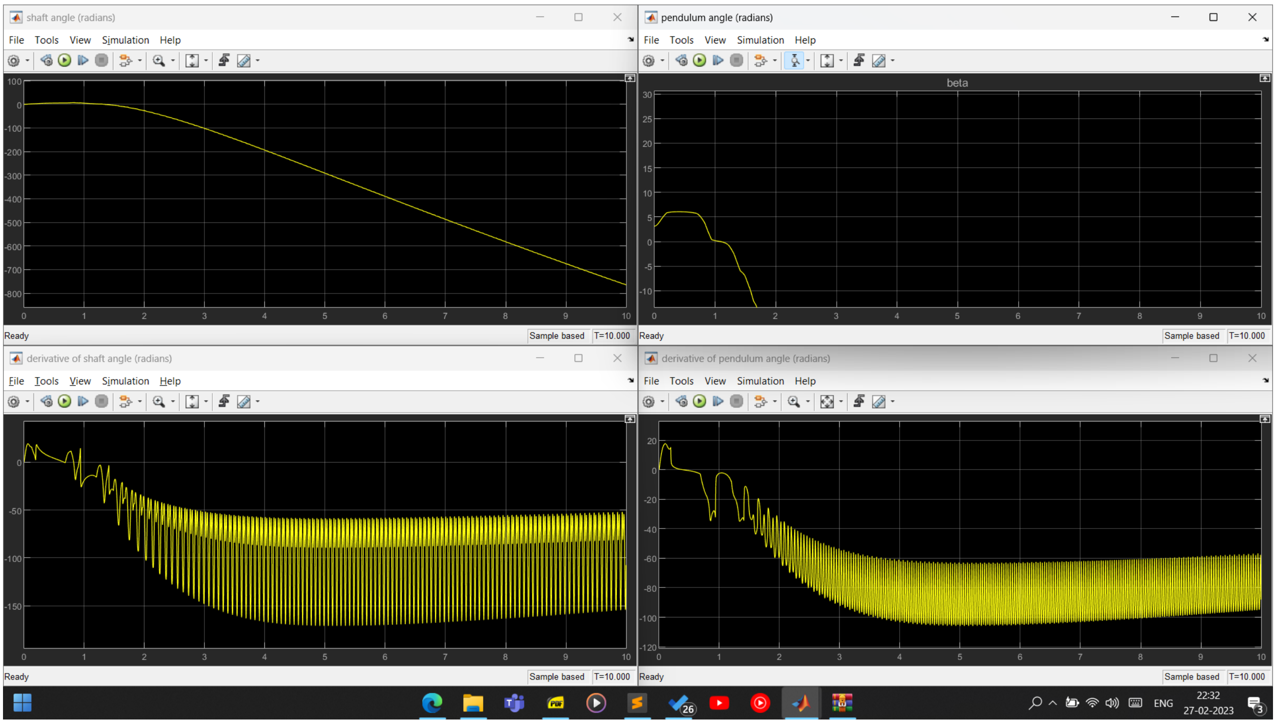

Figure 8: $k_1 = 2.8284, k_2 = -3.1409, k_3 = 0.8359, k_4 = -1$

Here, we can see for the first second, balancing was successful but after that the shaft velocity keeps decreasing which affects the LQR algorithm. Fluctuation is present due to the swing-up and LQR mechanism are still working and they take these velocities as input so a positive feedback is formed, where increasing (decreasing) velocity leads to an even more increase (decrease) in velocity.

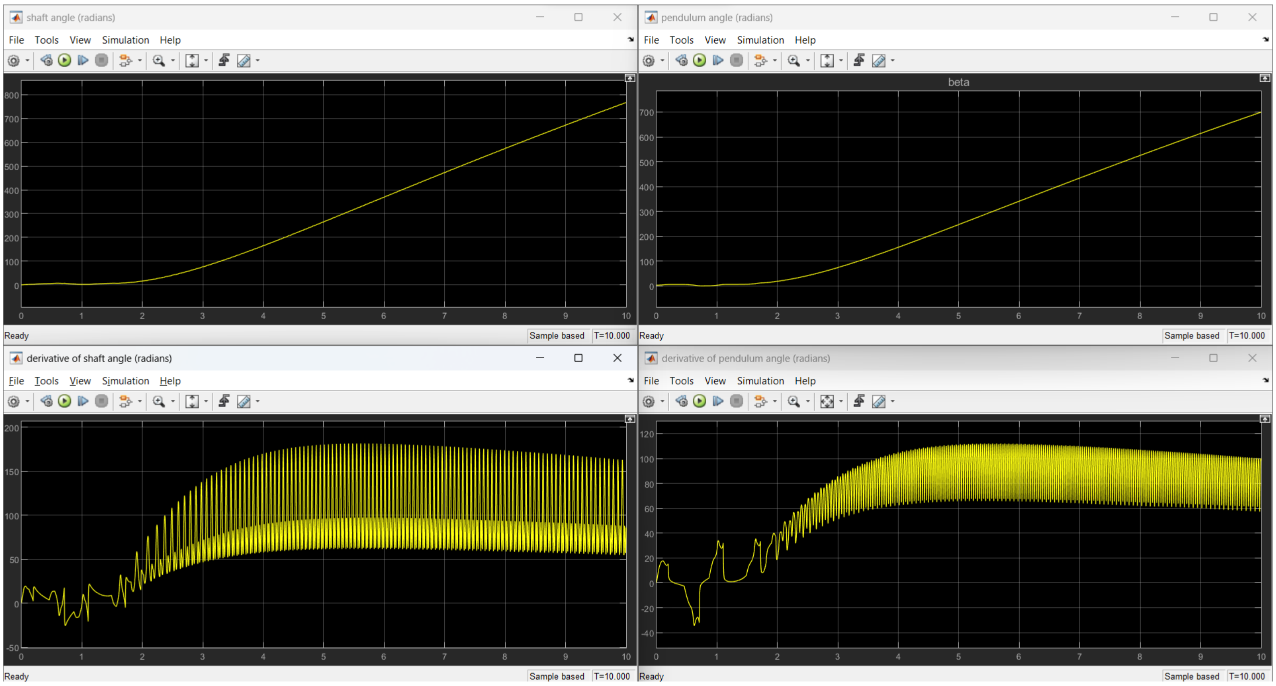

Figure 9: $k_1 = 2.8284, k_2 = -3.1409, k_3 = 0.7, k_4 = -0.5$

Here, the shaft velocity keeps increasing. Now, we notice a pattern, if $k_3 \neq -k_4$ then we will get such blow-ups. As the torque depends on $k_3 \frac{\mathrm{d}\alpha}{\mathrm{d}t} + k_4 \frac{\mathrm{d}\beta}{\mathrm{d}t}$, if $k_3 = -k_4$, then these terms will approximately cancel each other and then the torque will not depend on velocity. Hence, the positive feedback mentioned above doesn't occur.

7 Conclusion

7.1 Summary

We implemented swing-up control and LQR algorithm.
From the experimentation, we learnt parameter tuning techniques to get proper Q and R matrices.

7.2 Limitations and Future Work

More parameter tuning is required to get perfect behaviour.

Also, the current swing-up control mechanism is like a switching device as u is either 0 or $\pm\mu$. Limiting the variation in torque. To get a wide range of u values, we can take out the energy term from the sign function as shown in below equation. Now, u is either 0 or $\pm\mu E(\beta)$. Hence u takes all values from $[-\mu \max(E(\beta)), \mu \max(E(\beta))]$.

$$u = \mu E(\beta) \text{sgn}\left(\frac{d}{dt}\beta(t)\cos\beta(t)\right) \tag{11}$$

$$\tau_{\text{out}} = M_{\text{arm}} u r \tag{12}$$

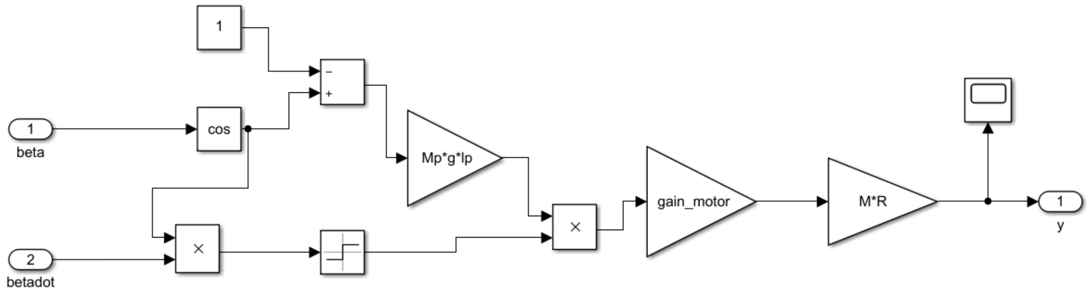

Figure 10: Updated Swing-up Control

References

[1] Quanser Inc. Qnet rotary pendulum trainer. URL: `http://eelabs.faculty.unlv.edu/docs/labs/ee370L/ee370L_07_experiment_7.pdf`.

[2] Vishwa Nath and R. Mitra. Swing-up and control of rotary inverted pendulum using pole placement with integrator. In *2014 Recent Advances in Engineering and Computational Sciences (RAECS)*, pages 1–5, 2014.

[3] Inc. The MathWorks. Linear-quadratic regulator (lqr) design. URL: `https://in.mathworks.com/help/control/ref/lti.lqr.html`.