Original Article

Modeling and Balancing Control of Inverted Pendulum

Wahied G. Ali Abdelaal [1,✉], Gihan Nagib [2]

1. Department of Computer and Systems Engineering, Faculty of Engineering, Ain Shams University, Cairo, EGY
2. Department of Electrical Engineering (Computers and Systems), Fayoum University, Cairo, EGY

Received: June 29, 2025 | Review began: July 15, 2025 | Review ended: April 13, 2026 | Published: April 21, 2026

Abstract

The cart-pendulum model plays a critical role in real-time control systems across diverse domains including robotics (balancing mechanisms and surgical manipulators), aerospace (missile/Unmanned Aerial Vehicle stabilization), industrial automation (crane sway mitigation and conveyor vibration suppression), energy infrastructure (wind turbine oscillation damping), and structural engineering (seismic dampers). Its inherent nonlinear dynamics necessitate rapid control interventions to maintain stability and operational efficiency. This paper presents a comprehensive development of the nonlinear dynamic model for an inverted pendulum using dual methodological approaches: Newtonian mechanics employing force-balance equations and Lagrangian formulation utilizing kinetic and potential energy principles. Linearization was performed about the upright equilibrium position, followed by derivation of a state-space representation. Controllability and observability were rigorously analyzed using MATLAB computational tools. Stability Analysis for closed loop system under Proportional-Integral-Derivative (PID) control is investigated using Routh-Hurwitz's criterion. This analysis is applied in four common control modes P, Proportional-Integral (PI), PD, and PID. The stability analysis approved that P and PD modes are not stabilizing modes, while PI and PID are stabilizing modes for inverted pendulum. Therefore, simulations in MATLAB/Simulink evaluated closed-loop behavior under PID control modes P, PI, PD, and PID. Extensive simulation results using MATLAB/SIMULINK affirmed the analytical study and showed that the inverted pendulum can be stabilized only using PI and PID control modes where tracking and regulation to reject disturbance effects are achieved.

Categories: Mechatronics and Robotics, Vehicle Dynamics and Control

Keywords: system modelling, self balancing robot, inverted pendulum, pid controller, stability analysis

Introduction

The Segway Robotic Transportation System is a self-balancing electric vehicle based on inverted pendulum dynamics, originally developed for personal mobility and now adapted for autonomous applications such as last-mile delivery and surveillance. Its core functionality relies on a sophisticated balance control system and is modeled as a cart-inverted pendulum that ensures stability under varying payloads and enables riderless operation. Utilizing inertial measurement units and Proportional-Integral-Derivative (PID) controllers, it detects and corrects imbalances in real time, ensuring safe and efficient navigation. As a zero-emission platform, it offers a sustainable solution for diverse mobility tasks [1,2]. The control algorithm maintains the system's center of gravity vertically aligned with the wheel axis, ensuring dynamic stability during navigation. Researchers have explored various approaches to improve the balancing control of Segway-style robots, including the implementation of advanced control techniques such as adaptive control [3], model predictive control [4], and deep learning-based methods [5]. These methods aim to enhance the system's robustness, responsiveness, and stability in the face of external disturbances and changing environmental conditions. Sliding mode control provides robust nonlinear stabilization for Segway platforms by enforcing state trajectory convergence to a predefined sliding manifold, effectively rejecting system uncertainties and disturbances [6,7]. Concurrently, LIDAR-SLAM integration enables centimeter-accurate environmental mapping, granting autonomous navigation capabilities in unstructured environments [8,9]. These advancements in balancing control and sensor integration have expanded the potential applications of Segway-based systems, from personal mobility to logistics and delivery tasks. The manuscript is structured as follows:

· Dynamic Modeling: Derivation of governing equations via Newton-Euler and Lagrange formulations.

· Control Synthesis: Implementation and tuning of PID using Routh-Hurwitz's stability criterion.

· Simulation Framework: High-fidelity dynamics validation using MATLAB/SIMULINK.

· Conclusion: Performance analysis and operational insights from simulation results.

Materials And Methods

Dynamic modeling

The Segway demonstrates inherent dynamic instability like that of an inverted pendulum, where the rider's center of mass functions analogously to the pendulum bob. Without active control, both systems would succumb to gravitational torque-the pendulum would fall to its stable downward equilibrium, and the Segway would topple. Stability is maintained through continuous dynamic compensation: the base (or cart) moves laterally to remain beneath the shifting center of mass. For instance, when the pendulum bob-or the rider's torso-leans rightward, the base accelerates rightward to counteract the imbalance, and vice versa. This active stabilization mechanism enables vertical equilibrium, modeled as a two-degree-of-freedom system (cart and pendulum) as shown in Figure *1* and analyzed in [10].

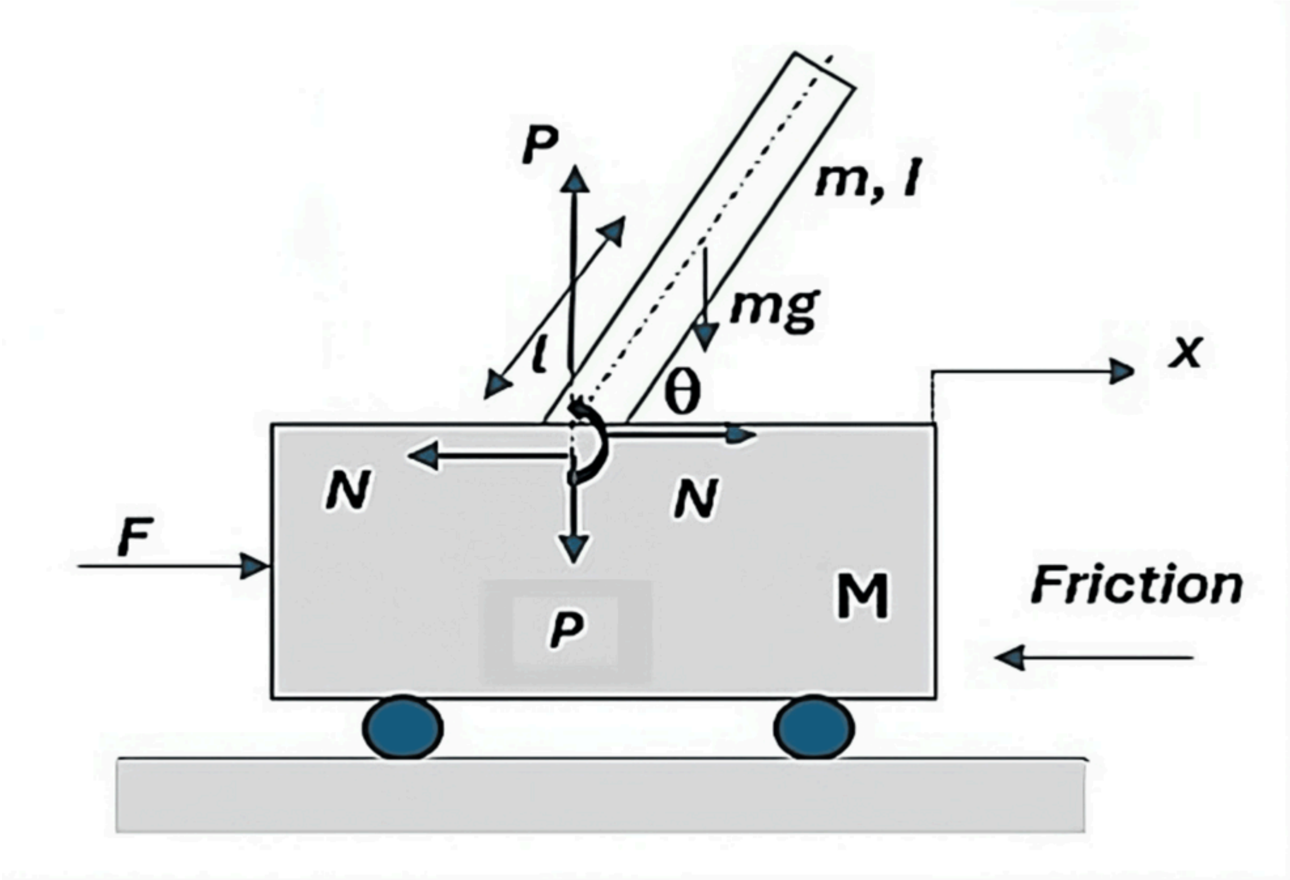

FIGURE 1: Inverted pendulum model

The model parameters are defined as follows [10-12]:

- *M* cart's mass (3 kg)
- *m* pendulum's mass (1.57 kg)
- *b* friction coefficient (0.1 N/m/s)
- *l* length to pendulum center of mass 2 m
- *I* inertia of the pendulum (8.37 kg.m^2)
- *F* force applied to the cart
- *N* reaction force applied to the cart in horizontal direction
- *P* reaction force applied to the cart in vertical direction
- *x* cart position coordinate
- Θ pendulum angle from vertical direction as shown in Figure *1*.
- *g* gravitational acceleration (9.8 m/s^2)

Where the pendulum moment of inertia is shown in Equation (1).

$I = \frac{1}{3}mL^2$ (1)

A. By Applying Newton's Second Law

This is the first approach for developing the dynamic model. By summing the forces in the free-body diagram of the cart in the horizontal direction, you get the following equation of motion.

$$M\ddot{x} + b\dot{x} + N = F \quad (2)$$

Summing the forces in the vertical direction for the cart is not useful to gain information. But, by summing the forces of in the horizontal direction, the following expression is obtained for the reaction force *N*.

$$N = m\ddot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta (3)$$

If you substitute Equation (3) into the Equation (2), you get one of the two governing equations for this system.

$$(M+m)\ddot{x} + b\dot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta = F(4)$$

To derive the second equation of motion for the system, consider the sum of forces perpendicular to the pendulum. Analyzing the system along this axis significantly simplifies the mathematical formulation. This yields the following equation:

$$P\sin\theta + N\cos\theta - mg\sin\theta = ml\ddot{\theta} + m\ddot{x}\cos\theta(5)$$

To eliminate the *P* and *N* terms from the above equation, take the moment of the pendulum's center of mass. This results in the following equation:

$$-Pl\sin\theta - Nl\cos\theta = I\ddot{\theta}(6)$$

Combining these last two Equations (5) and (6), you get the second governing equation.

$$(I+ml^2)\ddot{\theta} + mgl\sin\theta = -ml\ddot{x}\cos\theta(7)$$

Equations (4) and (7) represent the dynamic model which is a fourth order nonlinear dynamic model. The two equations are summarized below for the completeness

$$(M+m)\ddot{x} + b\dot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta = F(8)$$

$$(I+ml^2)\ddot{\theta} + mgl\sin\theta + ml\ddot{x}\cos\theta = 0(9)$$

This model can be represented in a generalized robotic form as follows:

$$D(q)\ddot{q} + C(q,\dot{q}) + G(q) = \tau(10)$$

Where:

D(q) is the inertia matrix

C is the Coriolis matrix

G is the gravitational vector

$\tau$is the driving forces/torques

By using matrix notations, we get

$$D(q) = \begin{bmatrix} M+m & ml\cos\theta \\ ml\cos\theta & I+ml^2 \end{bmatrix}; \quad \ddot{q} = \begin{bmatrix} \ddot{x} \\ \ddot{\theta} \end{bmatrix}$$

$$C(q,\dot{q}) = \begin{bmatrix} b\dot{x} - ml\dot{\theta}^2\sin\theta \\ 0 \end{bmatrix}; \quad G(q) = \begin{bmatrix} 0 \\ mlg\sin\theta \end{bmatrix}; \quad \tau = \begin{bmatrix} F \\ 0 \end{bmatrix}(11)$$

How to cite this article:
Ali Abdelaal W G, Nagib G (April 21, 2026) Modeling and Balancing Control of Inverted Pendulum. Cureus J Eng 3 : es44388-025-00051-w. DOI https://doi.org/10.7759/s44388-025-00051-w

B. Lagrange Method

Lagrange method is an energetic using kinetic and potential energies. Lagrange equation is defined as the following [13-18].

$$L = K - P(12)$$

Where K is the total kinetic energy and P is the total potential energy of the dynamic system.

The kinetic energy of cart = $\frac{1}{2}M\dot{x}^2$ (13)

The center of gravity of pendulum is defined as x_p and y_p coordinates:

$$x_p = x + l\cos(\theta - 90^\circ) = x + l\sin\theta$$
$$y_p = l\sin(\theta - 90^\circ) = -l\cos\theta(14)$$

Differentiating the above equation yields:

$$\dot{x}_p = \dot{x} + l\dot{\theta}\cos\theta; \quad \dot{y}_p = l\dot{\theta}\sin\theta(15)$$

The total kinetic energy is computed by the following equation.

$$K = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m(\dot{x}_p^2 + \dot{y}_p^2) + \frac{1}{2}I\dot{\theta}^2(16)$$

$$\dot{x}_p^2 + \dot{y}_p^2 = \dot{x}^2 + 2l\dot{x}\dot{\theta}\cos\theta + l^2\dot{\theta}^2\cos^2\theta + l^2\dot{\theta}^2\sin^2\theta$$
$$\dot{x}_p^2 + \dot{y}_p^2 = \dot{x}^2 + 2l\dot{x}\dot{\theta}\cos\theta + l^2\dot{\theta}^2; \quad \text{where } \sin^2\theta + \cos^2\theta = 1(17)$$

The potential energy is defined as:

$$P = mgl\sin(\theta - 90^\circ) = -mgl\cos\theta(18)$$

Therefore, Lagrange's equation will be.

$$L = \frac{1}{2}M\dot{x}^2 + \frac{1}{2}m(\dot{x}^2 + 2l\dot{x}\dot{\theta}\cos\theta + l^2\dot{\theta}^2) + \frac{1}{2}I\dot{\theta}^2 + mgl\cos\theta(19)$$

The dynamic equation for the cart will be as follows:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{x}}\right) - \frac{\partial L}{\partial x} = F - b\dot{x}$$
$$\frac{\partial L}{\partial x} = 0; \quad \frac{\partial L}{\partial \dot{x}} = (M+m)\dot{x} + lm\dot{\theta}\cos\theta$$
$$(M+m)\ddot{x} + lm(\ddot{\theta}\cos\theta - \dot{\theta}^2\sin\theta) = F - b\dot{x}(20)$$

$$(M+m)\ddot{x} + b\dot{x} + lm\ddot{\theta}\cos\theta - lm\dot{\theta}^2\sin\theta = F(21)$$

The dynamic equation for the pendulum will be as follows:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) - \frac{\partial L}{\partial \theta} = 0$$

$$\frac{\partial L}{\partial \theta} = -lm\dot{x}\dot{\theta}\sin\theta - mgl\sin\theta; \quad \frac{\partial L}{\partial \dot{\theta}} = lm\dot{x}\cos\theta + ml^2\dot{\theta} + I\dot{\theta}$$

$$(ml^2 + I)\ddot{\theta} + ml\ddot{x}\cos\theta + mlg\sin\theta = 0(22)$$

Equations in (21) and (22) are summarized below for completeness.

$$(M + m)\ddot{x} + b\dot{x} + ml\ddot{\theta}\cos\theta - ml\dot{\theta}^2\sin\theta = F$$

$$(I + ml^2)\ddot{\theta} + mgl\sin\theta + ml\ddot{x}\cos\theta = 0(23)$$

The above equations in (23) are the same results as developed in Equations (8 and 9) using Newton's second law. This result affirms the quality of the developed dynamic model using two different methods. Equation (23) can be linearized about the upward equilibrium position, $\Theta - \pi$. We assume that the system remains within a small neighborhood of the equilibrium point. This assumption is justified, as the control objective is to limit the pendulum's deviation to within 20 degrees of the upright position. The angle ϕ represents the deviation of the pendulum's position from equilibrium position, that is, $\Theta = \phi + \phi$ Again, presuming a small deviation from equilibrium, we can use the following small angle approximations of the nonlinear functions in our system equations [11]:

$$\cos\theta = \cos(\varphi + \pi) \approx -1; \quad \sin\theta = \sin(\varphi + \pi) \approx -\varphi$$

$$\ddot{\theta} = \ddot{\varphi}; \quad \dot{\theta} = \dot{\varphi} = 0(24)$$

After substituting the above approximations into nonlinear governing equations, we get the two linearized equations of motion. Note that, the input force F is replaced by u. Therefore, the linearized model is given by:

$$(M + m)\ddot{x} + b\dot{x} - ml\ddot{\varphi} = u$$

$$(I + ml^2)\ddot{\varphi} - mgl\varphi - ml\ddot{x} = 0(25)$$

Transfer Function

The transfer function from Equation (25) can be obtained by applying Laplace transformation on system equations assuming zero initial conditions. The resulting Laplace transformations are shown below.

$$(I + ml^2)\varphi(s)s^2 - mgl\varphi(s) = mlx(s)s^2$$

$$(M + m)x(s)s^2 + bx(s)s - ml\varphi(s)s^2 = u(s)(26)$$

Recall that a transfer function represents the relationship between a single input and a single output at a time. To find the transfer function for the output ϕ(s) and an input of *u*(s) we need to eliminate *x*(s) from the above equations. Solve the first equation for *x*(s).

$$x(s) = \left[\frac{I+ml^2}{ml} - \frac{g}{s^2}\right]\varphi(s)(27)$$

Then substitute the above into the second equation.

$$(M + m)\left[\frac{I+ml^2}{ml} - \frac{g}{s^2}\right]\varphi(s)s^2 + b\left[\frac{I+ml^2}{ml} - \frac{g}{s^2}\right]\varphi(s)s - ml\varphi(s)s^2 = u(s)(28)$$

Rearranging, the transfer function is then the following:

$$\frac{\varphi(s)}{u(s)} = \frac{\frac{ml}{q}s^2}{s^4 + \frac{b(I+ml^2)}{q}s^3 - \frac{(M+m)mlg}{q}s^2 - \frac{bmlg}{q}s}(29)$$

Where $q = \left((M+m)(I+ml^2) - (ml)^2\right)$

From the transfer function above, there is a pole and a zero at the origin. These can be canceled, and the transfer function becomes the following.

$$\frac{\varphi(s)}{u(s)} = \frac{\frac{ml}{q}s}{s^3 + \frac{b(I+ml^2)}{q}s^2 - \frac{(M+m)mlg}{q}s - \frac{bmlg}{q}} \quad (30)$$

State-Space Model

The linearized equations of motion in Equation (25) can be expressed in state-space form by rewriting them as a set of first-order differential equations. Given their linearity, they can be arranged into the standard matrix form shown below. Define the state variables as follows:

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} \varphi \\ \dot{\varphi} \\ x \\ \dot{x} \end{bmatrix}; \quad \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} x_2 \\ \ddot{\varphi} \\ x_4 \\ \ddot{x} \end{bmatrix}$$

Putting all the values of $\ddot{\varphi}$ and $\ddot{x}$ in Equations (25) we get

$$\ddot{x} = \frac{(ml)^2 g}{q} x_1 - \frac{(I+ml^2)b}{q} x_4 + \frac{I+ml^2}{q} u$$

$$\ddot{\varphi} = \frac{(M+m)mlg}{q} x_1 - \frac{mlb}{q} x_4 + \frac{ml}{q} u$$

Where $q = \left((M+m)(I+ml^2) - (ml)^2\right)$

The state space model will be in the standard form as follows:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \frac{(M+m)mlg}{q} & 0 & 0 & \frac{-mlb}{q} \\ 0 & 0 & 0 & 1 \\ \frac{(ml)^2 g}{q} & 0 & 0 & \frac{-b(I+ml^2)}{q} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{ml}{q} \\ 0 \\ \frac{I+ml^2}{q} \end{bmatrix} u$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \varphi \\ \dot{\varphi} \\ x \\ \dot{x} \end{bmatrix} (31)$$

The state-space model in matrix form using numerical values of parameters is given by:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 4.95 & 0 & 0 & -0.0111 \\ 0 & 0 & 0 & 1 \\ 3.4 & 0 & 0 & -0.0295 \end{bmatrix}; \quad B = \begin{bmatrix} 0 \\ 0.111 \\ 0 \\ 0.295 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}; \quad D = [0] (32)$$

The eigen values of the open loop without control using MTLAB commands are given by:

```
>> eig(a)

ans =

 0

 2.2211

 -2.2287

 -0.0219
```

The open loop system is unstable due positive eigenvalues. The controllability test for control design is investigated using MATLAB commands.

```
>> e=ctrb(a,b)

e =

 0.0000 0.1110 -0.0033 0.5495

 0.1110 -0.0033 0.5495 -0.0204

 0.0000 0.2950 -0.0087 0.3777

 0.2950 -0.0087 0.3777 -0.0223

>> rank(e)

ans = 4
```

The system is completely controllable where the rank of controllability matrix is equal to the system's order (four). The transfer function in Equation (33) is obtained using MATLAB Commands.

```
>>sys=ss(a, b, c, d)

>> tf(sys)

ans =
```

$$\frac{0.111\ S-3.904\times10^{-19}}{S^3+0.029\ S^2-4.95\ S-0.1083} \quad (33)$$

The constant value in numerator is too small and can be neglected. The obtained transfer function using MATLAB gives the same result as in Equation (30). The order of transfer function is less than the order of state-space model. It is expected that the system is not completely observable, and we can check this property using MATLAB. The observability test using MATLAB commands gives the following result:

```
>> e=obsv(a, c)

e =

 1.0000 0 0 0

 0 1.0000 0 0

 4.9500 0 0 -0.0111

 -0.0377 4.9500 0 0.0003

>> rank(e)

ans = 3
```

The rank of observability matrix is not equal to the system's order. Therefore, the system is not completely observable and consequently the transfer function is reduced in order. We have almost a zero at the origin (35.2e-19) and the roots of the transfer function computed by MATLAB command are given by:

```
>> roots([1 0.0295 -4.95 -0.1083])

ans =

 2.2211

 -2.2287

 -0.0219
```

Control design

The PID control equation in S-domain is given by:

$$PID(s) = K_p + \frac{K_i}{s} + K_d s \quad (34)$$

Where

K_p Proportional gain

K_i Integral gain

K_d Derivative gain

S Laplace operator

Different researchers have applied PID control for inverted pendulum in [16,18,19]. PID controllers have four modes to be used in industry to limit the gain at low and high frequencies. These modes are P, PI, PD, and PID. Therefore, we will investigate stability condition using the transfer function of inverted pendulum in Equation (29).

Stability Condition Using Proportional Controller

Assume a unity feedback control loop using K_p as the controller, and the closed loop stability can be investigated using the Routh-Hurwitz's criterion (Table 1). The closed loop c/c's equation will be:

$$s^3 + 0.0295s^2 + (0.111K_p - 4.95)s - 0.1083 \quad (35)$$

s^3	1	0.111Kp-4.95
s^2	0.0295	-0.1083
s^1	$0.111K_p$ -1.2788	0
s^0	-0.1083	0

TABLE 1: Routh's table for P controller

It is clear from the table that proportional controllers alone cannot stabilize the system due to the change of sign in the first column independent of Kp values.

Stability Condition Using PI Controller

Assume a unity feedback control loop using ($K_p + K_i/S$) as the controller, and the closed-loop stability can be investigated using the Routh-Hurwitz's criterion (Table 2). The closed-loop c/c's equation will be:

$$s^3 + 0.0295s^2 + (0.111K_p - 4.95)s + (0.111K_i - 0.1083) \quad (36)$$

S^3	**1**	**$0.111*K_p-4.95$**
S^2	0.0295	$0.111*K_i-0.1083$
S^1	$0.111*K_p -3.762711*K_i-1.2788136$	0
S^0	$0.111*K_i-0.1083$	0

TABLE 2: Routh's table for PI controller

PI, Proportional-Integral

From the above Routh's table, to guarantee that there is no change in the sign in the first column, we must satisfy the following conditions

$0.111*K_i-0.1083 > 0$ and $K_i > 1$, select $K_i = 2$ &

$0.111*K_p -3.762711*K_i-1.2788136 > 0$ and $K_p > 79.32$, select $K_p = 90$

Stability Condition Using PD Controller

Assume a unity feedback control loop using ($K_p + K_dS$) as the controller, and the closed-loop stability can be investigated using the Routh-Hurwitz's criterion (Table 3). The closed-loop c/c's equation will be:

$$s^3 + (0.0295 + 0.111K_d)s^2 + (0.111K_p - 4.95)s - 0.1083 \quad (37)$$

s^3	**1**	**$0.111K_p$-4.95**
s^2	$0.0295 + 0.111K_d$	-0.1083
s^1	$(0.111Kp-4.95) + 0.1083/(0.0295 + 0.111K_d)$	0
s^0	-0.1083	0

TABLE 3: Routh's table for PD controller

From the above Routh's table, the PD controller cannot stabilize the system due to the permanent change of the sign in the first column.

Stability Condition Using PID Controller

Assume a unity feedback control loop using ($K_p + K_i/S + K_dS$) as the controller, the closed-loop stability can be investigated using the Routh-Hurwitz's criterion (Table *4*). The closed-loop c/c's equation will be:

$$s^3 + (0.0295 + 0.111K_d)s^2 + (0.111K_p - 4.95)s + (0.111K_i - 0.1083) \quad (38)$$

s^3	**1**	**$0.111K_p$-4.95**
s^2	$0.0295 + 0.111K_d$	$0.111K_i$-0.1083
s^1	$(0.111Kp-4.95) + (0.1083-0.111K_i)/(0.0295+0.111K_d)$	0
s^0	$0.111K_i$-0.1083	0

TABLE 4: Routh's criterion for PID controller

PID, Proportional-Integral-Derivative

From the above Routh's table, to guarantee that there is no change in the sign in the first column, we must satisfy the following conditions:

$(0.111*K_i-0.1083) > 0$ and $K_i > 1$, select $K_i = 2$ &

$(0.0295 + 0.111*K_d) > 0$ and $K_d > -0.2657$, select $K_d = 1$

$(0.111K_p-4.95) + (0.1083-0.111K_i)/(0.0295 + 0.111K_d) > 0$

By substituting $K_i = 2$ and $K_d = 1$, K_p must be >51.89 and select $K_p = 100$

Results And Discussion

Simulation work

Figure 2 illustrates the open-loop block diagram implemented in MATLAB/SIMULINK, which represents the nonlinear dynamic model described by Equation (23). The diagram provides a visual representation of how system states and inputs interact in the absence of feedback control. Each block corresponds to a specific physical or mathematical component in the model, such as integration of motion equations, external input forces, and state variable transformations.

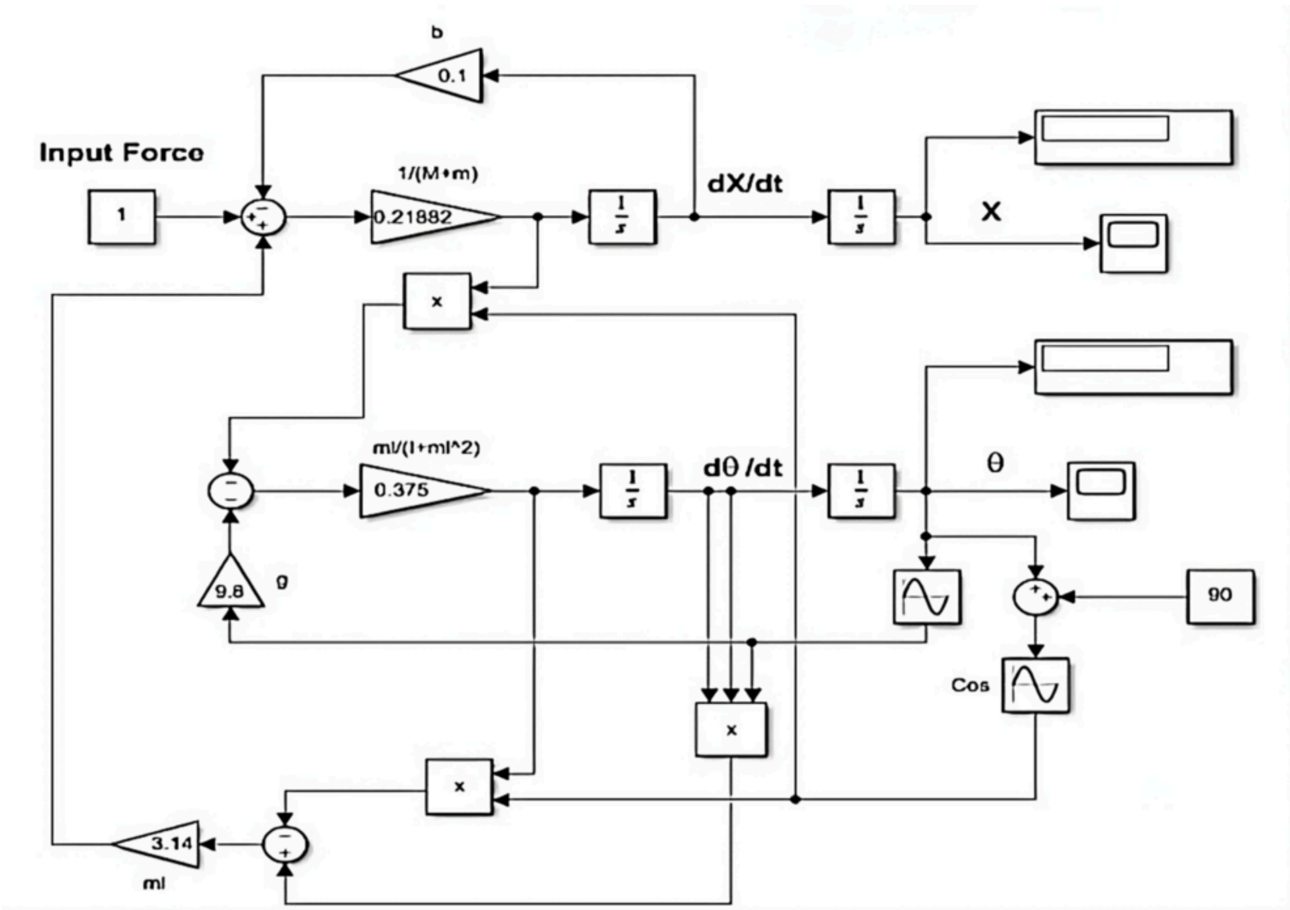

FIGURE 2: Nonlinear dynamic model (open loop)

Figure 3 illustrates the step response of the nonlinear model, providing insight into the system's transient and steady-state behavior. Unlike linear systems-where the response is solely determined by system parameters such as poles and zeros-the stability and performance of nonlinear systems are highly sensitive to initial conditions. This figure highlights how the system responds to a unit step input, emphasizing potential nonlinear effects such as overshoot, oscillations, or divergence depending on the starting state.

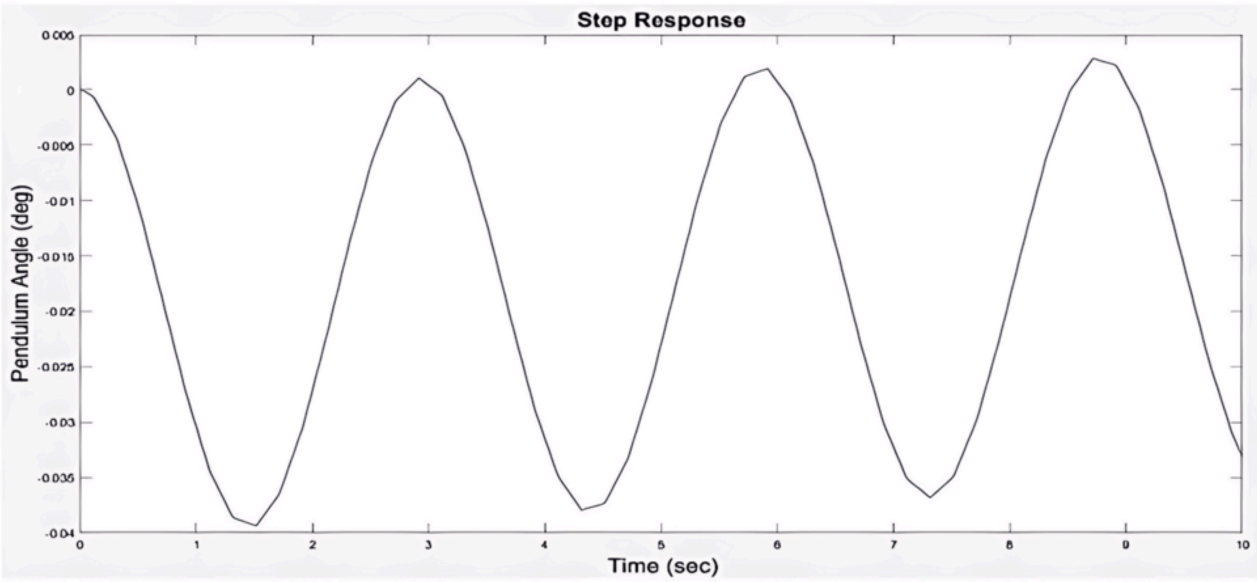

FIGURE 3: Step response for nonlinear model

Figure 4 presents the MATLAB/SIMULINK implementation of the linearized dynamic model described by Equation (25). This model results from approximating the original nonlinear system around an equilibrium point using first-order Taylor series expansion. The linearization process simplifies the system dynamics, making it more tractable for control design, stability analysis, and simulation. The block diagram captures the state-space representation or transfer function form of the linear system, allowing for straightforward evaluation of system responses under various input conditions.

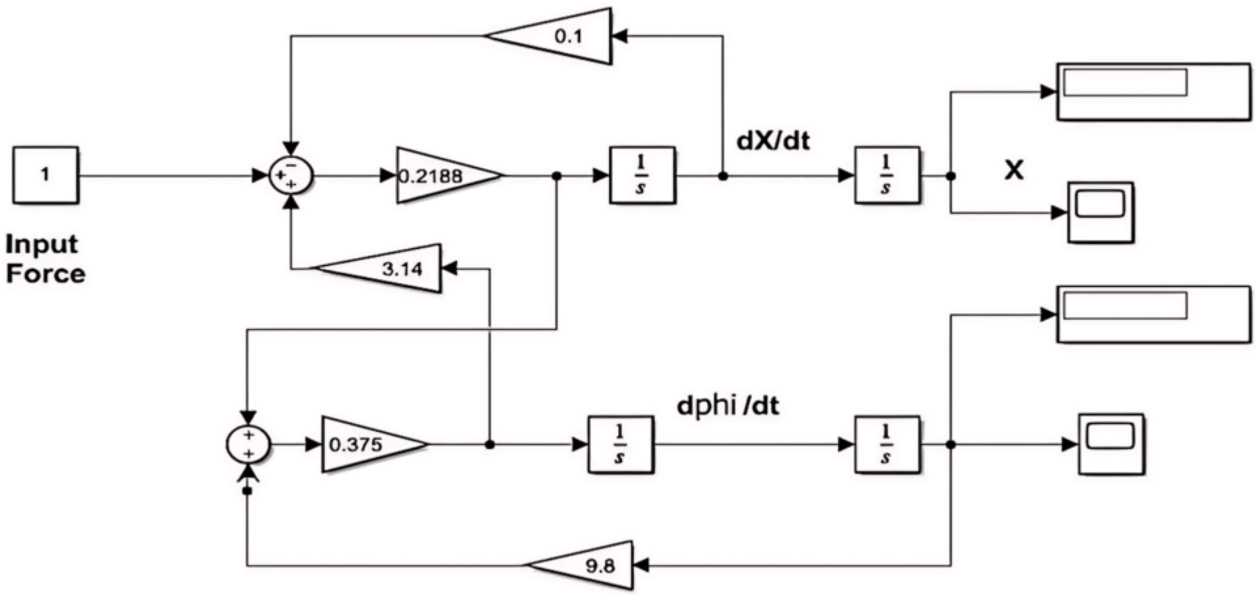

FIGURE 4: Linearized model (open loop)

Figure 5 illustrates the step response of the linearized model, revealing that the system exhibits unstable behavior. This response indicates that, even after linearization, the system dynamics around the chosen equilibrium point result in divergence over time rather than convergence to a steady state. The output grows without bound, confirming the presence of unstable poles in the system's transfer function or state-space representation. This instability suggests that the open-loop linearized system is inherently unstable and cannot maintain equilibrium in response to disturbances or step inputs. The figure underscores the need for designing an appropriate feedback control strategy to shift the system poles to the left-half of the complex plane and achieve the desired stability and performance.

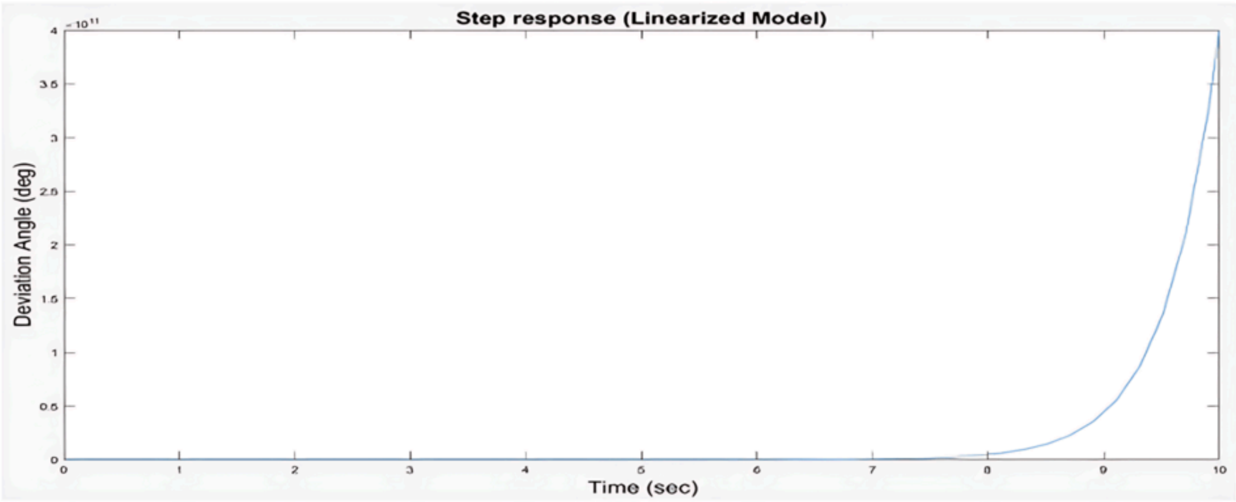

FIGURE 5: Step response for linearized state-space model

Figure 6 shows the step response of the linearized model, generated using its transfer function representation in MATLAB. This approach captures the input-output dynamics of the system without explicitly modeling the internal state variables, making it a convenient method for analyzing system behavior in the frequency and time domains. The step response provides insights into key characteristics such as rising time, settling time, overshoot, and stability. By comparing this response to the state-space simulation, one can validate the accuracy and consistency of both modeling approaches.

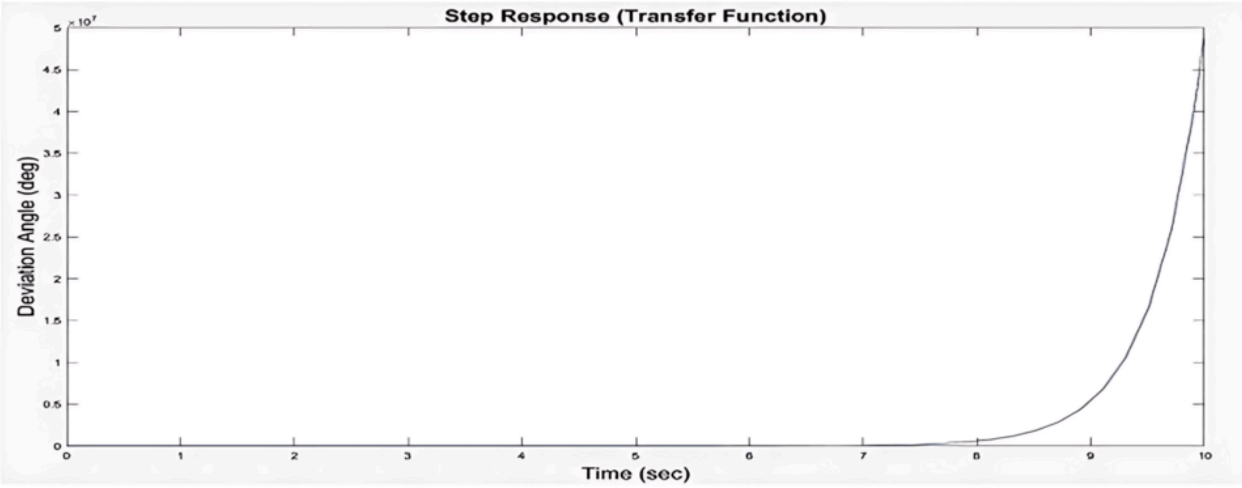

FIGURE 6: Step response for a linearized model (transfer function)

The linear system described in Equation (33) is controlled using a PID controller, as implemented in Figure 7. This figure shows the SIMULINK block diagram, where the PID controller is tuned to stabilize the system and ensure desirable dynamic behavior. This implementation marks a critical step toward achieving closed-loop stability, improved tracking accuracy, and robust disturbance rejection.

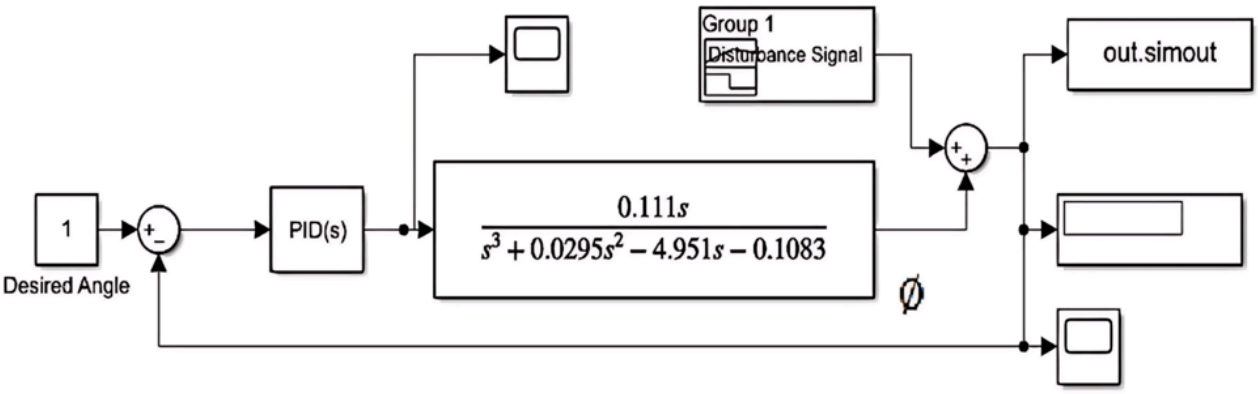

FIGURE 7: Closed-loop control using linearized model

By employing the stabilizing mode of PI control, with controller gains set to K_p = 90 and K_i = 2, the system's response is depicted in Figure 8. As shown in the figure, the system achieves stable behavior, maintaining the output close to the desired equilibrium position, with a maximum deviation of approximately two degrees. The figure confirms that the chosen Proportional-Integral (PI) gains provide sufficient control authority to counteract the inherent instabilities of the open-loop system, resulting in a smooth and well-damped response.

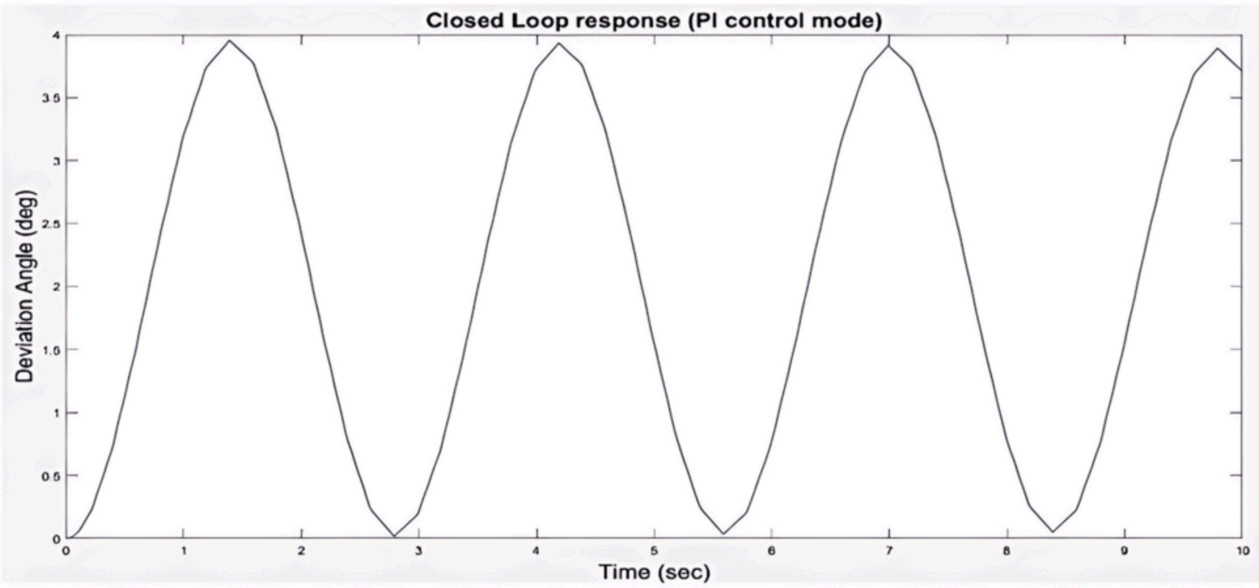

FIGURE 8: Closed loop response using PI with Kp = 90 and Ki = 2

PI, Proportional-Integral

The PI control mode produces an oscillatory response, primarily due to the delay introduced by the integral action, which accumulates error over time and can lead to overshoot and sluggish correction. This behavior is especially noticeable in systems with fast dynamics or when the integral gain is relatively high. To address this issue, the introduction of a derivative component is proposed, leading to a full PID control strategy. The derivative term acts on the rate of change of the error, providing a predictive element that helps dampen oscillations and improve response speed. Figure 9 presents the closed-loop control configuration using the PID stabilizing mode, where the controller gains are set to $K_p = 2$, $K_d = 1$, and $K_i = 70$. With these parameters, the system achieves enhanced performance, exhibiting improved stability, reduced overshoot, and faster settling time compared to the PI-only controller. The addition of derivative action mitigates the lag caused by integration and allows for more efficient and responsive control, demonstrating the effectiveness of PID tuning in stabilizing the system and optimizing its dynamic behavior.

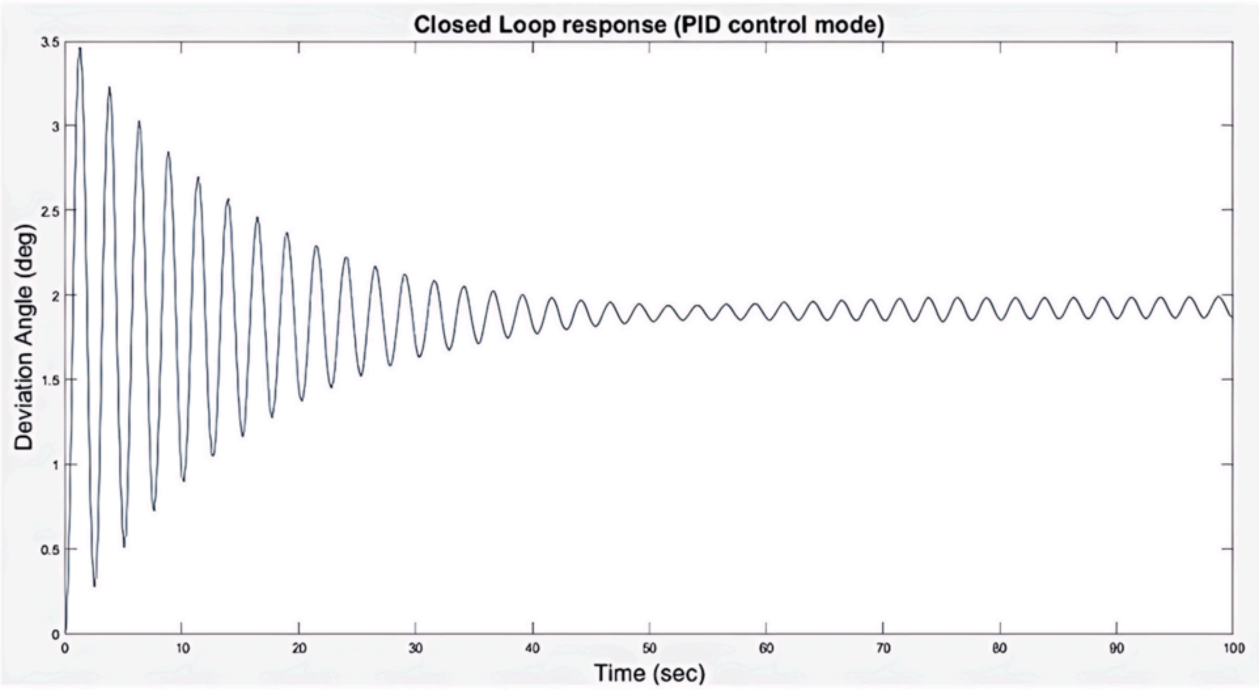

FIGURE 9: Closed loop response using PID control with Kp = 100, Ki = 2, and Kd = 1

PID, Proportional-Integral-Derivative

It is observed that the system is successfully stabilized, maintaining its output within approximately two degrees of deviation from the upright equilibrium position of the pendulum. This small deviation indicates that the closed-loop control strategy is effectively maintaining balance while compensating for nonlinear dynamics. The deviation error continues to decay over time, confirming that the analytical stability condition derived during the controller design is satisfied. This behavior reflects a stable and well-damped response, with the system naturally returning to equilibrium after small perturbations. Figure *10* illustrates the system's disturbance rejection capability, where a constant disturbance of unit magnitude is introduced at t = 60 seconds. The figure demonstrates that the control system can detect and counteract the disturbance, restoring the pendulum to its original operating point with minimal overshoot and no sustained oscillations. This confirms the robustness of the control design in handling external perturbations and maintaining steady-state accuracy in the presence of constant disturbances.

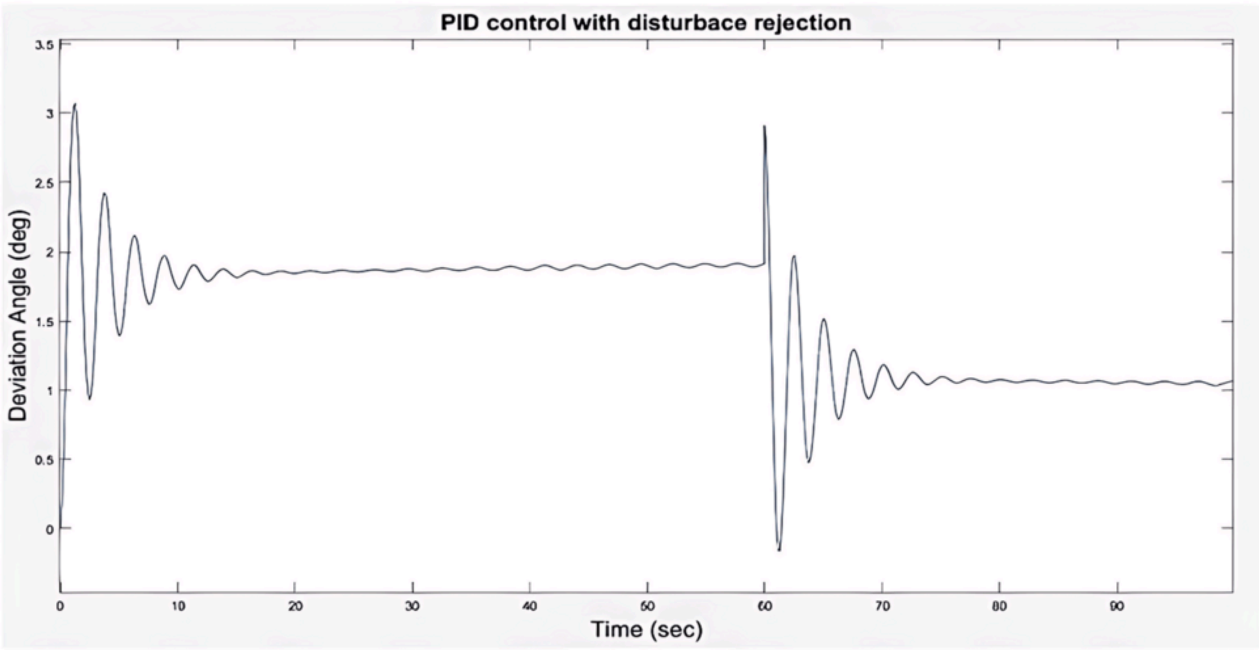

FIGURE 10: Closed loop response using PID control with Kp = 100, Ki = 2, and Kd = 5

PID, Proportional-Integral-Derivative

The above results clearly demonstrate the effectiveness of the control design in rejecting external disturbances and maintaining system stability. Specifically, the system is subjected to a unit step disturbance at 60 seconds, and the controller responds promptly, minimizing the impact of the disturbance while successfully driving the system back toward its desired equilibrium state. This behavior highlights the robustness and responsiveness of the implemented control strategies. The simulation results validate that both PI and PID control modes can stabilize the linearized dynamic model of the inverted pendulum, a system known for its inherent instability and nonlinear behavior. While the PI controller offers sufficient performance for basic stabilization and steady-state error elimination, the PID controller provides enhanced dynamic performance, reducing overshoot and settling time through the addition of derivative action. Overall, the results confirm that the proposed controllers are effective in achieving stability, disturbance rejection, and tracking performance for the linearized model, making them suitable candidates for real-time implementation.

Conclusions

The cart-inverted pendulum, a highly unstable, nonlinear, and underactuated system, serves as an excellent benchmark for control engineers to validate algorithms, particularly linear PID controllers. Its mathematical model, derived via two analytical methods-Newton's second law with force balance equations and the Lagrange approach using kinetic and potential energy-yields a fourth-order nonlinear system with coupled cart-pendulum dynamics. Linearization around the pendulum's upright position produces a state-space model, where MATLAB-based analysis confirms controllability and observability. The resulting third-order transfer function contains an unstable pole, necessitating a control design that first stabilizes the system and then optimizes performance. The linear PID controller is implemented in its four common modes: P, PI, PD, and PID. Prior to simulation, stability conditions for the unity-feedback closed-loop system are analyzed using the Routh-Hurwitz criterion for each mode. The findings are summarized as follows:

> P and PD control modes cannot stabilize the system.

> PI controller mode stabilizes the system with $K_i > 1$ and $K_p > 79.32$. But it produces oscillatory response.

> PID controller mode stabilizes the system with $K_p > 1$, $K_d > -0.2657$, $K_i > 51.89$. It gives superior performance.

Extensive simulations confirm that the balancing control system significantly enhances stability and disturbance rejection via PI and PID control modes, advancing the development of self-balancing cart-inverted pendulum platforms. The future work will be focused on two directions:

> Fuzzy control or adaptive PID control design to stabilize the nonlinear model.

> Experimental work to investigate stability and system performance in a real-time environment.

Additional Information

Author Contributions

All authors have reviewed the final version to be published and agreed to be accountable for all aspects of the work.

Concept and design: Wahied G. Ali Abdelaal, Gihan Nagib

Acquisition, analysis, or interpretation of data: Wahied G. Ali Abdelaal, Gihan Nagib

Drafting of the manuscript: Wahied G. Ali Abdelaal, Gihan Nagib

Critical review of the manuscript for important intellectual content: Wahied G. Ali Abdelaal, Gihan Nagib

Supervision: Wahied G. Ali Abdelaal, Gihan Nagib

Disclosures

Human subjects: All authors have confirmed that this study did not involve human participants or tissue. **Animal subjects:** All authors have confirmed that this study did not involve animal subjects or tissue. **Conflicts of interest:** In compliance with the ICMJE uniform disclosure form, all authors declare the following: **Payment/services info:** All authors have declared that no financial support was received from any organization for the submitted work. **Financial relationships:** All authors have declared that they have no financial relationships at present or within the previous three years with any organizations that might have an interest in the submitted work. **Other relationships:** All authors have declared that there are no other relationships or activities that could appear to have influenced the submitted work.

Acknowledgements

Authors would like to thank the following free AI tools to improve manuscript quality: 1- Original text is written by authors themselves, AI like ChatGPT is used only to improve authors-generated texts for readability and style, and to ensure that the texts are free of errors in grammar and spelling. 2- Free AI tool is used to enhance Figures quality, https://imgupscaler.ai/.

References

1. Mohsin HH, Aldair AA, Al-Hussaibi WA: Robust control design for two-wheel self-balanced. Iraqi Journal for Electrical and Electronic Engineering. 2022, 19:38-46.
2. Nader K, Sarsri D: Modelling and control of a two-wheel inverted pendulum using fuzzy-PID-modified state feedback. Journal of Robotics. 2023, 2023:1-13.
3. Tsai CC, Huang HC, Lin SC: Adaptive neural network control of a self-balancing two-wheeled scooter. IEEE Transactions on Industrial Electronics. 2010, 57:1420-1428. 10.1109/TIE.2009.2039452
4. Zhang H, Nor NM: Control strategies for two-wheeled self-balancing robotic systems: a comprehensive review. Robotics. 2025, 14:101.
5. Baltes J, Christmann G, Saeedvand S: A deep reinforcement learning algorithm to control a two-wheeled scooter with a humanoid robot. Engineering Applications of Artificial Intelligence. 2023, 126:106941.

6. Babazadeh R, Khiabani AG, Azmi H: Optimal control of Segway personal transporter. 2016 fourth International Conference on Control, Instrumentation, and Automation (ICCIA), Qazvin, Iran. 2016, 18-22.
7. Shtessel Y, Edwards C, Fridman L, Levant A : Sliding Mode Control and Observation. Birkhäuser New York, 2014.
8. Wang E, Huo W, Xu S, Qu P, Na L: Autonomous navigation of indoor wheeled robots based on improved Gmapping and improved Bidirectional A*. Springer Nature, 2025.
9. Mester G: Applications of Mobile Robots. Conference: 7th International Conference on Food Science At: Szeged, Hungary. 2006,
10. Lima JL, Goncalves JC, Costa PG, Moreira A: Inverted pendulum virtual control laboratory. The 7th Portuguese Conference on Automatic Control - CONTROLO'2006. 2006,
11. Inverted Pendulum: System modelling, Control Tutorials for MATLAB and Simulink - Inverted Pendulum: System Modeling. (2025). Accessed: June 16, 2025: https://ctms.engin.umich.edu/CTMS/index.php?example=InvertedPendulum&section=SystemModeling.
12. Sultan K: Inverted Pendulum Analysis, Design and Implementation. 2003.
13. Spong MW, Hutchinson S, Vidyasagar M: Robotic Modeling and Control. John Wiley & Sons, Ltd, 2020.
14. Corke P: Robotics, Vision and Control: Fundamental Algorithms in MATLAB. Springer Cham, 2023.
15. Lynch KM, Park FC: Modern Robotics: Mechanics, Planning, and Control. Cambridge University Press, 2017.
16. Rani M, Kamlu SS: Optimal LQG controller design for inverted pendulum systems using a comprehensive approach. Scientific Reports. 2025, 15:4692. 10.1038/s41598-025-85581-3
17. Agarana MC, Akinlabi ET, Agboola OO: Lagrangian-Taylor differential transformation analysis of self-balancing inverted pendulum robot. International Journal of Advanced and Applied Sciences. 2019, 6:54-57.
18. Siradjuddin I, Amalia Z, Setiawan B, et al.: Stabilising a cart inverted pendulum with an augmented PID control scheme. MATEC Web of Conferences. 2018, 197:11013. 10.1051/matecconf/201819711013
19. Tun HM, Nwe MS, Naing ZM, Latt MM, Pradhan D, Sahu PK: Research on self-balancing two wheels mobile robot control system analysis. Electrical Science & Engineering. 2022, 4:7-20. 10.30564/ese.v4i1.4398