International Journal of Engineering and Management Research e-ISSN: 2250-0758 | p-ISSN: 2394-6962
Volume-12, Issue-4 (August 2022)
www.ijemr.net https://doi.org/10.31033/ijemr.12.4.14

Advanced Control Strategies for Rotary Double Inverted Pendulum

Susheel Sriram Ananthan
Graduate, Department of Instrumentation and Control Engineering, National Institute of Technology, Tiruchirappalli, INDIA

Corresponding Author: susheel_official@yahoo.com
ORCID: 0000-0002-6950-8844

ABSTRACT

This paper presents the stabilization control of the rotational double inverted pendulum system. It is generally a non-linear, non-minimum phase, unstable, and underactuated system of high order. It has some significant real-life applications such as position control, aerospace vehicle control, and robotics. The objective is to determine the control law to the motor's output torque such that the inverted pendulum motion will be stabilized about a vertical axis and position the rotary arm to a commanded angular position. In this study, multi-PID controllers, Linear-Quadratic Regulator (LQR), and Internal Model Control (IMC) controllers are designed and MATLAB-based simulations are performed to understand and compare the performance of the three control schemes.

Keywords-- RDIP System, Linear Quadratic Regulator, PID, Internal Model Control, Non-Minimum Phase

I. INTRODUCTION

The rotary double inverted pendulum plant considered has two main components, the rotary motion plant(servo base unit) to which the rotary arm is attached, and the rotary double inverted pendulum in which a short 7-inch bottom rod, an encoder hinge, and the top 12-inch rod in such a manner that the rod is moving as an inverted pendulum in a plane that is at all times perpendicular to the rotating arm. Hence, the RDIP has 2 unactuated and 1 actuated degree of freedom. The RDIP is an extension of the Rotary Inverted Pendulum (RIP) which was invented at Tokyo Institute of Technology by Katsuhisa Furuta and his colleagues in 1992. Since the invention of RIP there have been a lot of research on controllers in combination with PID, LQR, SMC, fuzzy and neural network based. But in the case of RDIP very few controllers and results have been demonstrated. The main reason is because the RDIP is said to be easily affected by uncertainties apart from the non-linearity and instability faced by inverted pendulums. In this paper we model the Quanser SRV02 Rotary Double Inverted Pendulum system and implement various control methods such as LQR, PID and Internal model control to balance the pendulum in the upright position.

II. MATHEMATICAL MODEL OF THE PLANT

A. Model Convention

The mathematical modelling forms the basis of any control methods. The more that is known about a dynamic system, the more accurate a mathematical model can be. Accurate mathematical modelling allows the design of faster, more accurate and effective controllers. The rotary pendulum model is shown in Fig 1. The rotary arm pivot is attached to the SRV02 system(servo base unit) and is actuated. The arm has a length of L_r, 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, i.e., $V_m > 0$. The double-pendulum assembly is connected to the end of the rotary arm. The short-sized, bottom pendulum has a total length of L_{p1} and a center of mass of l_{p1}. The moment of inertia about its center of mass is J_{p1} and it has a mass of M_{p1}. The top medium-sized pendulum has a total length of L_{p2}, a center of mass of l_{p2}, a moment of inertia of J_{p2}, and a mass M_{p2}. The short bottom pendulum angle, α, and the medium top pendulum angle, ϕ, are both zero when it is perfectly upright in the vertical position and they increase positively when rotated CCW. The hinge between the two pendulums has a mass of M_h.

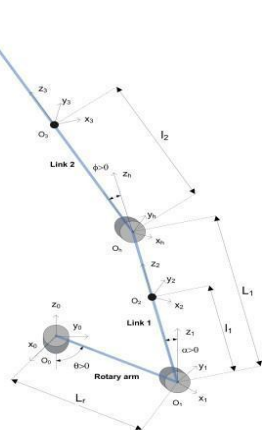


Figure 1: Rotary Double-Inverted Pendulum conventions

B. Non-Linear Equations of Motion

The Lagrange method is used to find the equations of motion of the system. The Euler-Lagrange equation is a systematic method of finding the equations of motion, i.e., EOMs, of a system. The equations that describe the motions of the rotary arm and the pendulum with respect to the servo motor voltage, i.e. the dynamics, will be obtained using the Euler-Lagrange equation:

$$\frac{\partial^2 L}{\partial t \partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = Q_i \quad (1)$$

The variables Q_i are called generalized coordinates. For this system let,

$q(t)^\top = [\theta(t) \quad \alpha(t) \quad \phi(t)]$

where θ(t) is the rotary arm angle, α(t) is the shorter bottom inverted pendulum angle and ϕ is the longer top inverted pendulum angle. The Lagrangian of the system is described as:

$$L = T - V \quad (2)$$

where T is the total kinetic energy of the system and V is the total potential energy of the system. Thus the Lagrangian is the difference between a system's kinetic and potential energies. The generalized forces Q_i are used to describe the non-conservative forces (e.g., friction) applied to a system with respect to the generalized coordinates. In this case, the generalized force acting on the rotary arm is:

$$Q_1 = \tau - D_r \dot{\theta} \quad (3)$$

and acting on the bottom and top pendulum are:

$$Q_2 = -D_{p1}\dot{\alpha} \quad (4)$$

$$Q_3 = -D_{p2}\dot{\phi} \quad (5)$$

Our control variable is the input servo motor voltage, Vm. Opposing the applied torque is the viscous friction torque, or viscous damping, corresponding to the term Dr. Since the pendulum is not actuated, the only force acting on the link is the damping. The viscous damping coefficient of the short (bottom) and medium (top) pendulums are denoted by Dp1 and Dp2.

The total kinetic energy for the whole system is given by the combination of kinetic energy of each link in our system. Hence, we will have the total kinetic energy as:

$$K = \frac{1}{2}J_1\dot{\theta}_1^2 + \frac{1}{2}J_2\dot{\theta}_2^2 + \frac{1}{2}J_3\dot{\theta}_3^2 + \frac{1}{2}m_2\left[\left(L_1\dot{\theta}_1 + l_2\dot{\theta}_2\cos\theta_2\right)^2 + \left(-l_2\dot{\theta}_2\sin\theta_2\right)^2\right] +$$

$$\frac{1}{2}m_3\left[\left(L_1\dot{\theta}_1 + l_2\dot{\theta}_2\cos\theta_2 + l_3\dot{\theta}_3\cos\theta_3\right)^2 + \left(-l_2\dot{\theta}_2\sin\theta_2 - l_3\dot{\theta}_3\sin\theta_3\right)^2\right] \quad (6)$$

and tl

$$P = m_2 g l_2 cos(\theta_2) + m_3 g(L_2 cos(\theta_2) + l_3 cos(\theta_3)) \quad (7)$$

Where,

$J_1 = J_r$; $\theta_1 = \theta$; $L_1 = L_r$; $m_2 = M_{p1}$

$J_2 = J_{p1}$; $\theta_2 = \alpha$; $L_2 = L_{p1}$; $m_3 = M_{p2}$

$J_3 = J_{p2}$; $\theta_3 = \phi$; $L_3 = L_{p2}$; l_i = Distance from centre of rotation to link i(=1,2,3)

The torque applied at the base of the rotary arm (i.e., at the load gear) is generated by the servo motor as described by the equation,

$$\tau = \frac{\eta_g K_g \eta_m k_t (V_m - K_g k_m \dot{\theta})}{R_m} \quad (8)$$

Further nonlinear equations of motion are obtained from equation 1, 2 & 3. Then for Linearization, Initial conditions for all the variables are taken zero. θ = 0◦, α = 0◦, ϕ = 0◦, $\dot{\theta} = 0$, $\dot{\alpha} = 0$, $\dot{\phi} = 0$◦. For a multivariable system control variable z is defined:

$$z^{\top} = [z_1 \ z_2]$$

The linearized function is:

$$f_{lin} = f(z_0) + \left(\frac{\partial f(z)}{\partial z_1}\right)\Big|_{z=z_0} (z_1 - a) + \left(\frac{\partial f(z)}{\partial z_2}\right)\Big|_{z=z_0} (z_2 - b) \quad (9)$$

C. Linear State-Space Model

The linear state-space equations is given by:

$$\dot{x} = Ax + B$$
$$y = Cx + Du \quad (10)$$

where x is the state, u is the control input, A, B, C, and D are state-space matrices. For the rotary pendulum system, the state and output are defined as:

$$x^T = [\theta \ \alpha \ \varphi \ \dot{\theta} \ \dot{\alpha} \ \dot{\varphi}]$$
$$y^T = [x_1 \ x_2 \ x_3]$$

where, θ(t) is the position of rotary arm angle and α(t) is the position of the smaller arm of pendulum and ϕ is the position of second arm of the pendulum as shown in the Fig. 1. On linearizing the nonlinear equations of motion of the system which is obtained from Lagrange equation, solving for the acceleration terms and substituting the state given in we obtain the following state-space matrices:

$$A = \frac{1}{J_T}\begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & a_{42} & -M_{p1} l_{p1} L_r M_{p2} g(-l_{p1} + L_{p1}) & 0 & 0 & 0 \\ 0 & a_{52} & -M_{p2} g(-L_r^2 L_{p1} M_{p1} + L_{p1} L_r^2 M_{p1} + J_r L_{p1}) & 0 & 0 & 0 \\ 0 & a_{62} & a_{63} & 0 & 0 & 0 \end{bmatrix}$$

Where,

$$a_{42} = L_r g(M_{p1}^2 l_{p1}^2 + 2M_{p1} l_{p1} M_h L_{p1} M_h L_{p1}^2 M_{p2} M_{p1} l_{p1}^2 M_{p2} M_h^2 L_{p1}^2)$$

$$a_{52} = g(L_r^2 L_{p1} M_{p1} M_{p2} + L_{p1} L_r^2 M_h M_{p2} + L_{p1} L_r^2 M_h^2 + L_r^2 L_{p1} M_{p1}^2 + J_r L_{p1} M_h + J_r L_{p1} M_{p1} + L_{p1} L_r^2 M_h M_{p1} + L_r^2 L_{p1} M_h M_{p1})$$

$$a_{62} = -\frac{g}{L_{p2}}\Big(L_{p1} L_r^2 L_{p1} M_{p1}^2 + L_{p1} L_r^2 L_{p2} M_h M_{p1} + L_r^2 L_{p1} L_{p2} M_h M_{p1} + L_{p1} L_r^2 L_{p2} M_h^2 + L_r^2 L_{p1} L_{p2} M_{p1}^2 - M_{p1} L_{p1}^2 L_r^2 M_{p2} + L_r^2 L_{p1} L_{p2} M_{p1} M_{p2} + L_{p1} L_r^2 L_{p2} M_h M_{p2} + J_r L_{p1} L_{p2} M_h + J_r L_{p1} L_{p2} M_{p1} - M_{p1} L_{p1}^2 J_r + L_{p1} J_r L_{p1} M_{p1} - M_{p1} L_{p1}^2 M_h L_r^2 + L_r^2 L_{p1} M_{p2} M_{p1} L_{p1} + M_{p1} L_{p1} L_r^2 M_h L_{p1} - L_r^2 M_{p1}^2 L_{p1}^2\Big)$$

$$a_{63} = \frac{g}{L_{p2}}\Big(J_r L_{p1} L_{p2} M_{p2} + J_r L_{p1}^2 M_{p2} + L_{p1}^2 L_r^2 M_{p1} M_{p2} + M_{p1} L_{p1}^2 L_r^2 M_{p2} + L_{p1} L_r^2 L_{p2} M_{p1} M_{p2} - L_r^2 L_{p1} L_{p2} M_{p1} M_{p2} + M_{p1} L_{p1}^2 J_r + M_h L_{p1}^2 J_r + M_h L_{p1}^2 M_{p1} L_r^2 + M_{p1} L_{p1}^2 M_h L_r^2 - 2L_r^2 L_{p1} M_{p2} M_{p1} L_{p1} - 2M_{p1} L_{p1} L_r^2 M_h L_{p1}\Big)$$

$$B = \frac{1}{J_T}\begin{bmatrix} 0 \\ 0 \\ 0 \\ M_h L_{p1}^2 + M_{p1} L_{p1}^2 \\ L_r(M_h L_{p1} + M_{p1} L_{p1}) \\ -\frac{L_r}{L_{p2}}(-M_{p1} L_{p1}^2 + M_{p1} L_{p1} L_{p2} + M_h L_{p1} L_{p2} + M_{p1} L_{p1} L_{p1}) \end{bmatrix}$$

In the output equation, only the position and velocity of the servo and link angles are being measured. Based on this, the C and D matrices in the output equation are:

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$D = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

On substitution of the values from the Quanser manual, the state space model obtained is:

$$dx/dt = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 291.98 & -15.89 & -18.56 & -0.75 & 1 \\ 0 & 257.22 & -28.92 & -14.15 & -0.7 & 1.15 \\ 0 & -340.87 & 127.44 & 18.75 & 1.15 & -3.11 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 85.1 \\ 64.87 \\ -85.97 \end{bmatrix} u(t) \quad (11)$$

$$y(t) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} u(t) \quad (12)$$

To comprehend the system, the root locus of the model is simulated for which the response obtained is as shown in Fig. 2 and the open loop step response of the system is obtained as shown below in Fig. 3:

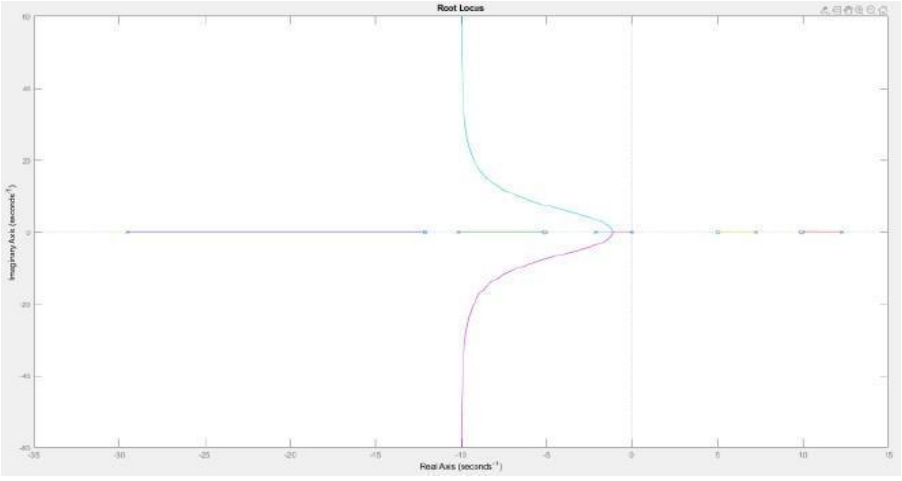
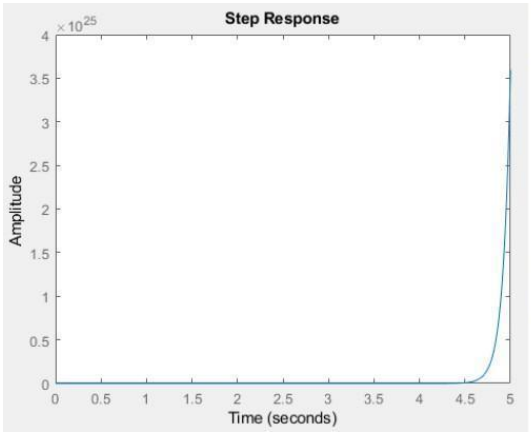


Figure 2: Root locus of the system




```
>> pole(Gservo)

ans =

         0
  -29.5581
   12.2802
  -10.1924
    7.2381
   -2.1379

>> zero(Gservo)

ans =

  -10.2666
    8.6800
    0.0034
    0.0000
```

Figure 3: Step response of rotary arm

Therefore, from the presence of poles on the origin and right-hand side of the imaginary axis it is understood that there is an output component that increases exponentially without limits thereby making the system highly unstable.

III. CONTROL USING LINEAR QUADRATIC REGULATOR

In general, the theory of optimal control is concerned with operating a dynamic system at minimum cost. When the system dynamics are described by a set of linear differential equations and the cost is described by a quadratic function, it is referred to as a Linear Quadratic(LQ) problem. One of the main results in the theory is that the solution is provided by the linear–quadratic regulator(LQR), a feedback controller whose equations are explained in the upcoming sub-section. The LQR is an important part of the solution to the LQG (Linear–Quadratic–Gaussian) problem. Like the LQR problem itself, the LQG problem is one of the most fundamental problems in control theory. The LQR algorithm reduces the amount of work done by the control systems engineer to optimize the controller. The LQR algorithm is essentially an automated way of finding an appropriate state-feedback controller.

D. LQR Optimization Method

A system can be expressed in state variable form as:

$$\dot{x} = Ax + Bu \tag{13}$$

with x(t) ∈ R_n, u(t) ∈ R_m. The initial condition is x(0). We assume here that all the states are measurable and seek to find a State-Variable Feedback(SVFB) control that gives desirable closed-loop properties:

$$U = -Kx + v \tag{14}$$

The closed-loop system using this control becomes:

$$\dot{x} = (A - B.K)x + Bv = A_c x + Bv$$

(15)

with A_c the closed-loop plant matrix and v(t) the new command input.

Note that the output matrices C and D are not used in SVFB design. To design a SVFB that is optimal, we may define the performance index (PI):

$$J = \frac{1}{2}\int_0^\infty x^T Qx + u^T Ru \; dt \tag{16}$$

Substituting the SVFB control into this yields

$$J = \frac{1}{2}\int_0^\infty x^T Qx + u^T Ru \; dt \tag{17}$$

We assume that input v(t) is equal to zero since our only concern here are the internal stability properties of the closed-loop system. The objective in optimal design is to select the SVFB K that minimizes the performance index J. The performance index J can be interpreted as an energy function, so that making it small keeps the total energy of the closed-loop system small. Note that both the state x(t) and the control input u(t) are weighted in J, so that if J is small,then neither x(t) nor u(t) can be too large. Note that if J is minimized, then it is certainly finite, and since it is an infinite integral of x(t), this implies that x(t) goes to zero as t goes to infinity. This in turn guarantees that the closed loop system will be stable. Linear Quadratic Regulator is an

optimal controller used to achieve desired target value with minimum control effort and time. In MATLAB, the command [K] = lqr (A,B, Q, R) calculates the optimal feedback matrix K such that it minimizes the cost function subject to the constraint defined by the state equation. The response of a system for a different set of state feedback gain matrices is determined by varying Q values, BY keeping R=10, and choosing the one which gives the best performance. The K matrix obtained from Matlab is:

$$K = [0.8049 - 15.3019 - 43.08420.6508 - 5.6899 - 4.3549]$$

E. Simulation

In this section, SIMULINK block diagram as shown in Figure 4, was used to simulate the closed-loop control of the Rotary Double Inverted Pendulum system. The system is simulated using the linear model of RDIP with a square wave signal with an amplitude of 30Deg(0.5236Rad) as input. The SIMULINK model uses the state feedback control for which the feedback gain K is calculated from the LQR function. The goal is to ensure that the SVBF gain calculated successfully stabilizes the system (i.e., keeps it inverted), tracks the reference servo position, and does not saturate the 10V DC motor.

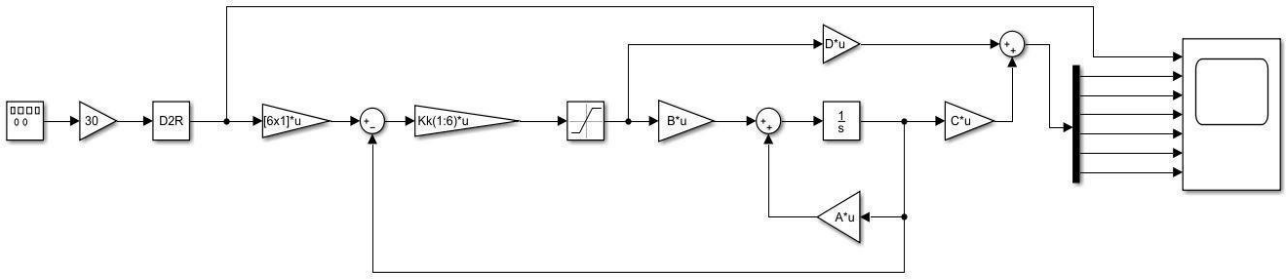


Figure 4: Simulink model used to simulate RDIP using LQR Controller

The simulation of the above control system is performed and the following responses are obtained.

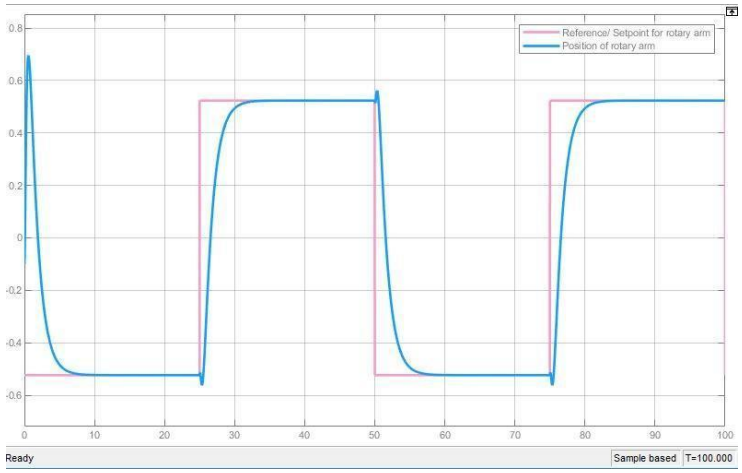


Figure 5: Simulated response for position of servo rotary arm using LQR (θ)

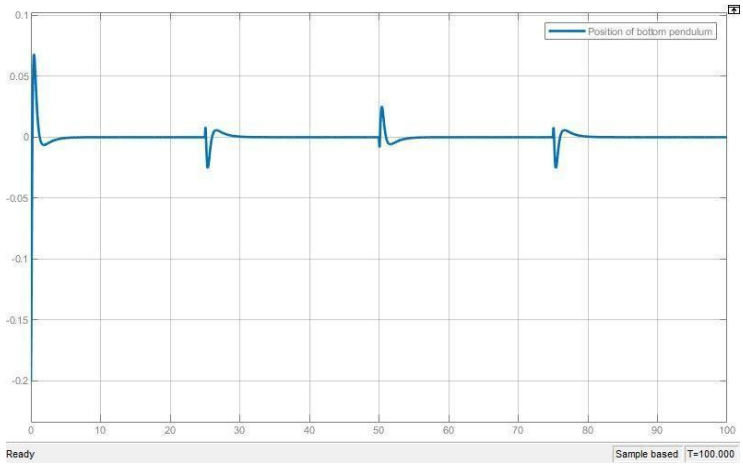


Figure 6: Simulated response for position of bottom pendulum using LQR (α)

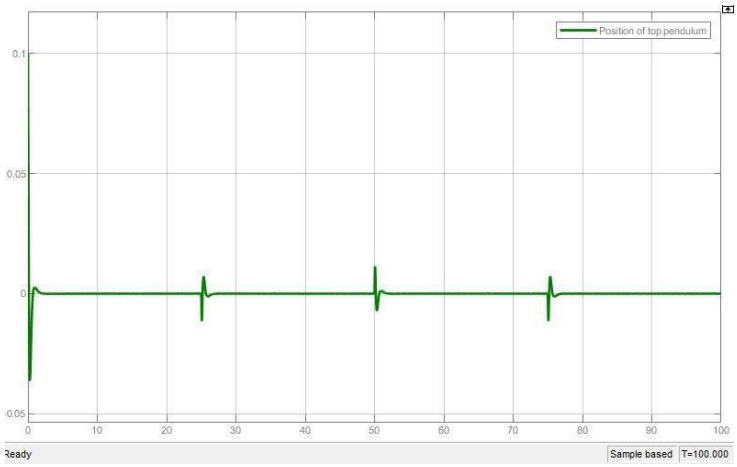


Figure 7: Simulated response for position of top pendulum using LQR (φ)

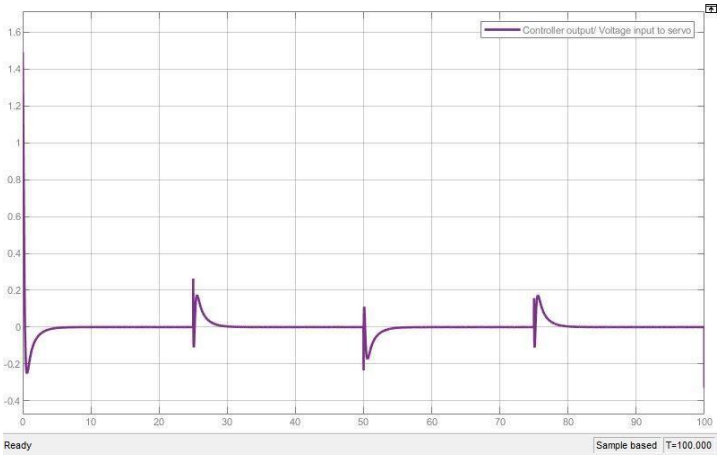


Figure 8: Controller Output (Input voltage to Servo)

As shown by the response in Fig 5, 6, 7 & 8, the pendulum maintains its balance about the upright vertical position while tracking the ±30 degree servo angle.

IV. PID BASED CONTROL

A PID controller is a control loop feedback mechanism widely used in industrial control systems. A PID controller continuously calculates an error value as the difference between a desired reference and a measured process variable and appliesa correction based on its proportional, integral and derivative terms. The controller attempts to minimize the error over time by adjustment of a control variable to a new value determined by a weighted sum of the control terms. A simple example is the cruise control function of a vehicle,where the engine power will be increased or decreased during an uphill or a downhill run to ensure that the vehicle moves at constant speed. A schematic of a standard PID controller can be found in Fig 9.

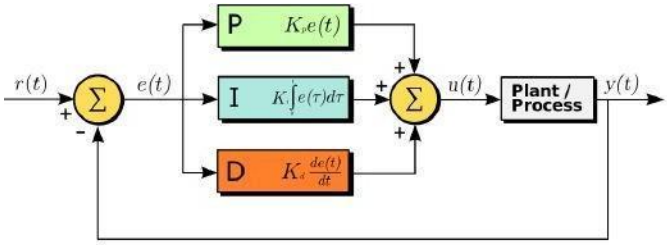


Figure 9: PID loop schematic

The response of the proportional term is proportional to the error and if there is no error, there is no proportional response. The integral term accounts for past values and integrates them over time. When the error becomes zero, the integral term will cease to grow. The derivative term is an estimate of the future trend of the error based on its current rate of change. This means that the derivative term alone cannot bring the error down to zero. This is why it is also called anticipatory control. Implementations of controllers that include a derivative action tend to also include a low-pass filtering for the derivative term to limit high-frequency gain and noise.

A. PID Tuning

Generally, PID controllers are used to control the SISO system. However, in the proposed system there are three parameters: position of servo arm, the angle of the pendulum 1 and the angle of the pendulum 2 to be controlled which is basically a SIMO system. To solve this problem, the triple PID controller[5] is used in this project to control the rotary double inverted pendulum using a full state feedback controller[1]. The control design is built in Matlab's SIMULINK environment as shown in Fig 10.

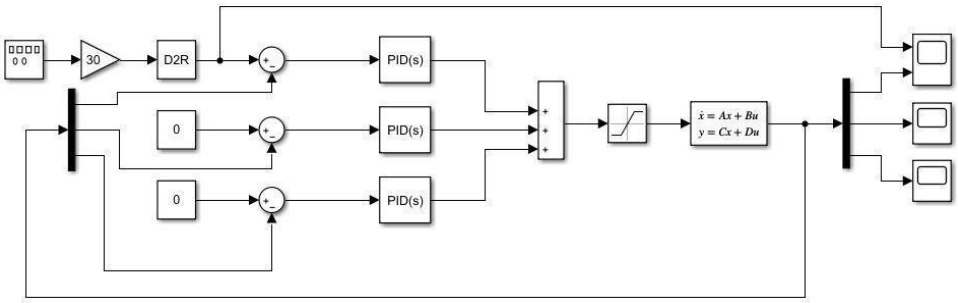


Figure 10: Simulink model for control of RDIP system using PID Controller

In Fig 10, the PID controller 1 is to control the position of the servo arm, and the value of bottom pendulum and the top pendulum is controlled by controller 2 and controller 3, respectively.

Full state feedback (FSF) method, or pole placement method[2], is a method employed in feedback control system theory to place the closed-loop poles of a plant in predetermined locations in the s-plane. Placing poles is desirable because the location of the poles corresponds directly to the eigenvalues of the system, which control the characteristics of the response of the system. In the pole placement technique, if the system is completely state controllable and all state variables are available for feedback, then poles of the closed-loop system may be placed at any desired locations by means of state feedback through an appropriate state feedback gain matrix K.

Full state feedback is utilized by commanding the input vector u. Consider an input proportional (in the matrix sense) to the state vector, System with state feedback (closed-loop):

$$u = -Kx \quad (18)$$

Substituting into the state space equations above, we have

$$\dot{x} = (A - BK)x \quad (19)$$

$$y = (C - DK)x \quad (20)$$

The poles of the FSF system are given by the characteristic equation of the matrix A − BK, det[SI − (A−BK)] = 0. Comparing the terms of this equation with those of the desired characteristic equation yields the values of the feedback matrix K (the PID gains K_p, K_i and K_d) which force the closed-loop eigenvalues to the pole locations specified by the desired characteristic equation. The state feedback controller can be used only if the system is controllable and it tracks the input signal or improves damping of the system. For a given system, the state feedback gain matrix K is not the same but depends on the desired closed loop pole location. This will also determine the speed and damping of the response.

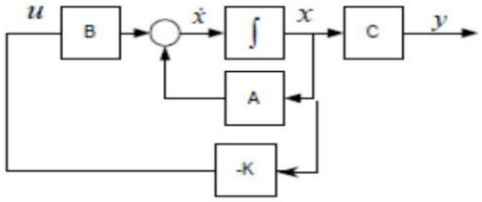


Figure 11: Block Diagram of Pole-placement Controller

The value of the obtained control parameters of the PID controllers using Ackermann's formula through Matlab command are shown in Table 1 as follows:

Table 1: Parameters of the 3 PID controllers

PARAMETERS	CONTROLLER 1	CONTROLLER 2	CONTROLLER 3
KP	0.8049	-15.3014	-43.0837
KI	0.3162	0	0
KD	0.6508	-5.6898	-4.3549

B. Simulation

The response of the linear double inverted pendulum plant controlled by the multi-PID controller is shown as below. Based on the control results, the double inverted pendulum system is stabilized by the multi-PID controller and the system, tracks the reference servo position, and does not saturate the DC motor.

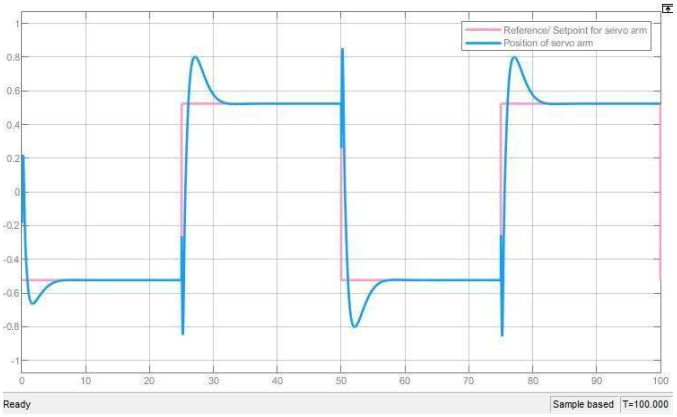


Figure 11: Simulated response for position of servo arm using PID (θ)

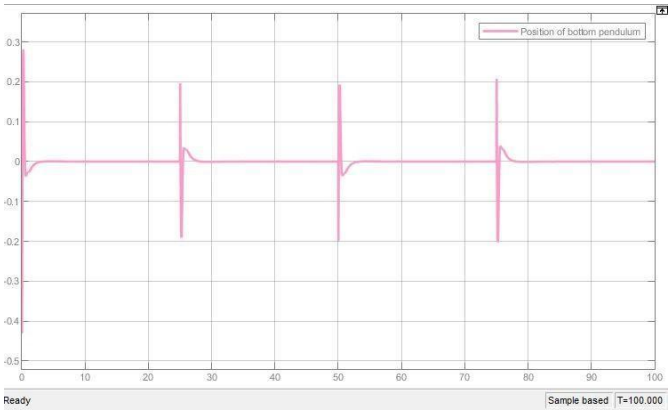


Figure 12: Simulated response for position of bottom pendulum using PID (α)

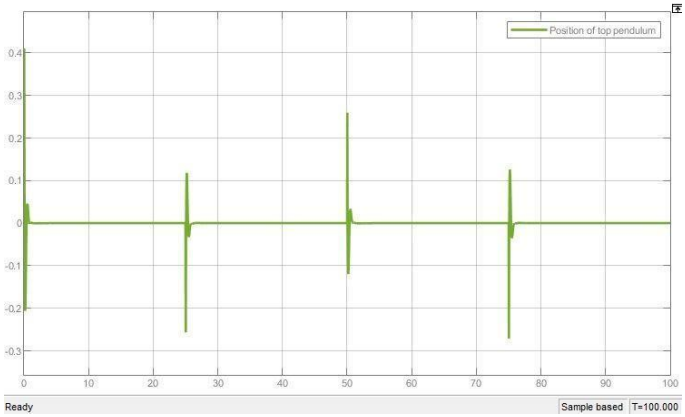


Figure 13: Simulated response for position of top pendulum using PID (φ)

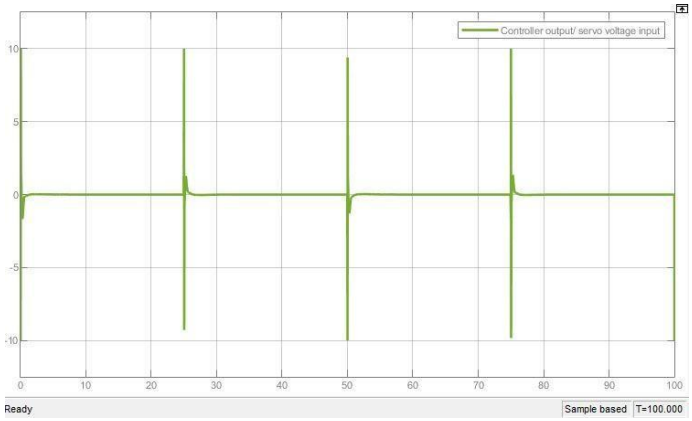


Figure 14: Controller Output (Input voltage to Servo)

As shown by the responses in Figure 11, 12, 13 & 14 the pendulum maintains its balance about the upright vertical position while tracking the 30 degree servo angle.

V. INTERNAL MODEL CONTROL

Every feedback controller is designed by employing some form of a model for the process that is to be controlled and/or the dynamics of the exogenous signal affecting the process. Consequently, the term "model-based" is often used here. 30- years ago, a new model-based controller design algorithm named "Internal Model Control" (IMC) has been presented by Garcia and Morari, which was developed upon the internal model principle to combine the process model

and external signal dynamics. In the control theorem, the control systems design is fundamentally determined by the steady state and dynamic behavior of the process to be controlled. It is an important issue to know the way in which the process characteristics influence the controller structure. The internal model control (IMC) viewpoint appeared as an alternative to traditional feedback control algorithms, which link the process model with the controller structure.

The theory of IMC states that "control can be achieved only if the control system encapsulates, either implicitly or explicitly, some representation of the process to be controlled". The main objective is to design an IMC Controller for the proposed pendulum system to reduce the effect of disturbance due to mismatching in modeling. A schematic of a block diagram of Internal Model Control can be found in Fig 15.

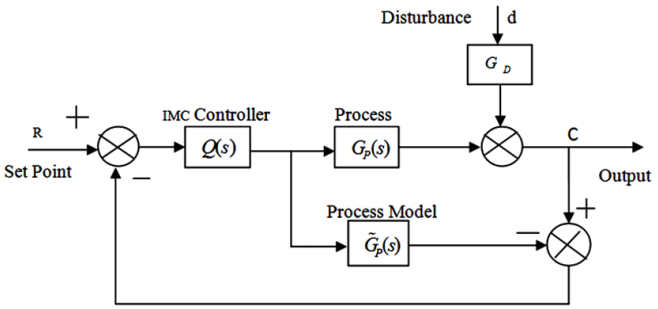


Figure 15: Block Diagram of Internal Model Control

where, $G_p(s)$ represents the process itself. $G_d(s)$ the process transfer functions of the disturbance, $\tilde{G}_p(s)$ the mathematical model (transfer function) of the process, and Q(s) the transfer function of the IMC controller. As may be observed from the block diagram of the IMC structure, there are two parallel paths starting from the manipulated variables u(s): one passes through the real process Gp(s) and the other passes through the model process $\tilde{G}_p(s)$. The role of the parallel containing the model $\tilde{G}_p(s)$ is to make possible the generation of the difference between the actual process output y(t) and an estimation prediction of the manipulated variable effect on the process output. Assuming that the process model is a perfect representation of the real process that is $\tilde{G}_p(s) = G_p(s)$.

A. IMC Design Procedure

The characteristic roots of the open-loop system are located at:

[0 -29.5581 12.2802 7.2381 -10.1924 -2.1379];

Therefore, the system is unstable. To stabilize it, K_s is used to relocate the poles of the inner-loop at:
[-4+2.3946j; -4-2.3946j; 2*([-4+2.3946j; -4-2.3946j]); 4*([-4+2.3946j; -4-2.3946j])].
Using pole placement method, the static state feedback gain vector, K_s is computed as:

$$K_s = [2.5003 \quad -23.3135 \quad -40.7481 \quad 4.4464 \quad -7.0241 \quad -4.2596]$$

Now, the new transfer function is obtained as follows:

$$G_{IL} = \frac{\theta(s)}{v(s)} = \frac{85.1s^4 + 189.6s^3 - 1.242e04s^2 - 5332s + 2.628e05}{s^6 + 56s^5 + 1352s^4 + 1.696e04s^3 + 1.176e05s^2 + 4.232e05s + 6.571e05} \quad (21)$$

where v(s) is the control input and θ(s) is the output, the angle of the rotary arm needs to be controlled. To avoid the complexity, SVD (singular value decomposition) based method (singular perturbation approximation) is used to reduce the order of the equation[3]. The reduced model thus obtained is given by:

$$\tilde{G}_{min} = \frac{0.2501s^2 - 2.766s + 8.445}{s^2 + 6.618s + 21.11} \quad (22)$$

Now, $\tilde{G}_{min}$ is decomposed into $\tilde{G}^{+}_{min}$ and $\tilde{G}^{-}_{min}$ due to its non-minimum phase, where

$$\tilde{G}^{-}_{min} = \frac{0.2501}{s^2 + 6.618s + 21.11} \quad (23)$$

$$\tilde{G}^{+}_{min} = 0.2501s^2 - 11.0593s + 33.766 \quad (24)$$

To improve robustness, the effects of mismatch between the process, and process model should be minimized. Since the differences between process and the process model usually occur at the systems high

frequency response end, a low-pass filter f(s) is usually added to attenuate this effect. Thus, IMC is designed using the inverse of the process model in series with a low-pass filter[4]. A common filter choice that conforms to this requirement is:

$$f(s) = \frac{1}{(\lambda s + 1)^n} \tag{25}$$

where, the filter order n is selected large enough to make $\tilde{G}^{-}_{min}$ proper. In addition to this criterion, the filter time constant λ must satisfy;

$$\lambda \geq \left(\lim_{s \to -\infty} \frac{D(s)N(s)}{20 s^n N(s) D(0)} \right)^{1/n} \tag{26}$$

Therefore, the suitably selected value of λ based on equation 26 is equal to 0.5. Final IMC controller on augmenting with a filter f(s) is given by,

$$Q(s) = (\tilde{G}^{-}_{min})^{-1}(s).f(s) \tag{27}$$

Hence, the obtained transfer function of Q(s) is given by:

$$Q(s) = \frac{s^2 + 6.618s + 21.11}{2.1s^2 + 8.45s + 8.45} \tag{28}$$

B. Simulation

The block diagram of the IMC control design is built in Matlab's Simulink environment, as shown in Fig 16.

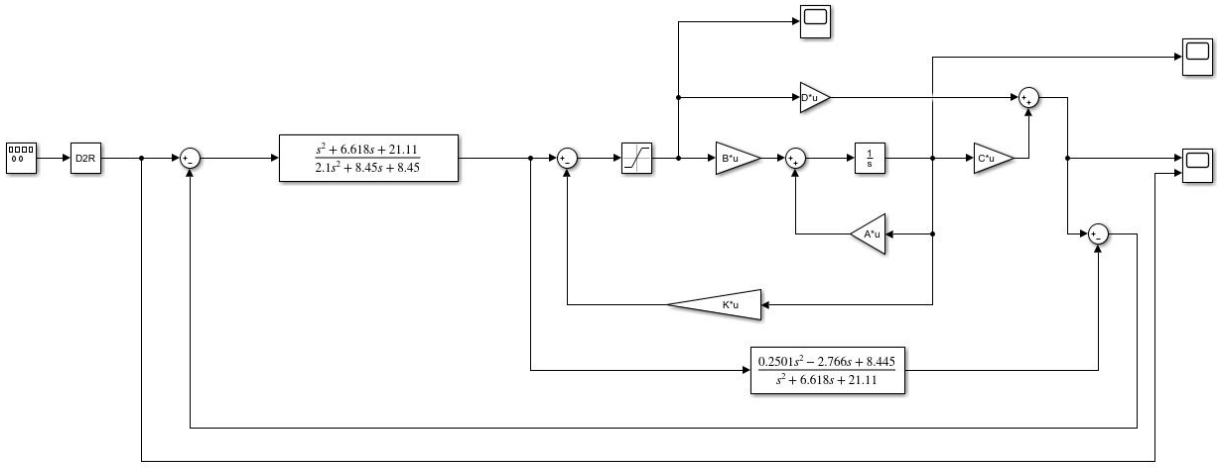


Figure 16: Simulink model for control of RDIP system using IMC

For the performance evaluation of the designed controller, a set point tracking is observed between t = 0s and t = 100s. The controller results of the linear double inverted pendulum are shown as below. According to the control results, the double inverted pendulum system is successfully stabilized by the IMC controller and the system, tracks the reference servo position, and does not saturate the dc motor.

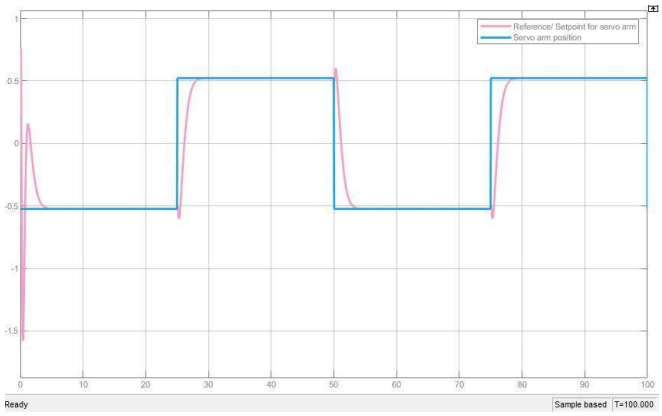


Figure 17: Simulated response for position of rotary arm using IMC (θ)

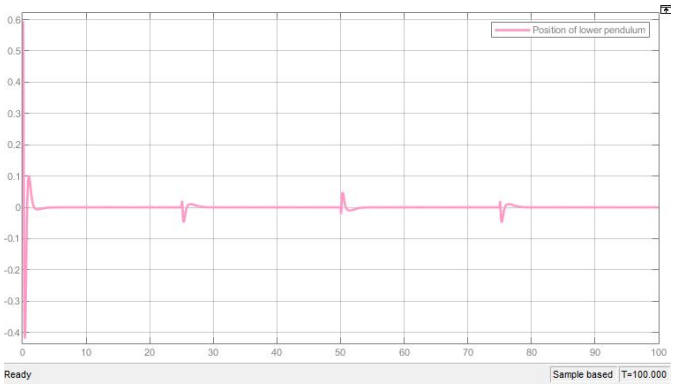


Figure 18: Simulated response for position of bottom pendulum using IMC (α)

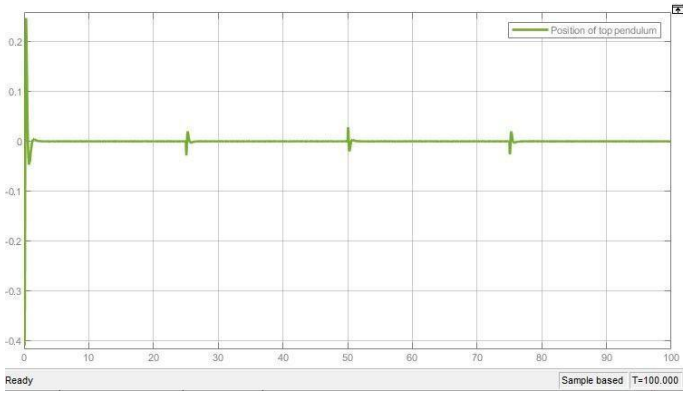


Figure 19: Simulated response for position of top pendulum using IMC (Φ)

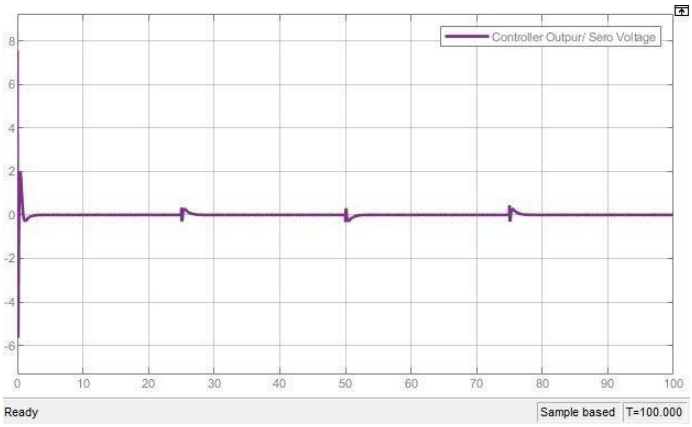


Figure 20: Controller Output (V_{in} to Servo)

As shown by the responses in Figure 17, 18, 19 & 20, the pendulum maintains its balance about the upright vertical position while tracking the ±30 degree servo angle.

VI. CONCLUSION

The Rotary Double Inverted Pendulum (RDIP) system is a classic problem in control theory. It is used in many applications. It is inherently an unstable but completely controllable system using various controllers available which can be used to hold the pendulum at the upright position. The controllers should maintain the bottom and top pendulum angle (α and φ) at 0^o to the vertical axis and the rotary arm is maintained at for its reference.

From Table 2, we see that the LQR controller is seen to produce comparatively more sluggish response and takes around 5 seconds to stabilize the pendulum. The IMC controller has a faster response compared to LQR and PID controllers. The PID controller has faster response than LQR Controller but a greater overshoot is observed compared to IMC and LQR controller. Overall, considering all the parameters we can say that Internal Model Control based controllers are more preferred which shows best results.

Table 2: Controller comparison based time domain specification

CONTROLLER	RISE TIME	SETTLING TIME	OVERSHOOT	STEADY STATE ERROR
LQR CONTROLLER	2.7463	5.3726	3.9089x10−04	-4.601x10−7
PID CONTROLLER	0.7670	4.9668	26.8016	-3.227x10−7
IMC	1.6017	3.0190	0.0158	3.132x10−7
NN CONTROLLER	2.9060	5.6467	2.8644x104	-2.731x10−7

The figure given below shows a clear comparison of the three controllers (from top-bottom): Linear Quadratic Regulator (LQR) based State Feedback Controller, PID controller and Internal Model Controller (IMC) controller.

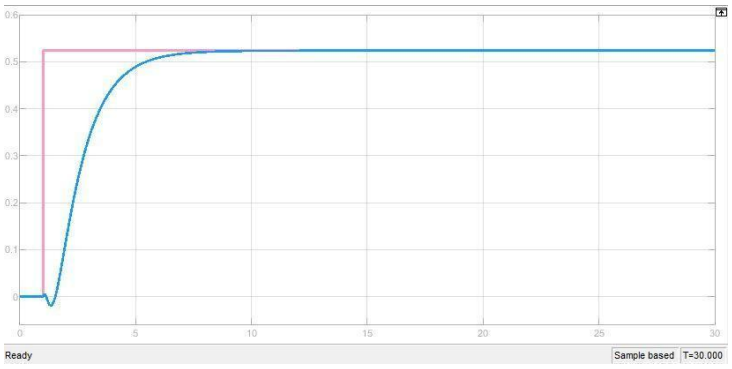


Figure 6.1: Step response of servo arm using LQR Controller

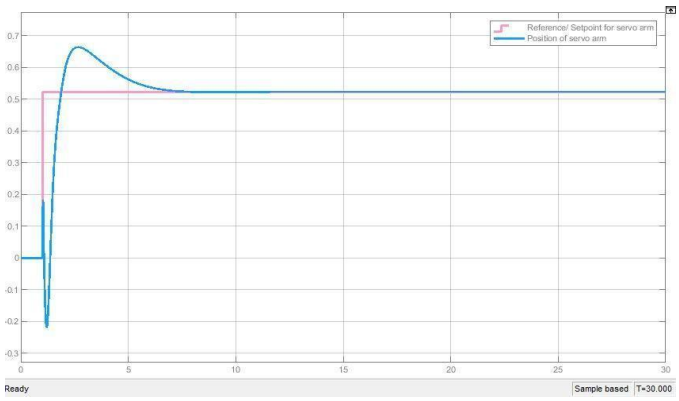


Figure 6.2: Step response of servo arm using PID Controller

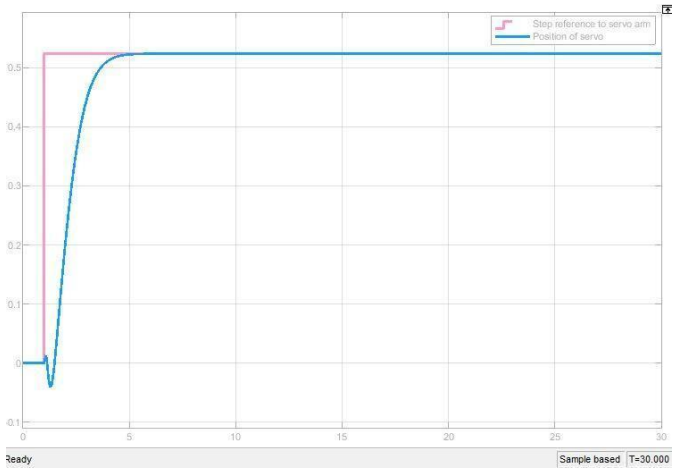


Figure 6.3: Step response of servo arm using Internal Model Control

ACKNOWLEDGMENT

I would like to express our deepest gratitude to the following people for guiding us through this course and without whom this project and the results achieved from it would not have reached completion.

Dr. D. Ezhilarasi, Assistant Professor, Department of Instrumentation and Control Engineering, for helping us and guiding us in the course of this project. Without her guidance, we would not have been able to successfully complete this project. Her patience and genial attitude is and always will be a source of inspiration to us.

We are also thankful to the faculty and staff members of the Department of Instrumentation and Control Engineering, our individual parents and our friends for their constant support and help.

Conflict of Interest:

The authors declare that they have no conflict of interest.

Data Availability Statement:

The authors confirm that the data supporting the findings of this study are available within the article and its supplementary materials.

REFERENCES

[1] Hany F. Mokbel Amr A. Roshdy Yu zheng Lin & Tongyu Wang. (2012). *Stabilization of real inverted pendulum using pole separation factor*.

[2] Narinder Singh & Sandeep Kumar Yadav. (2012). *Comparison of LQR and PD controller for stabilizing double inverted pendulum system*.

[3] Rachid Mansouri, Ibrahim Mustafa, Mehedi Ubaid, M Al-Saggaf & Maamar Bettayeb. (2019). *Stabilization of a double inverted rotary pendulum through fractional order integral control scheme*.

[4] Dimple J. Vora & Ms. Nabanita Basu. (2016). *Modelling and simulation of inverted pendulum using internal model control*.

[5] Dr. V. I George, Bipin Krishna, Deepak Chandran & Dr. I. Thirunavukkarasu. (2015). Modeling and performance comparison of triple PID and LQR controllers for parallel rotary double inverted pendulum.