

Modelling and Implementation of PID Control for Balancing of an Inverted Pendulum

Buddhika Abeysekera, Indika Wanniarachchi*

Department of Physics, University of Sri Jayewardenepura, Nugegoda, Sri Lanka

Abstract

Inverted pendulum (IP) has become a popular topic among physicist and control engineers due to advanced applications related to this particular pendulum. Two wheel balancing robots, rockets and missile guidance systems are some complex applications of the IP. The IP is highly unstable and to maintain the vertical upright position, an active control system is needed. Proportional, Integral and Derivative (PID) controller is one of the best controlling methods used in various dynamic control systems recently because of the simplicity and applicability. In this paper, two PID controllers are used to balance the IP and an error optimization method is used for filtering unwanted sensor responses. The PID controllers have been designed for controlling of each sub-system. PID gain parameters are tuned separately and manually using trial and error approach. Mathematical model of this IP is developed to determine the dynamic properties of the IP system and through this model, state space model of the IP system is developed. With the help of state space model, Kalman filter is developed and used to optimize the error in sensor readings. Then control system of the IP based on the PID controllers and the Kalman filter (control algorithm) is implemented on a microcontroller based platform in order to balance the inverted pendulum on a trolley using the mechanical system which is controlled by the microcontroller. Main objective of this control system is to balance the IP in an upright position in the middle of the two rails. Behavior of the IP system with this control algorithm is monitored through the real-time data acquisition system. The data of the IP system is used to contrast the IP behavior with each PID gain parameter through a graphical representation. With proper gain parameters, the IP system shows best smooth behavior of it.

Keywords

Inverted Pendulum, Kalman Filter, Lagrangian, Microcontroller, PID, Space State Modeling

Received: October 30, 2018 / Accepted: November 23, 2018 / Published online: December 6, 2018

© 2018 The Authors. Published by American Institute of Science. This Open Access article is under the CC BY license.

<http://creativecommons.org/licenses/by/4.0/>

1. Introduction

Inverted pendulum is a classical problem in physics. The inverted pendulum is also popular among engineers due to its applications in relevant fields. This system is also used in study purposes to demonstrate and analyze some of the computer algorithms based on control theory. At present, inverted pendulum [1] and PID controllers [2] are popular due to their applications in robotics [3], automotive, aerospace, advanced mobility systems and military etc. A lot of researches have been done to analyze the IP behavior

through different types of controlling methods. In all those controllers, the primary object is to balance the IP in a vertically upright state while managing external disturbances [4-5].

In many of the cases, system control algorithms were developed by considering the Newtonian mechanics or the Lagrangian equations. The Lagrangian is used to construct the mathematical model. But the system model is a non-linear mathematical model. Therefore, it is converted into a simple linear model using space state modeling and the space state model is used furthermore to prove that the system is

* Corresponding author
E-mail address: iwanni@sjp.ac.lk (I. Wanniarachchi)

controllable and so on. This classical approach is popular due to its applicability on most of the control algorithms. These control algorithms are developed on computers [6] in order to analyze and check whether the algorithm is working as expected by giving desired results.

Computer based control systems consist of sensors which are used to get the system feedback. In the feedback of the system, noises could be included. Therefore, it is important to reduce these noises in the feedback signals in order to develop a better control algorithm. Recently Kalman filter has got a remarkable place in error optimizations [7] in the field of control engineering and aerospace applications. This filter is popular due to its predictor corrector behavior. When it comes to the development stage of the Kalman filter, a set of linear equations are used to optimize the sensor outputs and because of that the implementation has become less complex than most of other error optimization methods. In this paper, the control algorithm based on PID controllers is implemented on a microcontroller and used to analyze and contrast the performance of the inverted pendulum according to each PID gain parameters. Kalman filter is also used as the error optimizer in the control algorithm slightly with the PID controllers.

2. Review of Related Literature

The IP problem [8] has become a famous topic among the physicist and control engineering community. Due to the complexity and the applicability of the IP problem, a lot of research papers have been published on it. Following literature reviews are to give a brief idea about practical applications of IP control systems, PID controllers and Kalman filter.

2.1. Applications

Segway is a personal transport vehicle [1] that uses an autonomous two-wheel balancing platform. The Segway is also based on the IP model. Segway is balanced by computer algorithms and position sensors inside it with the help of three gyroscopes and two backup gyroscopes. But the cost is considerably higher than the other transportation methods.

2.2. PID Controllers in IP Problems

The Microchip Company [9] has developed an IP control system using their PIC16F684 microcontroller. PID controller has been used as the control algorithm in their solution. In order to eliminate the noise signals, a low pass filter (the Bessel filter) has been used. This is a simple method because the horizontal position has not been involved here. That means the system needs only one PID controller.

Jia-Jun Wang [6] has published a research paper on the IP

and the PID based control system. In this paper he has developed three types of IP models. The models are based on 1D, 2D and 3D space coordinates. For each model the equations of the motion have been determined using the Lagrangian of the system. Then nonlinear equations of the motion were converted into linear equations using space state systems. Then the linearized system has been used to apply the PID algorithm. However, this is just a computer-based simulation. New computers have enough processing power to handle this kind of long and complex algorithms.

Mahbubeh Moghaddas et al, [10] have tried to implement a control system for the IP problem using Genetic Algorithm (GA). Even in this project, the mathematical model of the pendulum has been derived using Lagrangian equation. They have used MATLAB Simulink to build up the model in a computer. Parameter adjustment through hard mathematical calculation take too much time at different problems. In this method the researchers have tried to give an optimal answer with the GA.

2.3. New Trends of PID Controllers

In the paper by Kada and Ghazzawi [2] have proposed an effective method for implementing of PID controller to UAV. The reason was the trial and error or analyze and iterate tuning process of PID algorithms consumes a large amount of manpower until the desired performance of the control system is obtained. The new method is about to propose a mathematical based framework for designing PID controllers for the complex systems like airplanes. Here the method known as deadbeat step response was used combined with the proposed solution. This method was to reduce the number of tuning parameters in complex PID algorithms.

In the paper by Moghavvemi et al. [11], behavior of a quadcopter was studied due to the interesting features of this small size vehicle and the dynamic features of it. In quadcopters, the rotational speed of the four rotors are independent. Therefore, three-dimensional control axes called pitch, roll and yaw can be controlled individually. This is caused to increase the mobility of the vehicle. In order to make this UAV (or drone) balance in the air, a proper control system is needed and this paper discusses about the PID controller to fulfill that requirement.

2.4. Kalman Filter as State Update Algorithm

As described by Greg Welch and Gary Bishop, “The Kalman filter is a set of mathematical equations that provides an efficient computational (recursive) means to estimate the state of a process, in a way that minimizes the mean of the squared error. The filter is very powerful in several aspects: it supports estimations of past, present, and even future states,

and it can do so even when the precise nature of the modeled system is unknown.” [12]. The filter provides an accurate current state of the system. This property is used in the IP problem to measure the vertical angle of the pendulum to control the stability of the pendulum.

In the article “Kalman Filtering” written by Dan Simon, [13] has mentioned that “The Kalman filter is a tool that can estimate the variables of a wide range of processes. In mathematical terms we would say that a Kalman filter estimates the states of a linear system. The Kalman filter not only works well in practice, but it is theoretically attractive because it can be shown that of all possible filters, it is the only one that minimizes the variance of the estimation error. Kalman filters are often implemented in embedded control systems because in order to control a process, you first need an accurate estimation of the process variables”.

3. Theoretical Approach to Mathematical Model

Physical behavior of the pendulum is analyzed by considering the kinetic and potential energies of the system. The objective of this approach is to use the Euler-Lagrangian

Equations to develop the mathematical model. Then the equations are used to construct the control system through mathematical analysis to keep the pendulum in a vertically upright position.

In this inverted pendulum, the pendulum has been pivoted on a movable cart (or Trolley) which can move along two rails. Therefore, Lagrangian of the pendulum system can be modelled using classical mechanical equations through the physical parameters (Table 1) of the pendulum system. The pendulum can move on x-y plane while the cart can only move in the x- direction as shown in the Figure 1.

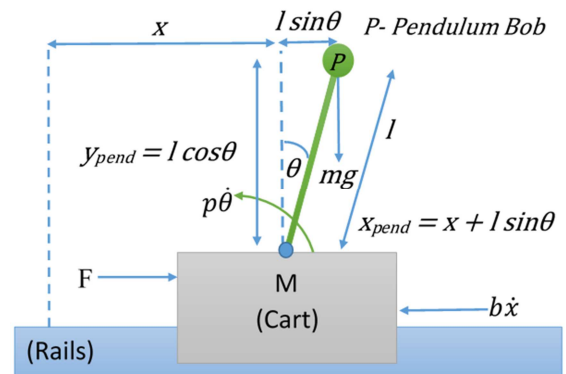


Figure 1. Inverted Pendulum System.

Table 1. IP System Physical Parameters in Si Units.

Parameters	Explanations	Value	Units
M	Mass of the cart	0.48	kg
m	Mass of the pendulum	0.03	kg
g	Gravitational acceleration	9.8	ms^{-2}
b	Friction Coefficient of the cart	0.01	kg/s
p	Damping Coefficient of the Pendulum	0.01	kg/s
l	Length to pendulum center of mass	0.3	m
F	Force applied to the cart	$F(t)$	kgms^{-2}
x	Cart position coordinate	$x(t)$	m
θ	Pendulum angle from vertical	$\theta(t)$	rad

3.1. Classical Mechanical Approach

The Lagrange of the system is defined to start the mathematical modeling process as,

$$L = T - V \quad (1)$$

Here, L is the Lagrangian of the system, T is the kinetic energy and V is the potential energy of the system. Then the Euler-Lagrangian equation can be defined as,

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q} = Q \quad (2)$$

Where, q is the coordinate system and the Q is the generalized force.

All the forces which are acting on the inverted pendulum has been shown in the Figure 1. Frictional forces which are acting on the system are denoted in $b\dot{x}$ and $p\dot{\theta}$.

The kinetic energy of the system can be defined as,

$$T = \frac{1}{2} M \dot{x}^2 + \frac{1}{2} m (\dot{x}_{pend}^2 + \dot{y}_{pend}^2) \quad (3)$$

$$x_{pend} = x + l \sin \theta \quad (4)$$

$$y_{pend} = l \cos \theta \quad (5)$$

Then 4, 5 equations are differentiated with respect to time to substitute for the equation 3.

$$\dot{x}_{pend} = \dot{x} + l\dot{\theta} \cos \theta \quad (6)$$

$$\dot{y}_{pend} = -l\dot{\theta} \sin \theta \quad (7)$$

Then the equation 3 becomes,

$$T = \frac{1}{2} (M + m) \dot{x}^2 + m \dot{x} l \dot{\theta} \cos \theta + \frac{1}{2} m l^2 \dot{\theta}^2 \quad (8)$$

The potential energy of this IP is,

$$V = mgl \cos \theta \quad (9)$$

Now using equations 1, 8 and 9, the Lagrangian of the IP can be obtained as,

$$L = \frac{1}{2}(M + m)\dot{x}^2 + m\dot{x}l\dot{\theta} \cos \theta + \frac{1}{2}ml^2\dot{\theta}^2 - mgl \cos \theta \quad (10)$$

This system has two coordinates as x and θ . Each coordinate has an impact on the system. To analyze the system furthermore, Euler-Lagrangian equation is obtained for both coordinates as follows.

Euler-Lagrangian Equation for x is,

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{x}} \right) - \frac{\partial L}{\partial x} + b\dot{x} = F \quad (11)$$

$$(M + m)\ddot{x} + m\ddot{\theta}l \cos \theta - ml\dot{\theta}^2 \sin \theta + b\dot{x} = F \quad (12)$$

Euler-Lagrangian Equation for θ is,

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} + p\dot{\theta} = 0 \quad (13)$$

$$m\dot{x}l \cos \theta + m\ddot{\theta}l^2 - mgl \sin \theta + p\dot{\theta} = 0 \quad (14)$$

Equation 12 and 14 are 2nd order non-linear differential equations. These equations are linearized in order to get a better understanding about the mathematical aspects and then the linearized model is used to construct the Kalman Filter to optimize the system performance by reducing sensor noises. Linearization process is started by assuming that $\sin \theta \approx \theta$, $\cos \theta \approx 1$ and $\dot{\theta}^2 \approx 0$. This is possible since the angles are measured in radians and they are tiny angles. Then the resulting equations can be rewritten as,

$$\ddot{x} = \frac{F}{(M+m)} - \left(\frac{ml}{(M+m)} \right) \ddot{\theta} - \frac{b}{(M+m)} \dot{x} \quad (15)$$

$$\ddot{\theta} = \left(\frac{g}{l} \right) \theta - \left(\frac{1}{l} \right) \dot{x} - \left(\frac{p}{ml} \right) \dot{\theta} \quad (16)$$

3.2. Space State Model

Now the equations 15 and 16 have been isolated from the trigonometric components. But still the equations are non-linear differential equations. Therefore using space state modelling methods [7], the equations are linearized [14] for a new coordinate system. The new system coordinates are,

$$x_1 = x \quad (17)$$

$$\dot{x}_1 = \dot{x} = x_2 \quad (18)$$

$$x_2 = \dot{x} \quad (19)$$

$$\dot{x}_2 = \ddot{x} \quad (20)$$

$$x_3 = \theta \quad (21)$$

$$\dot{x}_3 = \dot{\theta} = x_4 \quad (22)$$

$$x_4 = \dot{\theta} \quad (23)$$

$$\dot{x}_4 = \ddot{\theta} \quad (24)$$

$$F = u(t) \quad (25)$$

x_1, x_2, x_3 and x_4 are the new coordinate system introduced for a and θ . Therefore, x_1, x_2, x_3 and x_4 are substituted for the equation 15 and 16. Then the equations become,

$$\dot{x}_2 = \frac{u}{(M+m)} - \left(\frac{ml}{(M+m)} \right) \dot{x}_4 - \frac{b}{(M+m)} x_2 \quad (26)$$

$$\dot{x}_4 = \left(\frac{g}{l} \right) x_3 - \left(\frac{1}{l} \right) \dot{x}_2 - \left(\frac{p}{ml} \right) x_4 \quad (27)$$

But still both equations have differential terms in both sides of the equation. Therefore, these two equations are simultaneously solved to separate differential components in both of them. Then the resulting equations become much simpler and they are,

$$\dot{x}_2 = \left(\frac{u}{M} \right) - \left(\frac{b}{M} \right) x_2 - \left(\frac{mg}{M} \right) x_3 + \left(\frac{p}{M} \right) x_4 \quad (28)$$

$$\dot{x}_4 = \left(\frac{g(M+m)}{Ml} \right) x_3 - \left(\frac{u}{Ml} \right) + \left(\frac{b}{Ml} \right) x_2 - \left(\frac{p(M+m)}{Mml} \right) x_4 \quad (29)$$

Now the Space State Model is used to recapitulate all the above equations to interpret them in a mathematical manner. The space state model for linear time invariant system like this is,

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (30)$$

$$y(t) = Cx(t) + Du(t) \quad (31)$$

Where, $x(t)$ – State Vector, $y(t)$ –Output Vector, $u(t)$ - Control/ Input Vector, A – State Matrix, B – Input Matrix, C – Output Matrix and D – Feed Forward Matrix.

Hence, by the equations 17, 21, 28, and 29 the space state model can be modelled as,

$$\dot{x}(t) = \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \left(\frac{-b}{M} \right) & \left(\frac{-mg}{M} \right) & \left(\frac{p}{M} \right) \\ 0 & 0 & 0 & 1 \\ 0 & \left(\frac{b}{Ml} \right) & \left(\frac{(M+m)g}{Ml} \right) & \left(\frac{-p(M+m)}{Mml} \right) \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ \left(\frac{1}{M} \right) \\ 0 \\ \left(\frac{-1}{Ml} \right) \end{bmatrix} u \quad (32)$$

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \left(\frac{-b}{M} \right) & \left(\frac{-mg}{M} \right) & \left(\frac{p}{M} \right) \\ 0 & 0 & 0 & 1 \\ 0 & \left(\frac{b}{Ml} \right) & \left(\frac{(M+m)g}{Ml} \right) & \left(\frac{-p(M+m)}{Mml} \right) \end{bmatrix}$$

And

$$B = \begin{bmatrix} 0 \\ \left(\frac{1}{M} \right) \\ 0 \\ \left(\frac{-1}{Ml} \right) \end{bmatrix}$$

The output matrix is,

$$y(t) = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x \\ \theta \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u \quad (33)$$

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

3.3. Stability, Controllability and Observability

Using the space state model of the system, the system STABILITY, CONTROLLABILITY and OBSERVABILITY is tested. This method is useful when a control system is constructed for a system.

If the system is stable under open loop condition, then there is no need for a control system. This condition is tested using the matrix A of the space state system. If the system is stable, Eigen values of the matrix A should be less than or equal to zero. For this system, the condition for stability is violated. Therefore the system is unstable and a control system is needed.

Then the controllability of the system is tested. This can be done by using matrix A and B. if the rank of controllability of the matrices A and B is a full rank, the system is consider to be a controllable system. Otherwise the system cannot be controlled using a control system. For IP system in this research, this rank is a full rank (rank is equal to the number of states). This test implies that the system is controllable.

Sometimes all the parameters in the system wouldn't be able to measure directly. In such cases the system parameters are given or defined inside the system. This is only possible if the system is observable. Therefore, the observability is tested using the matrices A and C of the system. If the rank of observability of the matrices A and C is a full rank, the system is said to be an observable system. Here the inverted pendulum is an observable system.

3.4. Kalman Filter for Noise Reduction

This system includes two digital rotary encoders for tracking the linear and angular motions of the system. But these signal consist of some noises and they have to be removed or reduced in order to construct a better control system. Kalman

filter is widely used with control algorithms to reduce noises in signals in the fields of engineering and physics. In this case Kalman filter is used as a predictor corrector algorithm to get rid of noises in the system. Kalman filter gives the best prediction about the next state of the system. Equations of the Kalman filter is given below.

$$x_{k+1|k} = F_k x_k + \nabla_k u_k \quad (34)$$

$$P_{k+1|k} = F_k P_k F_k^T + Q_k \quad (35)$$

$$K_{k+1} = \frac{P_{k+1|k} H_{k+1}^T}{H_{k+1} P_{k+1|k} H_{k+1}^T + R_{k+1}} \quad (36)$$

$$x_{k+1} = x_{k+1|k} + K_{k+1} (z_k - H_k x_{k+1|k}) \quad (37)$$

$$P_{k+1} = (I - K_{k+1} H_{k+1}) P_{k+1|k} \quad (38)$$

In this mathematical model of the IP, the space state system is an analog system. But the control algorithm of the system is a discrete time model (Digital Systems are discrete time systems). Therefore the system has to be converted in to its discrete time model in order to implement on the microcontroller. This is also applied for the Kalman filter algorithm. Therefore the space state system of the IP is converted into its new discrete time model and hence the matrix A has become F, B has become ∇ and C has become H. Q is the process noise covariance matrix and the R is the measurement noise covariance matrix. The notation $k+1|k$ represent that the $k+1$ state is given by the previous state k . this Kalman filter process is performed before the sensor signals are fed to the control algorithm.

3.5. Optimized PID Controller Design for Inverted Pendulum

As the control algorithm, PID controller is used. Here two PID controllers are needed for the system to get the best performances. The reason is in this IP, two parameters are taken as linear position and the angle of the pendulum. Classical PID controller has been developed for the analog control systems. The well-known PID controller $p(t)$ equation is shown below.

$$p(t) = k_p \times e(t) + k_i \times \int_0^t e(t) dt + k_d \times \frac{de(t)}{dt} \quad (39)$$

In the equation 39, k_p , k_i and k_d are the proportional, integral and derivative gain constants respectively. $e(t)$ is the error (or the deviation from the set point).

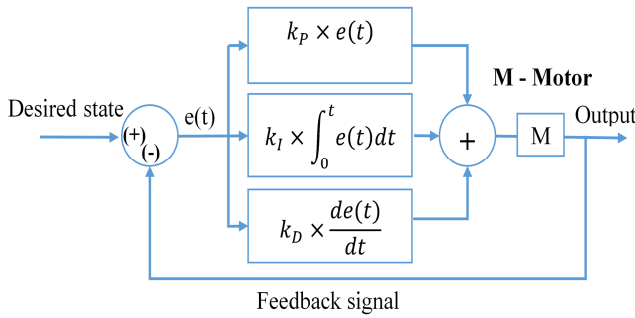


Figure 2. Basic block diagram of PID controller.

This classical PID controller (see Figure 2) is an analog system. Therefore, this controller has to be converted into its discrete time form to implement on the microcontroller. The equation 39 can be re-defined for infinitesimal time period as,

$$p(t) = k_p \times e(n) + k_I \times \sum_0^n e(n)dt + k_D \times \frac{e(n) - e(n-1)}{dt} \quad (40)$$

Where $e(n)$ is the error at n^{th} state. If,

$$K_p = k_p, K_I = k_I \times dt \text{ And } K_D = \frac{k_D}{dt}$$

$$p(t) = K_p \times e(n) + K_I \times \sum_0^n e(n) + K_D \times \{e(n) - e(n - 1)\} \quad (41)$$

Then the final PID control equation can be constructed as in the equation 41. In this equation integrations and derivations have become simple additions and subtractions. Therefore it is less complex to implement on the microcontroller.

But here two PID controllers have to be used. This two PID controllers can be construct as shown in the Figure 3.

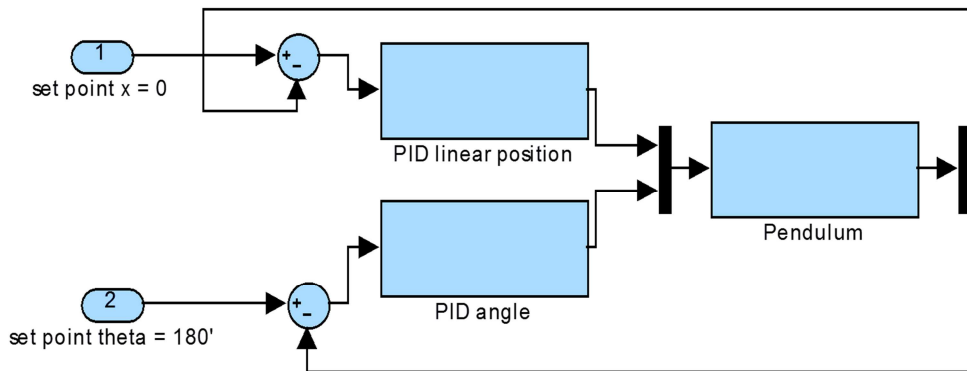


Figure 3. Two PID controllers are combined together.

4. Implementation Procedure

4.1. Construction of the Mechanical & Electronics System

At the beginning, weights of necessary parts of the pendulum was measured. Then the mechanical system of the pendulum was constructed. In the constructed IP, two digital rotary encoders were used to track the angular and linear positions of the pendulum. The encoders have a resolution of 2400 pulses per 2π angle. Therefore precision of the sensor was reliable. The pendulum was pivoted to the rotary encoder which was installed on the cart. The cart consist of two bearings which reduces the frictional forces. Two rails were used to set up the cart on them.

As the electronic control system, ATMEGA 2560 microcontroller [15] was selected. This microcontroller was selected due to its six external interrupt capabilities and the 16MHz of clock speed. It was very useful for this project because two rotary encoders needed four external interrupt sensors to track the positions with a higher accuracy. To control the DC motor, H-bridge motor controller module was

used. Motors are controlled through PWM (Pulse Width Modulation) Signals. Encoder data were taken from the real time data acquisition system implemented on the microcontroller program via serial communication to the computer. These data were used to analyze the system behavior with various PID gain parameters. Figure 4 shows the completed pendulum system.



Figure 4. Inverted Pendulum after construction.

4.2. Development of the Control Algorithm

To develop the control algorithm, C language was used. C language has several advantages over most of the other computer languages. Especially when we are dealing with embedded technologies, C language is used due to its mesmerizing performances. The platform used for the

programming was Arduino IDE which is an open source embedded programming package. Following shows (See Figure 5) the flow chart of the developed computer algorithm on the microcontroller.

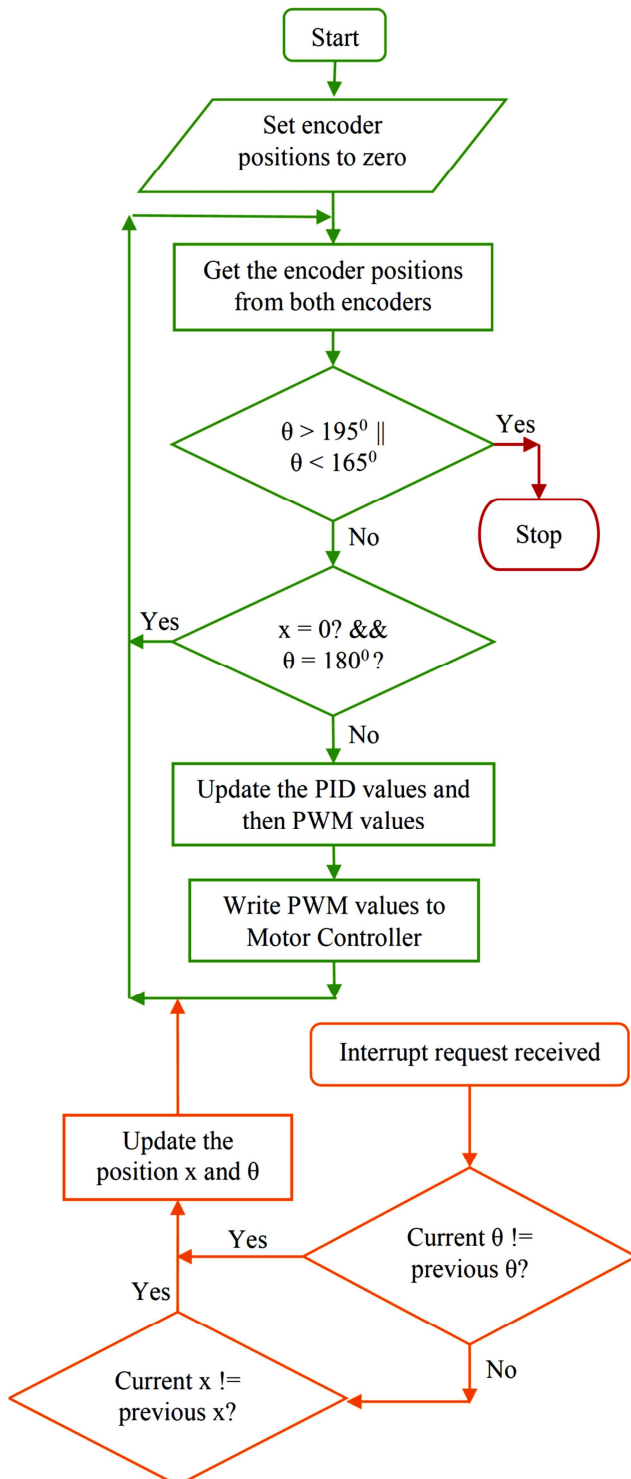


Figure 5. Flow Chart of the Control algorithm.

Then the PID algorithm with the Kalman state predictor was implemented on the program and then the program was burnt into the microcontroller.

4.3. Tuning Of PID Gain Constants

Tuning of the PID gain constants was the difficult part here. The reason is this process was done through trial and error approach. As the first gain constant, proportional gain constant of the angular position PID controller, K_p was adjusted and output of the system was tested. This adjustment process had to be performed until the system get comfortable. After the K_p gain was tuned, K_i gain was chosen as the next parameter. This was selected due to a reason. After the K_p gain was properly tuned, the IP had a lot of oscillations. These oscillations were gradually increased with the time. To overcome this problem the best selection is the K_i gain because, when the error is increased, integration part of the PID controller dominates the control effect on the system. Then the K_i gain constant was adjusted until the system get stable. But there were some oscillations in the IP system. To reduce those oscillations, K_d gain was adjusted and finally the system got stable. After these adjustments the IP was almost stable.

But the problem was the pendulum could be balanced anywhere in between the rails. But it was necessary to keep the pendulum at the middle of the rails. So the next task was to tune the PID parameters in the linear position PID controller. Again the tuning process was started with the K_p of the linear position PID controller. Then the system was balanced at the middle of the rails. But there were some oscillations and it was an unwanted disturbance to the pendulum. Those oscillations were very close to the middle point of the rails. In this case the best choice is to select the K_d gain. The reason was at this point the error was not gradually increased with time. Instead the error was slightly smaller than the previous oscillations and most likely to be a constant error. Therefore the effect of the K_i gain is very small on the system. So the best gain parameter was the K_d gain. This was also affected to the system oscillations, hence damped the oscillations of the system. After adjusting the K_d parameter, the system was almost stable and there were only few oscillations around the set points ($x = 0$ and $\theta = 180^\circ$) of the IP system (see Figure 5).

5. Results and Observations

5.1. Final Outcome of the IP System

After all necessary adjustments, the IP was balanced autonomously by itself and it could manage external disturbances automatically. The balanced IP is shown in Figure 6. This was the main objective of the research. The analytical study of the IP was done by using the data taken through the real time data acquisition system to the computer. This analyze was performed considering a specific time

period in which all the behaviors of the IP corresponding to each PID parameter could be analyzed.

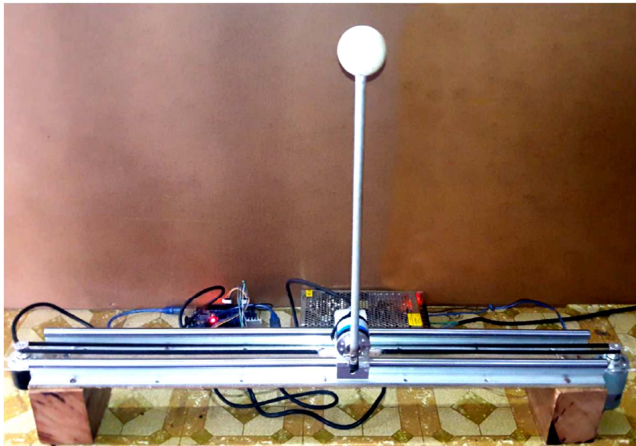


Figure 6. Balanced Inverted Pendulum based on PID Controller.

5.2. Analytical Representations of the Data of the Inverted Pendulum

In this scenario, mainly consider about the data coming from the angle encoder. This is done because the stability of the system technically depend on the angle of the pendulum. Inside the microcontroller program, a terminator point (see Figure 5) had to be defied to protect the system by damaging itself due to highly unstable situations. These unstable behaviors are represented in the following graphs along with the final stable desired state. The data acquisition system was designed to send the raw encoder positions instead of the converted angles. This is possible because of the proportional relationship between angle and the encoder values. Therefore the corresponding value of the encoder for π angle (set point for the angle PID) is 1200 pulses.

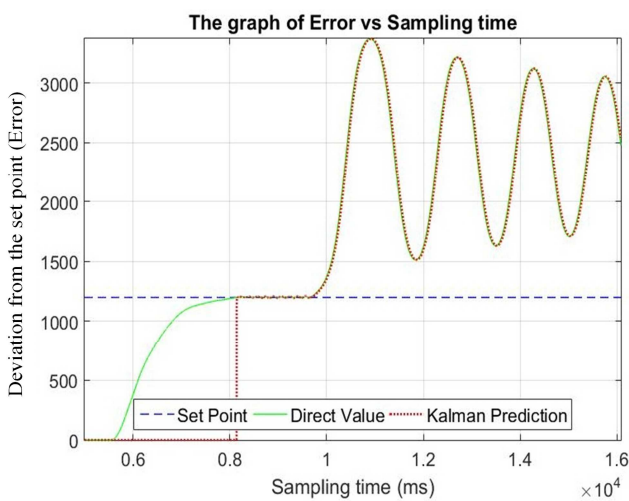


Figure 7. The graph of Error vs. sampling time when $K_P = 18$, $K_I = 0$ & $K_D = 0$ and all linear PID constant are zero.

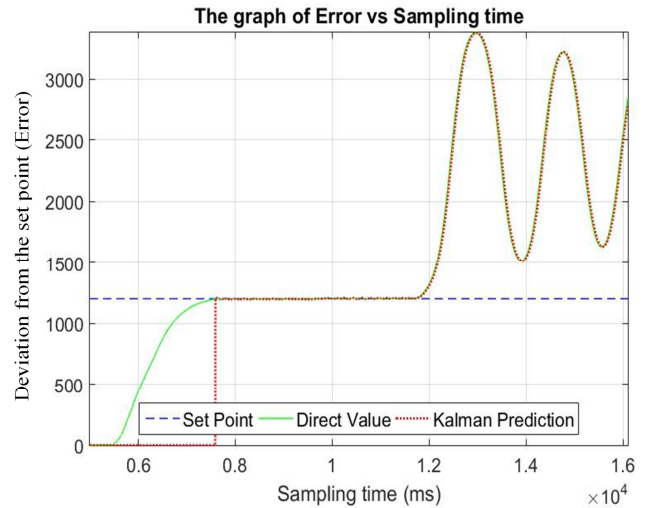


Figure 8. The graph of Error vs. sampling time when $K_P = 18$, $K_I = 0$ & $K_D = 50$ and all linear PID constant are zero.

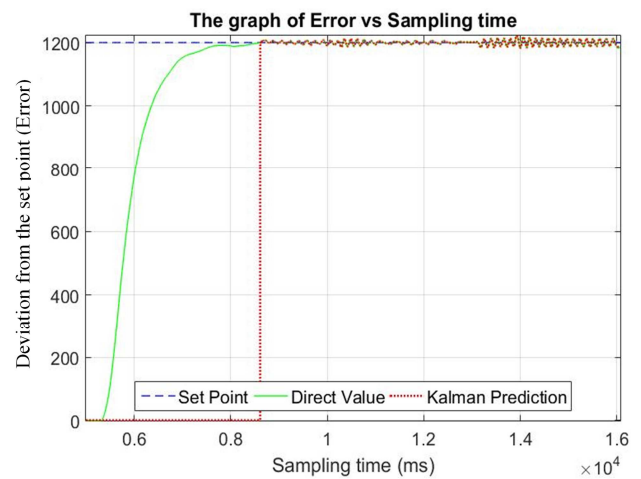


Figure 9. The graph of Error vs. sampling time when $K_P = 18$, $K_I = 0.05$ & $K_D = 50$ and all linear PID constant are zero.

In the above graphs (Figure 7 to 9), only the angle PID controller was involved. According to the graph in Figure 7, K_P of angle PID controller equals to 18. At this point it was stable for a moment but after few milliseconds the system got unstable and collapsed.

Then the K_D was introduced to the PID controller (Figure 8) and even with effect of derivative gain K_D , the system could not keep its balance. But in this case, it was stable for a very small time period which was longer than the previous case (Figure 7).

Finally, all gain constants including K_I of the angle PID were set to their optimum values and the system was very stable (Figure 9). At this point the pendulum was balanced at the vertical upright position. However, the balanced position on the rails was not accurate. Therefore, the linear position PID controller had to be involved for balancing the pendulum at the middle of the rails.

Figure 10 shows the final balanced inverted pendulum with the effect of both PID controllers. Here the pendulum was stable around the middle set point of the rails. However, some small oscillations were there. But the system gave the expected behavior by reacting to external disturbances and keeping pendulum vertically upright. In this stage, even for an intentionally made impacts on the pendulum, the pendulum always tried to keep the pendulum upright while trying to reach to the middle of the rails.

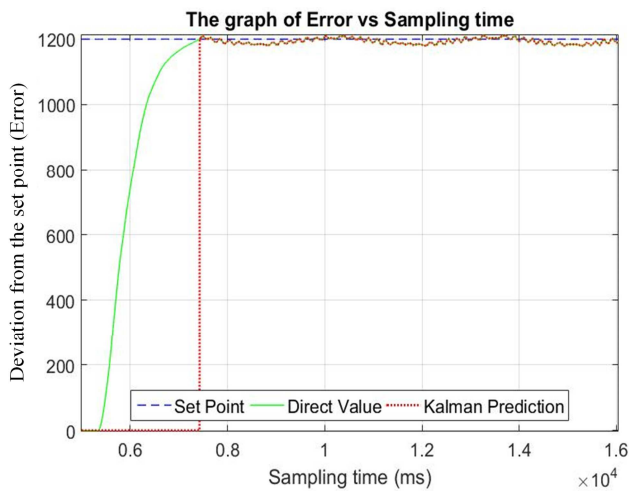


Figure 10. The graph of Error vs. sampling time when $K_p = 18$, $K_i = 0.05$ & $K_D = 50$ and Linear $K_p = 0.002$, Linear $K_i = 0.0005$.

This pendulum can also be used to contrast the PID controller effect on the pendulum stability. This experiment was performed on the balanced pendulum system by introducing new values for K_p of the angle PID controller while keeping all the other PID gain parameters unchanged. Here, the effect of the K_p gain is demonstrated in the Figure 11 and 12.

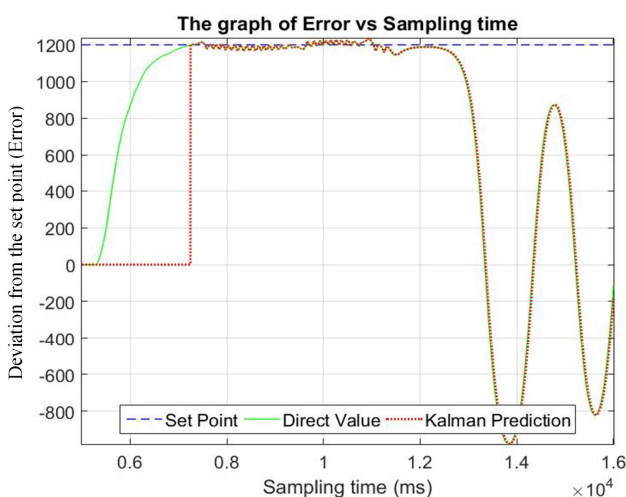


Figure 11. The graph of Error vs. sampling time when $K_{p(\text{angle})} = 1$.

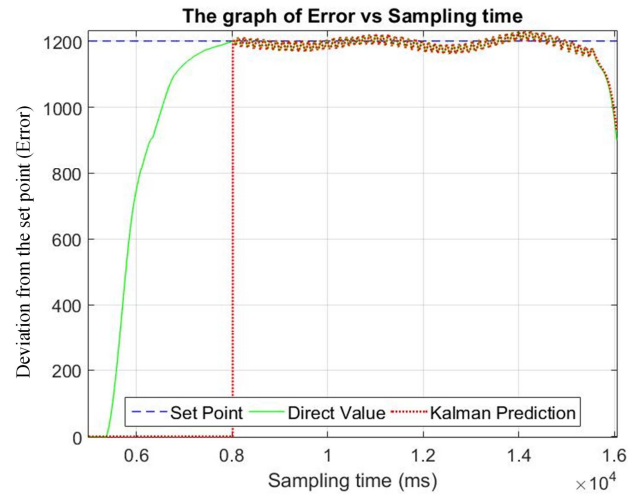


Figure 12. The graph of Error vs. sampling time when $K_{p(\text{angle})} = 100$.

In the Figure 11 and Figure 12, the $K_{p(\text{angle})}$ was set to 1 and 100 respectively. But for both values the system was not stable as in the Figure 10 with $K_{p(\text{angle})} = 18$. This result can be used to clearly distinguish the behavior of the pendulum with high K_p values and low K_p values. According to the Figure 12, even though the K_p is high the system could not keep its stability. But it has given far better performance than the low K_p gain (Figure 11). However, both systems have eventually collapsed by losing the stability of the pendulum. This implies that the correct gain parameters can be obtained by selecting two initial guesses like one as small value and the other as a larger value. This will help to reduce the number of trials and narrow down the value range for corresponding gain parameter.

6. Discussions

Under this study of the inverted pendulum, mainly focused on the PID controller implementation on a microcontroller program. But this kind of control algorithms need higher processing power to execute the program. The reason is error optimizations and state predictors take a long time to execute due to their mathematical calculations with floating point numbers. But this complexity can be overcome by limiting the number of decimal points up to two in the values by using round off methods. Especially, the C language also contributes to the execution speed.

Hardcore part of the study was the tuning process of the PID parameters of two PID controllers of the program. But the complexity of tuning process can be decreased by selecting PID gain constants wisely. Here the idea of this wisely chosen values is, for an example, the tuning process can be started with $K_p = 1$ and if the system seems to be stable then a larger value can be selected and checked whether the system is stable or not. If the system seems to be stable, which implies the best value for this gain parameter definitely lie in between 1 and the

second-guessed larger value. However, it is not necessary to start with 1. But choosing one as the initial guess, there is a higher possibility to get a better idea about the tuning process. If the system is unstable at $K_p = 1$ and even higher values larger than 1, which implies that the K_p might be less than 1. This method is not limited for proportional gain. It can also be applied for the other two gain constants too.

The Kalman filter was used to optimize the error in the system. Data coming from the rotary encoders are not analog signals. Which means they are discrete values. Therefore, the Kalman predictor corrector algorithm used in this program increases the smoothness of the data which is fed to the PID controller. Due to this reason most of the unwanted oscillations could be decreased in the inverted pendulum.

7. Conclusions

The proposed model for balancing of an inverted pendulum was based on the PID controllers. By using Kalman filter along with PID controllers, unnecessary noise signals and discretization of the data can be reduced and the errors of the system can be optimized to obtain the desired state from the system. This microcontroller-based control algorithm gave the desired stability from the system and the inverted pendulum could manage most of the intentionally made disturbances too. Effectiveness and the efficiency of the developed PID controller was far better than expected. The concept of this research can be extended up to three dimensional inverted pendulums. PID based control algorithms are cheap but reliable. Therefore, this controlling method can be used with self-balancing robots, unmanned airplanes (quad copters) and even for missile guidance systems with lower computational power due to the simplicity of the PID controllers. Especially, PID based controllers are recommended to use with low power computer systems such as microcontrollers.

Data Availability

The data that support the findings of this study are openly available in Inverted-Pendulum-Data at <https://github.com/mbabeysekera/Inverted-Pendulum-Data>.

Abbreviations

DC:	Direct Current
GA:	Genetic Algorithm
IP:	Inverted Pendulum
IDE:	Integrated Development Environment
PID:	Proportional, Integral and Derivative
UAV:	Unmanned Aerial Vehicle

Conflicts of Interest

All the authors do not have any possible conflicts of interest.

Acknowledgements

The authors would like to thank the University of Sri Jayewardenepura (Grant No ASP/01/RE/SCI/2017/13) for the funding support.

References

- [1] N. M. A. Ghani, F. Naim and T. P. Yon, "Two Wheels Balancing Robot with Line Following," *International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering*, vol. 5, no. 7, pp. 1402, 1405, 2011.
- [2] B. Kada and Y. Ghazzawi, "Robust PID Controller Design for an UAV Flight Control System," *WCECS*, vol. II, 2011.
- [3] V. Balaji, M. Balaji, M. Chandrasekaran, A. khand and I. Elamvazuthi, "Optimization of PID Control for High Speed Line Tracking Robots," *Procedia Computer Science*, no. 76, pp. 147-152, 2015.
- [4] M. Nour, J. Ooi and K. Y. Chan, "Fuzzy Logic Control vs. Conventional PID Control of an Inverted Pendulum Robot," *IEEE*, pp. 1-6, 2007.
- [5] T. Sugihara, Y. Nakamura and H. Inoue, "Realtime Humanoid Motion Generation through ZMP Manipulation based on Inverted Pendulum Control," *CREST Japan Science and Technology Corporation (JST)*, 2007.
- [6] J.-J. Wang, "Simulation studies of inverted pendulum based on PID controllers," *Elsevier*, pp. 440, 441, 2010.
- [7] M. B. Rhudy, R. A. Salguero and K. Holappa, "A KALMAN FILTERING TUTORIAL FOR UNDERGRADUATE STUDENTS," *International Journal of Computer Science & Engineering Survey*, vol. 8, no. 1, pp. 1-9, 2017.
- [8] J. Riggs, *Chemical Process Control*, Lubbock: Ferret Publishing, 2001.
- [9] J. Charais, R. Lourens and Microchip Technology Inc., "Software PID Control of an Inverted Pendulum Using the PIC16F684," Microchip Technology Inc., California, 2004.
- [10] M. Moghaddas, M. RezaDastranj, N. Changizi and N. Khoori, "Design of Optimal PID Controller for Inverted Pendulum," *International Journal of Innovation, Management and Technology*, vol. 3, pp. 440, 441, 2012.
- [11] M. Moghavvemi, A. L. Salih, H. A. F. Mohamed and K. S. Gaeid, "Flight PID controller design for a UAV quadrotor," *Scientific research and essays*, vol. 5, no. 23, p. 3660, 2010.
- [12] G. Welch and G. Bishop, "An Introduction to the Kalman Filter," 2007. [Online]. Available: <http://www.es.unc.edu/~welch/kalman/kalmanIntro.html>. [Accessed 15 April 2017].
- [13] D. Simon, "Embedded Systems Programming," 2003. [Online]. Available: <http://www.controleng.com/article/CA325983.html>. [Accessed 11 April 2017].

[14] T. M. Fowler, LTI System and Control Theory, 2007.

Available: <https://store.arduino.cc/usa/arduino-mega-2560-rev3-956>. [Accessed 6 May 2017].

[15] Arduino, "Mega 2560 Rev3 - Boards & Modules," [Online].