Jurnal Teknologi, 50(D) Jun 2009: 93–108

A STUDY OF DIFFERENT CONTROLLER STRATEGIES FOR A BALL AND BEAM SYSTEM

HERMAN WAHID[1] & MOHD FUA'AD RAHMAT[2]

Abstract. This paper presents several control approaches that consist of a conventional controller, modern controller and intelligent controller and the performance of those controllers that were employed in a ball and beam system. The ball and beam system is one of the examples of a nonlinear and unstable control system. It consists of rigid beam which is free to rotate in the vertical plane at the pivot, with a solid ball rolling along the beam. The control problem is to position the ball at a desired point on the beam by controlling the motor voltage as the input of the system. Scope of investigation is on proportional (P) and proportional-derivative (PD) as the conventional controller, pole placement for the modern controller, and fuzzy logic for the intelligent controller. The mathematical modeling is linearised in order to be used with the linear control and followed by designing the entire system and simulation in MATLAB. Each controller performance is analyzed and compared using the step response. An appropriate graphic user interface (GUI) has been developed to view the animation of the ball and beam system.

Keywords: Ball and beam; modeling; PD controller; pole placement; fuzzy logic

Abstrak. Artikel ini membincangkan beberapa pendekatan untuk sistem kawalan yang terdiri daripada pengawal konvensional, pengawal moden dan pengawal pintar, dan akan menguji kecekapan setiap pengawal pada sistem bola dan pengimbang. Sistem bola dan pengimbang adalah satu contoh sistem tak linear serta tidak stabil. Ia terdiri daripada kayu pengimbang yang bebas berputar pada paksi, dengan sebiji bola bergerak di sepanjang batang pengimbang. Objektif sistem kawalan ialah untuk meletakkan bola pada kedudukan tertentu di atas batang pengimbang dengan mengawal voltan motor sebagai input kepada sistem. Artikel ini mengkaji pengawal P dan pengawal PD sebagai kawalan konvensional, perletakan kutub sebagai kawalan moden, dan logik kabur sebagai kawalan pintar. Permodelan matematik untuk sistem bola dan pengimbang dihuraikan yang melibatkan proses penglinearan model agar dapat digunakan dengan pengawal linear. Kemudian, semua pengawal tadi direka bentuk dan disimulasikan dengan menggunakan program MATLAB. Kecekapan setiap pengawal dianalisis berdasarkan beberapa ciri sambutan langkah. Pengantaramuka Pengguna Bergrafik (GUI) yang sesuai telah dibangunkan bagi memberi gambaran animasi untuk sistem bola dan pengimbang.

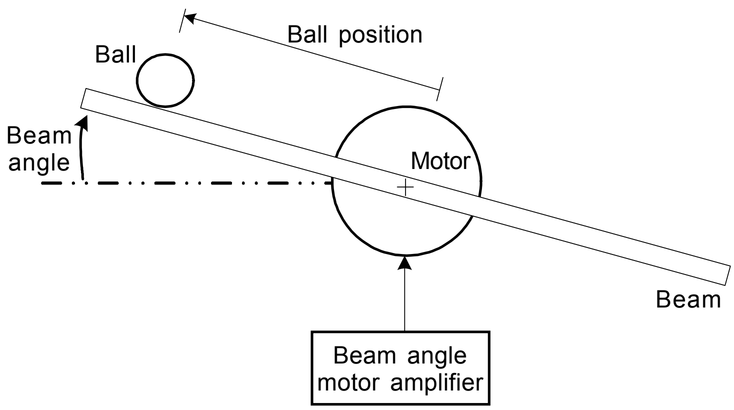
Kata kunci: Sistem bola dan pengimbang; permodelan; pengawal PD; perletakan kutub; logik kabur

[1&2] Control and Instrumentation Engineering Department, Faculty of Electrical Engineering, Universiti Teknologi Malaysia, 81310 Skudai, Johor Darul Takzim
Tel.: 07-5535317. Email: [1]herman@fke.utm.my, [2]fuaad@fke.utm.my

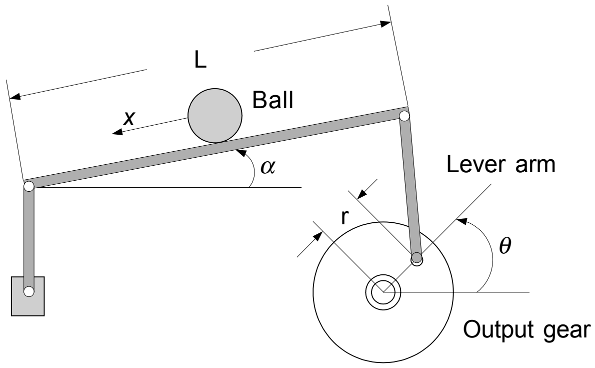
1.0 INRODUCTION

The ball and beam system is a non linear and unstable system, thus providing a challenge to the control engineers or researchers. Basically, there are two types of configuration for ball and beam system. The first configuration is shown in Figure 1 (a), which illustrates that the beam is supported in the middle, and rotates against its central axis. Most ball and beam systems use this type of configuration such as Hirsch (1999) [7], Rosales (2004) [8] and Lieberman (2004) [9]. This type of configuration is normally called as 'Ball and Beam Balancer'. The advantage of this form is that it is easy to build, and the mathematical model is relatively simple.

The other configuration is constructed with the beam and is supported on the both sides by two level arms. One of level arms acted as the pivot, and the other is coupled to motor output gear. The disadvantage is that more consideration of the mechanical parts, which meant adding difficulties in deriving a mathematical model. This type of



(a)



(b)

Figure 1 (a) Beam supported at the centre, (b) Beam supported at both side configuration is called as 'Ball and Beam Module'. The 'Quanser' ball and beam system uses this configuration for its commercial product as illustrated in Figure 1(b), (Quanser, 2005) [10]. The advantage of this system is that relatively small motor can be used due to the existing of gear box. This type of configuration is employed in this project.

The aim of ball and beam system is to position the rolling ball at the desired position through its acceleration. Thus, it will imply the presence of the two integrators plus the dynamical properties of the beam result in a difficult open loop unstable control problem, which is non-linear system [1]. The linear feedback control such as PID control can be applied and the stability analysis is based on linear state-space model or transfer function. Besides, it can also be solved by nonlinear controllers but this controller is very complex for real application. Furthermore, some intelligent controllers for ball and beam can also be found, such as fuzzy control [3], sliding mode control [4], neural control [5], etc.

This paper describes the design of a few types of controller for ball and beam system that consist of conventional controller, modern controller and intelligent controller type. P and PD controller represent the conventional controller, state space pole placement controller as a modern controller, and Fuzzy Logic that represent the intelligence controller. An analysis of the performance is carried out to the entire controllers, so that the best performance can be identified. Finally, a suitable general user interface (GUI) is developed to view the animation of ball and beam system.

2.0 MODELLING THE BALL AND BEAM SYSTEM

As shown in Figure 1(b), a ball is placed on a beam where it is free to roll at horizontal axis along the length of the beam. A lever arm is attached to the beam at one end and a servo gear at the other. The servo gear turns by an angle θ, and the lever changes the angle of the beam by α. The force that accelerates the ball as it rolls on the beam come from the component of gravity that acts parallel to the beam. The ball actually accelerates along the beam by rolling, but we can simplify the derivation by assuming that the ball is sliding without friction along the beam. The mathematical modeling of ball and beam system consists of DC servomotor dynamic, alpha-theta relation, and ball on the beam dynamic.

By using Lagrange method, we will obtain ball and beam dynamic that has been described by Hauser [2] as given in equation (1),

$$0 = \left(\frac{J_b}{R^2} + m \right) \ddot{r} + mg \sin \alpha - mr\dot{\alpha}^2 \quad (1)$$

where J_b is the moment inertia of the ball, R is radius of the solid ball, $\ddot{r}$ is acceleration of the ball, m is mass of the ball, g is gravitational constant, α is beam angle and $\dot{\alpha}$ is angular velocity of the beam angle. The derivation of equation (1) can be illustrated

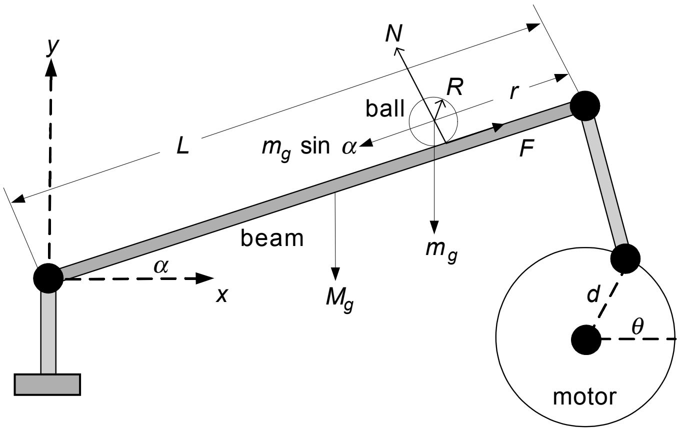


Figure 2 Force acting on the ball and beam system

from Figure 2. Linearization of equation (1) can be estimated when the system near to the stable point. At this point $\dot{\alpha} \approx 0$, where $\dot{\alpha}$ is the angular velocity of the beam angle. Therefore, the linear approximation of the system is given by differential equation in equation (2).

$$\ddot{r} = \frac{-mg \sin \alpha}{\left(\frac{J_b}{R^2} + m\right)} \tag{2}$$

Note that the nominal value J_b of the solid ball is given by $J_b = 2mR^2/5$ kgm^2. Substituting J_b into equation (2) will give,

$$\ddot{r} = -\frac{5}{7} g \sin \alpha \tag{3}$$

Since α angle is small when near to stable point, thus sin $\alpha \approx \alpha$. The transfer function for ball and beam dynamic is given in equation (4), where $\Re$ = ball position, α = beam angle, g = gravitational acceleration.

$$\text{Transfer function 1: } \frac{\Re(s)}{\alpha(s)} = -\frac{5g}{7}\frac{1}{s^2} \tag{4}$$

The beam angle (α) can be related with motor gear angle (θ) by approximate linear equation $\alpha L = \theta d$, where d = lever arm offset and L= beam length. Substitute L = 41.7 cm and d = 2.54 cm will give another transfer function as equation (5),

$$\text{Transfer function 2: } \frac{\alpha(s)}{\theta(s)} \cong \frac{1}{16} \tag{5}$$

Based on voltage Kirchhoff's Law, the electrical equation of the motor is given by,

$$V_{in} = IR_a + K_m\dot{\theta}_m + L_a\frac{di}{dt} \tag{6}$$

where V_{in} = input voltage (V), I = armature current (A), R_a = armature resistance = 9 Ω, K_m = motor torque constant = 0.0075 Nm/A, $\theta_m = \omega$ = angular velocity of output (rad/sec) and L_a = inductance in armature coil (mH). To simplify the motor model, the effect of the inductance L_a can be ignored because L_a contributes a very small effect in low speed application. Thus, the model will become $V_{in} = IR_a + K_m\dot{\theta}_m$. The torque produced at the motor shaft is given by,

$$T_m = K_m I \tag{7}$$

Assume that the load torque is the same as the produced torque,

$$T_l = T_m = \frac{\left(J_1\ddot{\theta} + B\dot{\theta}\right)}{K_g} \tag{8}$$

where $\dot{\theta} = \frac{\theta_m}{K_g}, \dot{\theta}$ = angular velocity of load (rad/sec), J_1 = total load inertia = 7.35×10^{-4} Nms^2/rad, B = total load friction = 1.6×1^{-3} Nms/rad and Kg = gear ratio = 75. From equation 6, 7 and 8, solve for $\frac{\theta(s)}{V_{in}(s)}$, we get transfer function for servomotor model,

$$\frac{\theta(s)}{V_{in}(s)} = \frac{K_m K_g}{s^2 R_a J_1 + s\left[R_a B + \left(K_m K_g\right)^2\right]} = \frac{K_m K_g / R_a J_1}{s^2 + s\left[\frac{B}{J_1} + \frac{\left(K_m K_g\right)^2}{R_a J_1}\right]}$$

Transfer function 3: $$\frac{\theta(s)}{V_{in}(s)} = \frac{a_m}{s^2 + b_m s} \tag{9}$$

In differential equation, a servomotor model can be written as,

$$\ddot{\theta} = -50\dot{\theta} + 85V_{in} \tag{10}$$

The linearized system equations can also be represented in state-space form. This can be done by selecting the ball's position (r) and velocity ($\dot{r}$) from equation (4) as the state variables. Besides, we select motor gear angle (θ) and motor angular velocity ($\dot{\theta}$) from equation (10) as another state variables, and the motor input voltage (V_{in}) as the input. The state-space representation is shown in equation (11). The mathematical model in state space form is used to design the pole placement controller in the next section.

$$\begin{bmatrix} \dot{r} \\ \ddot{r} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -\dfrac{5g}{7} * \dfrac{1}{16} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -50 \end{bmatrix} \begin{bmatrix} r \\ \dot{r} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 85 \end{bmatrix} V_{in} \tag{11}$$

3.0 CONTROLLER DESIGN

The control problem is to design a controller which computes the applied voltage V_{in} for the motor to move the ball in such a way that the actual position of the ball reaches the desired position. The motor is controlled to produce the desired α (beam angle), however it should be noted that α is controlled by the angle at the output of the servomotor plant (θ angle). The simplest control strategy is the 1-DOF topology shown in Figure 3, where the plant is treated as the cascade connection of transfer function 1, 2 and 3 in Equations (4), (5) and (9). Although it is possible to design the controller $C(s)$ such that the closed-loop system is stable, however, the multiple integrator exists in the plant that contributes –270° phase lag to the loop gain making it difficult to obtain "good" closed-loop performance.

Before making any decision of controller design, a few design specifications have been set. In this design, we only take two considerations to be met which are settling time less than 3 second and percentage of overshoot is less than 3%. The 3 second is chosen to determine the effectiveness of the designed controllers in terms of fast response. Whereas, the low overshoot is required to avoid the ball run out of the beam especially at the edge points.

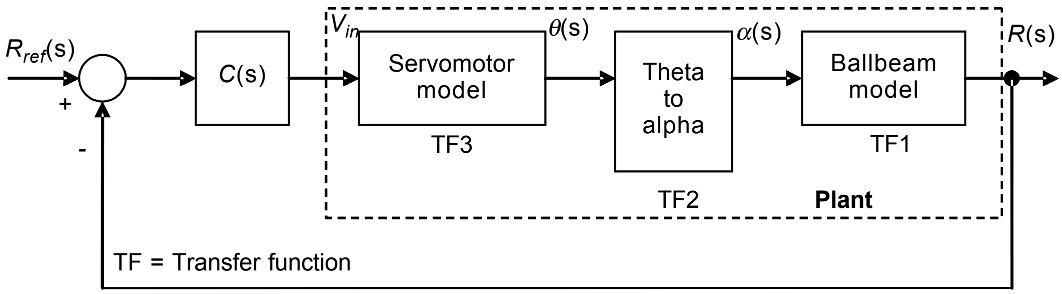


Figure 3 1-DOF control strategy for the linearized model

3.1 P and PD Controller Design

To make the controller design become easier and realizable, the whole system is separated into two feedback loops as shown in Figure 4. The purpose of the inner loop is to control the motor gear angle position. PID controller 1 ($C_1(s)$) should be designed so that gear angle (θ) tracks the reference signal (θ_{ref}). The outer loop uses the inner

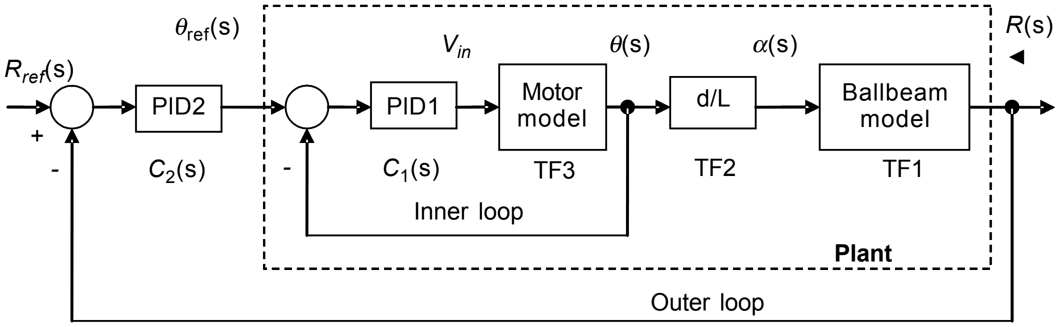


Figure 4 2-DOF control strategy for the linearized model

feedback loop to control the ball position. Therefore the inner loop definitely must be designed before the outer loop.

For the inner loop, PD controller is selected instead of PID because servomotor model is a second order system, thus PID controller will change the second order system to third order system which is quite hard to control whereby PD controller will preserve its second order. Thus, the equation for PD controller for inner loop is $C_1(s) = K_p + K_d s$, where K_p is proportional gain and K_d is derivative gain. By using Ziegler Nichols's method, PD parameter has been tuned to be $K_p = 5$ and $K_d = 0.1$.

The outer loop also consists of second order plant (with inner loop feedback is already compensated). Therefore the PD controller is supposing the most suitable for the outer loop feedback. If integrator is used, when the error is zero, the control variable (V_{in}) will not be zero because of the integrating offset. This means that if the error is zero, the ball is supposed to stop because the actuation voltage will be zero but because integrator is used, a zero error signal will produce a non-zero actuation signal (offset) and drive motor, and thus the ball will not stop. But, for the study purpose and comparison analysis, P and PD controller will be discussed here. The variation of that controller showed in equation (12) and (13).

$$\text{P controller:} \quad C_2(s) = K_p \tag{12}$$

$$\text{PD controller:} \quad C_2(s) = K_p + K_d s \tag{13}$$

3.2 Pole Placement Controller Design

A modern controller can be represented by a state space equation. There are various types of state space controller implementation such as full state feedback control, observer control, optimal control and etc. This paper analyzes a full state feedback controller which is using the pole placement approach. The schematic of a full-state feedback system is shown in Figure 5(a), [6]. From this figure $r_{ref}(t)$ is desired position, $r(t)$ is output position and K is full-state feedback gain.

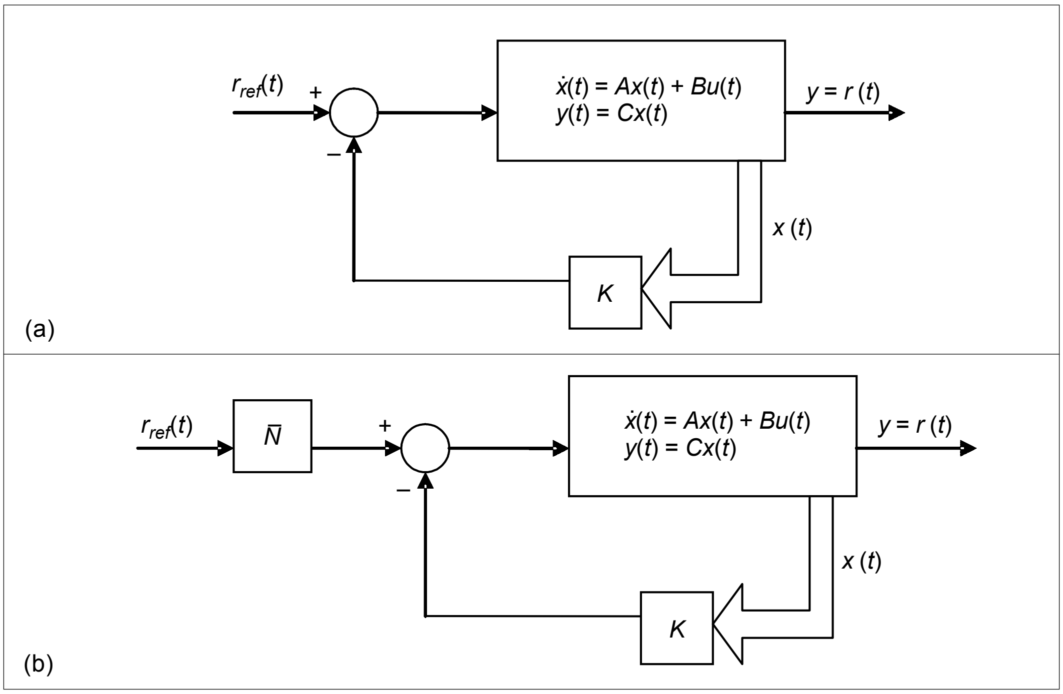


Figure 5 (a) State feedback control configuration, (b) State feedback control configuration with reference input

The characteristic equation for the closed-loop system is given by the determinant of [sI-(A-BK)]. For our system the A and B*K matrices are both 4×4. Hence, there should be four poles for our system. For the pole placement strategy we can move these poles anywhere we want.

For our design we desire an overshoot of less than 3% which corresponds to a damping ratio (ζ) of 0.8. On a root locus this criterion is represented as a 36.8 degree (i.e. $\zeta = \cos\,\theta$) line emanating from the origin and extending out into the left-half plane. We want to place our poles on or beneath this line. Our next criterion is a settling time less than 3 seconds, which corresponds to a sigma = 4.6/Ts = 4.6/3 = 1.53, represented by a vertical line at –1.53 on the root locus. Anything beyond this line in the left-half plane is a suitable place for our poles. Therefore we will place our poles at –2 + 1.5i and –2 – 1.5i. We will place the other poles far to the left for now, so that they will not affect the response too much. Thus we place the third pole at –20 and fourth pole at –80 respectively. We will use the 'place' command in Matlab to find the controller matrix gain, K.

$$K = \text{place}\,(A, B, [p1, p2, p3, p4]) \tag{14}$$

Basically by using State Feedback method only, we can found a very large steady state error. Therefore, in order to eliminate the steady-state error we have to enhance the design by adding a constant gain $\bar{N}$ after the reference ($R_{ref}(t)$). The schematic in Figure 5(b) shows this configuration. The value $\bar{N}$ can be found by using the user-defined function rscale.

$$\text{Nbar} = \text{rscale}\ (A, B, C, D, K) \tag{15}$$

Finally we will get the new matrix equation for A, B, C and D through this controller which are $A_c = A–B*K$, $B_c = B*\bar{N}$, $C_c = C$ and $D_c = D$.

3.3 Fuzzy Logic Controller Design

Basically, fuzzy logic controller (FLC) comprises of four main components; fuzzification interface, knowledge base, inference engine and defuzzification interface. Figure 6 illustrates the FLC components. The fuzzification interface transforms input crisp values into fuzzy values, whereas the knowledge base contains knowledge of the application domain and the control goals. The inference mechanism consists of decision-making logic that performs inference for fuzzy control actions and the defuzzification interface changes back the fuzzy values into the crisp values.

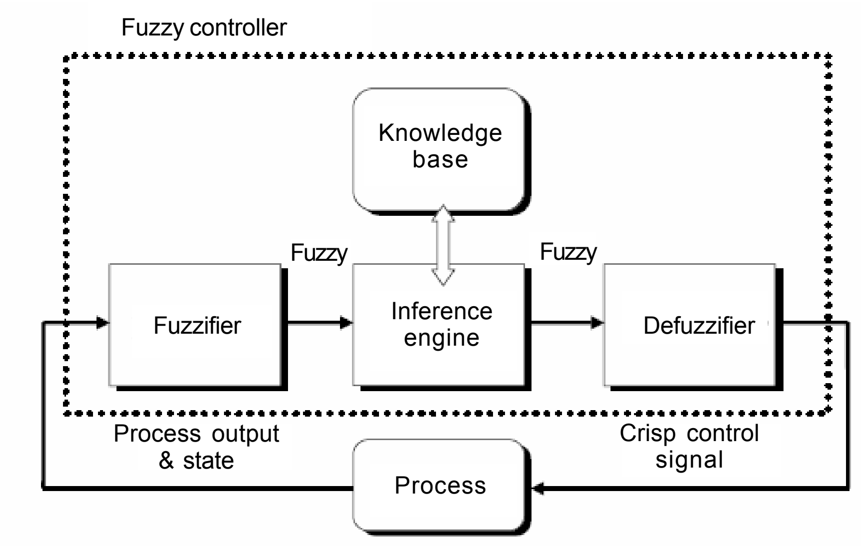


Figure 6 Components in the fuzzy logic controller

The inputs to the fuzzy controller are the error (e) which measures the system performance and the rate at which the error changes (de), whereas the output is the change of the control signal (du). Figure 7 shows the overall closed-loop system for FLC with the ballbeam plant. From the figure, the error (e) is computed by comparing the reference point (desired position) with the plant output. The change of error (de) is generated by the derivation of the error. The error and change of error is fed to the fuzzy controller through a multiplexer.

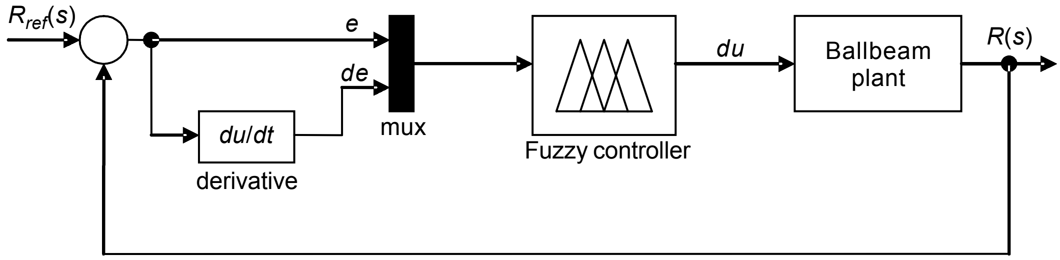
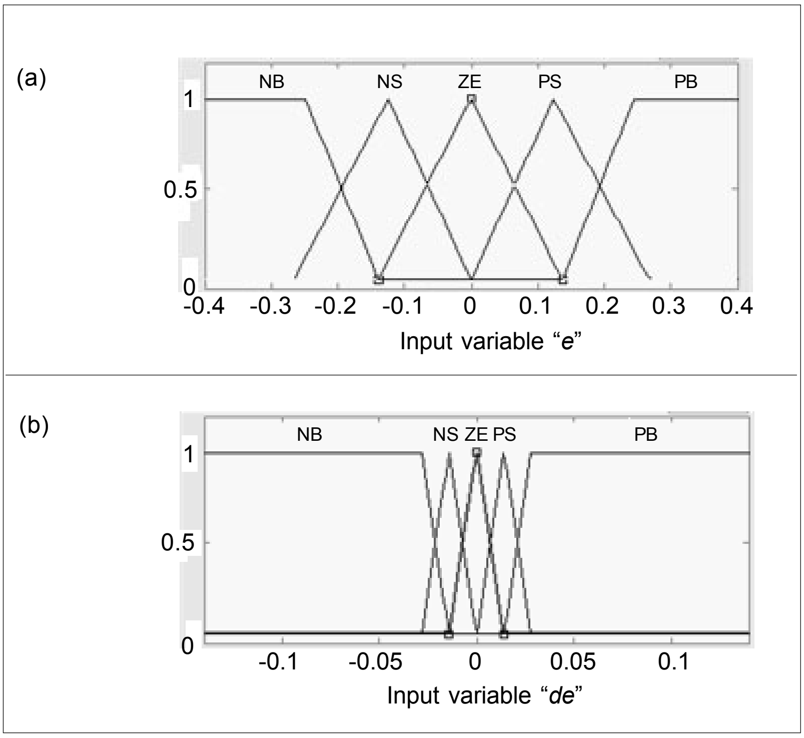


Figure 7 Fuzzy logic controller in feedback loop of ball beam system

The appropriate membership function to represent each fuzzy set need to be defined and each fuzzy set must have the appropriate universes of discourse. For our case study, we choose triangular and trap membership function for each fuzzy set. Figure 8(a), Figure 8(b) and Figure 8(c) shows the fuzzy set of the input (*e*) and (*de*), and the output (*du*). Each fuzzy set consists of 5 types of membership function, which is negative big (NB), negative small (NS), zero (ZE), positive small (PS) and positive big (PB). The universe of discourse is set between –0.4 to 0.4 that implies to the range of beam length (±0.4 meter).

The rule base used to represent the expert knowledge in a form of IF-THEN rule structure. In this design, we use the formulated control rule that was introduced by



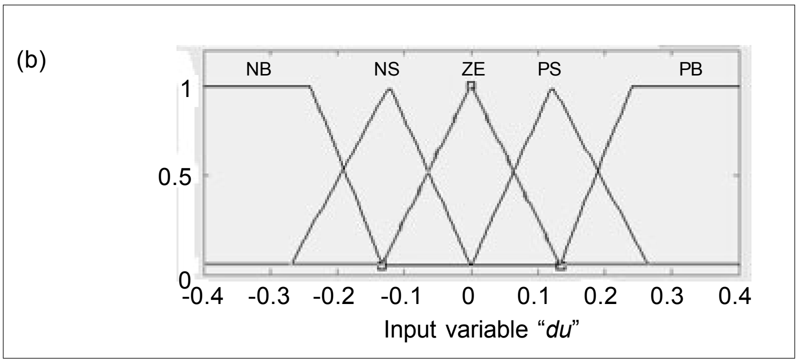


Figure 8 (a) Fuzzy set for the input '*e*', (b) Fuzzy set for the input '*de*', (c) Fuzzy set for the output '*du*'

C.C Lee [6]. These are nine rules that have been utilized in the designed controller, and the rules define in Table 1.

Table 1 Rules for the fuzzy controller

	Error, *e*	**Delta error, *de***	**Delta *u*, *du***
1	PB	ZE	PB
2	PS	ZE	PS
3	ZE	NB	NB
4	ZE	NS	NS
5	NB	ZE	NB
6	NS	ZE	NS
7	ZE	PB	PB
8	ZE	PS	PS
9	ZE	ZE	ZE

4.0 GUI FOR BALL AND BEAM SYSTEM

The purpose of this Graphical User Interface (GUI) is to allow the user to view an animation of the ball and beam system according to the desired setpoint. This allows the user to see the correlation between the plot and the systems physical response. Once the satisfactory compensators were obtained, it will then being interfaced with the graphical user interface (GUI). The GUI has been designed by *m*-files coding in the MATLAB features. The designed GUI then will be integrated with the 3 types of controller which are PID, state space controller, fuzzy logic controller. Figure 9 shows the developed GUI for ball and beam system. Basically, it consists of three main panels which are controller setup, plotting response for ball and beam, and ball and beam animation. The full source code for the GUI is available in MATLAB file, *bnb.m*.

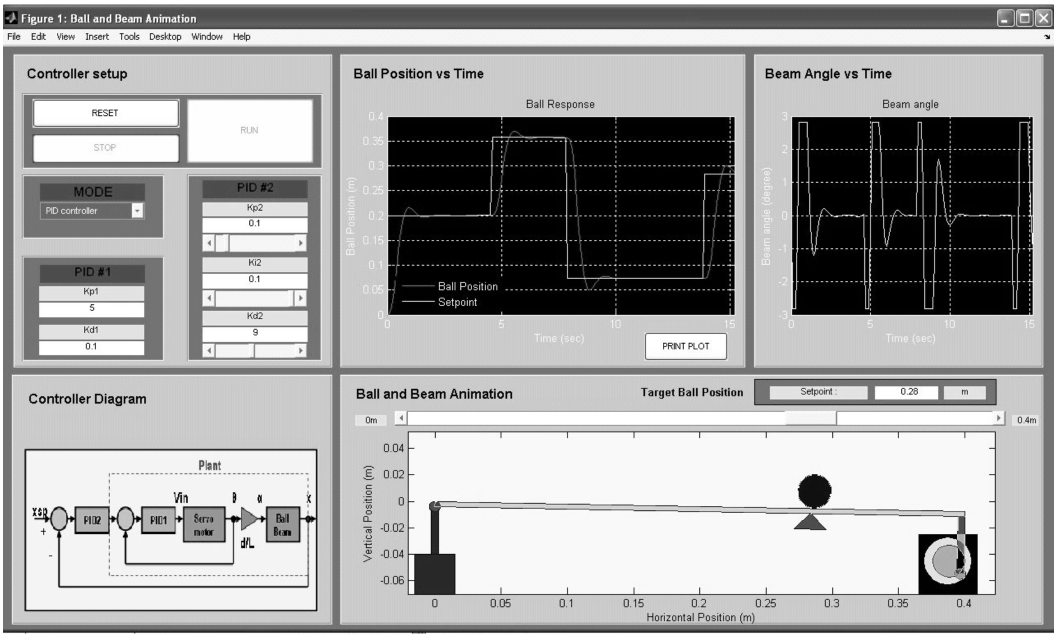


Figure 9 Graphical user interface layout in MATLAB

5.0 RESULT AND ANALYSIS

5.1 Results for Proportional (P) and Proportional-Derivative (PD) Controller

Proportional controller is the most basic strategy for the feedback control law. The controller output is made proportional to the error and the proportionality constant is called the proportional gain (K_p). Unfortunately, this controller is not capable of maintaining the output steady state value at the desired value as shown in Figure 10(a). From this figure, K_p is set to be 0.1 and setpoint at 0.2 meter. While increased K_p gain, the output will response faster because it decreases the rise time (T_r). However, K_p gain is limited by the dynamics of the system, where for ball and beam system the response is limited by the length of the beam (0.4 m). Therefore, when the response oscillates beyond this value, the system will stop as the ball reaches the maximum distance.

The derivative action can be used to create damping in a dynamic system and thus stabilize its behavior. With derivative action, the controller output is proportional to the rate of change of the measurement or error. Derivative action can compensate for a changing measurement and improve the system transient response. Figure 10(b) shows the simulation result of ball and beam system which is controlled by PD controller. The controller gives a very fast response with settling time of 1.2 second and rising time of 0.9 second. The transient response has a very small overshoot of 0.09% that

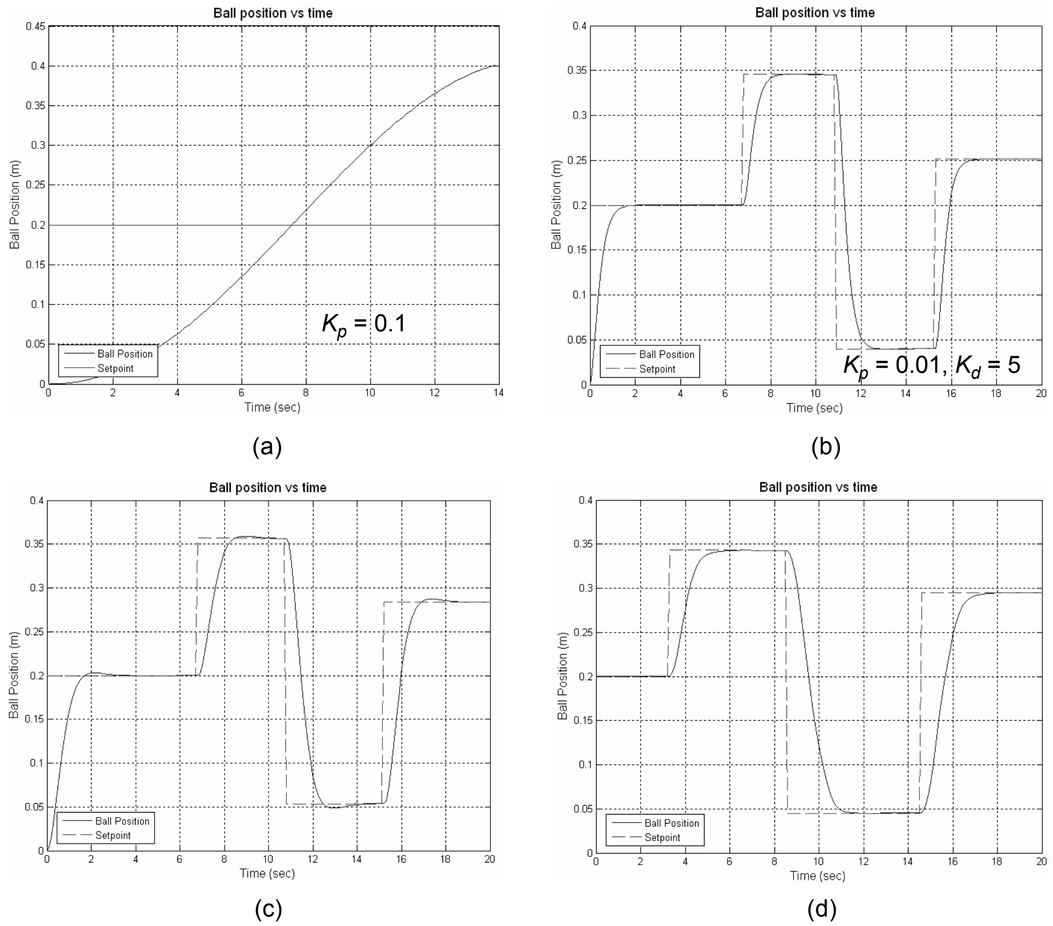


Figure 10 (a) Result for P controller; (b) Result for PD controller; (c) Result for pole placement controller; (d) Result for fuzzy logic controller

near to the critical damp. Besides, it also gives a good steady error at 0.094%. Thus, fuzzy logic controller has an overall good performance that meets all the requirement of the design specifications. While increasing K_d will make the response faster, but at the same time increases the overshoot.

5.2 Results for Pole Placement Controller

Figure 10 (c) shows the simulation results for the pole placement controller. As we can observe, this controller has a good capability to control the ball and beam system. The response is following the changes of the setpoint either in positive or negative direction. The settling time takes only short time to stabilize in its new final value (i.e. T_s = 1.6 second for 0.2 m setpoint). A small overshoot of 1.5% exist during the transient response at peak time, T_p of 2.2 second. It's meant that all the design specification is met.

If we want to reduce the overshoot further than we would make the imaginary part of the pole smaller than the real part. Furthermore, if we want a faster settling time we would move the third and fourth pole further in the left-half plane. It will make the response faster because it decreases the rise time but at the same time will increase the steady state error a little bits. From the plot, we can find that the response has a very small steady state error of 0.0084%.

5.3 Results for Fuzzy Logic Controller

As fuzzy theory is close to human reasoning, these intelligent controllers are derived from some prior information or input-output data of ball and beam system. The FLC provides good performance in terms of rise time and overshoot. The simulation result for fuzzy logic controller is shown in Figure 10(d). As depicted from the figure, we can observe that the ball is tracking follow the changes of the setpoint. This controller is able to give a good response without produce any overshoot. If the input setpoint is set to 0.1 meter, the response time is comparatively fast that give settling time of 1.7 second and rising time of 1.1 second. In further, it also gives a good steady state response of 0.0006%.

Unfortunately, the controller gives slightly different performance if we increase the setpoint value. For example, for the maximum input setpoint of 0.4 meter, it gives the settling time of 2.5 second, rising time of 2.2 second and zero steady state error. Although, the response is varying for a different condition, but it still can meet the design requirement.

5.4 Overall Comparison of the Controllers' Performance

The graph in Figure 11 shows the set-point and the output response for the entire controllers. The best that could be achieved for the entire methods of compensation studied is compared. A glance reveals that the designed PD controller has an overall better performance than P, pole placement and fuzzy logic controller, though it seems that the PD gives fastest response time with the reasonable percentage of overshoot and steady state error. The comparison of the response's characteristics is shown in Table 2.

As we can observe, P is not able to control the system, hence given a remaining unstable system. The other compensation schemes produce very smooth results, but with noticeably different shapes to the response curves. In terms of the settling time, the entire controllers met the design tolerance. The best settling time (T_s) is given by PD controller and the worst by fuzzy logic controller. Pole placement gives 1.5% which is the worst among the controllers; fortunately it gives excellent steady state error of 0%.

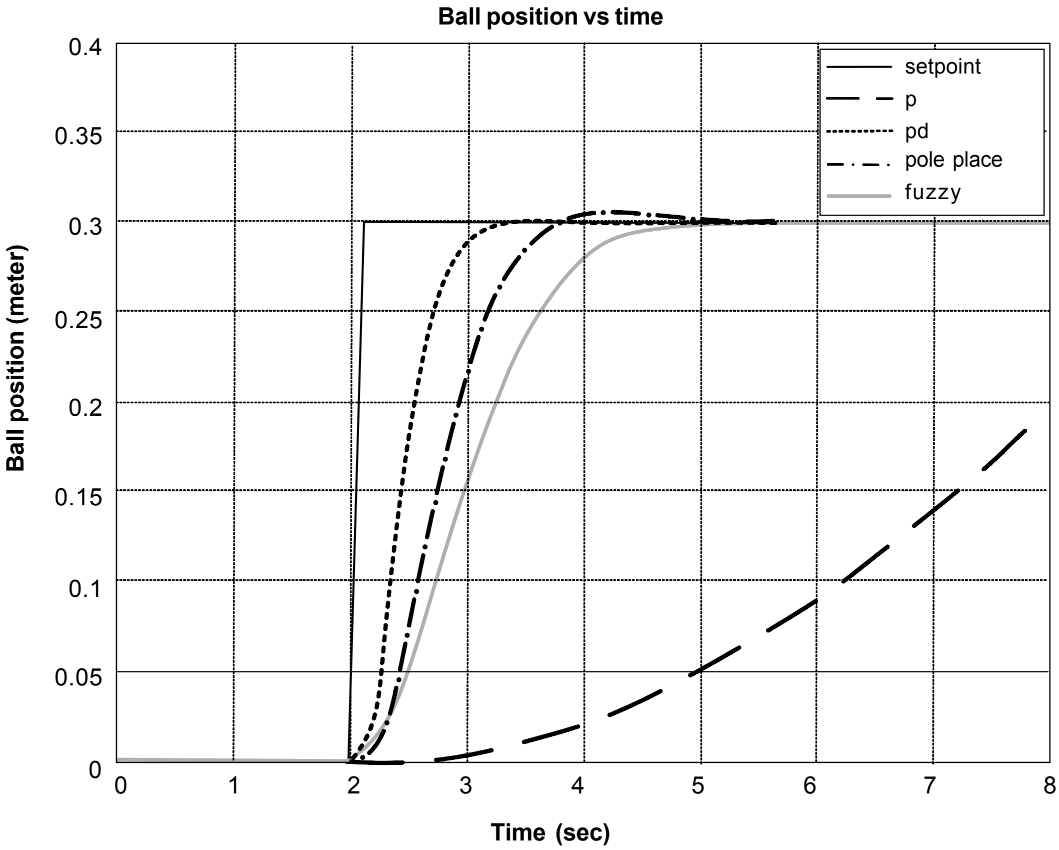


Figure 11 Comparison result for all controllers with 0.3 step input

Table 2 Comparison of the output response's characteristics

Controller	Overshoot (%)	Peak time, T_p (s)	Rise time, T_r (s)	Settling time, T_s (s)	Steady state error, e_{ss} (%)	System status
P	-	-	-	-	-	Unstable
PD	0.2	1.6	0.6	1.1	0.063	Stable
Pole Placement	1.5	2.2	1.0	1.7	0	Stable
Fuzzy Logic	0	-	1.5	2.5	0.030	Stable

Besides, fuzzy is another interesting and feasible method for control system design and can be another alternative for the conventional control and the modern control. In this limited study, it produced surprisingly promising results, which gives almost zero overshoot and very less steady state error. Even the response time is quite late as compared to others, but it is still within the specified tolerance.

6.0 CONCLUSION

The mathematical model for a ball and beam system has been derived successfully. The plant consists of three main components which are servomotor model, angle conversion gain, and ball on the beam dynamic equation. Both of servomotor and ballbeam dynamic has the second order transfer function. Besides, the state space equation was derived in order to design the state space controller. Several controller of conventional, modern and intelligent scheme have been successfully designed to control the ball and beam system. It is quite tedious to design the fourth order system, thus for conventional method, two controllers have been implemented to control those second order components. The modern controller implements the full state feedback control that utilizes of pole placement method, whereas fuzzy logic was utilized for the intelligent control strategy. Furthermore, an interesting ball and beam GUI has been successfully designed by using MATLAB program. The analysis results had shown that PD controller gives better performance than the others. With the basic configuration, it seems that the intelligent controllers are not giving good response time, but still can be an alternative to replace the conventional and modern controller. The results for intelligent controllers are possible to be improved further by using advance configuration and better tuning method.

REFERENCES

[1] Peter, E. Wellstead. 2000. *Introduction to Physical System Modelling*. Academic Press Limited: 221–227.

[2] Hauser, J., S. Sastry and P. Kokotovic. 1992. Nonlinear Control via Approximate Input-output Linearization: Ball and Beam Example, *IEEE Trans. on Automatic Control*. Vol. 37.

[3] Lam, H.K., F.H.F. Leung, P.K.S. Tam. 1999. Design of a Fuzzy Controller for Stabilizing a Ball-and-Beam System. *IEEE Proceeding*.

[4] Hirschorn, R.M. Incremental Sliding Mode. October 2002. Control of the Ball and Beam. *Automatic Control IEEE Transaction.* Volume 47.

[5] Yuhong Jiang C. McCorkell and R. B. Zmood. 1995. Application of Neural Networks for Real Time Control of a Ball-Beam System.

[6] Lee, C. C. 1990. Fuzzy Logic in Control Systems: Fuzzy Logic Controller - Part 1. *IEEE Transactions on Systems, Man & Cybernetics*. 20(2a): 404–419.

[7] Robert Hirsch. 1999. Mechatronic Instructional Systems Ball on Beam System, *Shandor Motion Systems*.

[8] Evencio A. Rosales, Bennett T. Ito, Katie A. Lilienkamp, Kent H. Lundberg, An Open-Ended Ball-Balancing Laboratory Project for Undergraduates. *Proceeding of the 2004 American Control Conference Boston. Massachusetts*. June 30 July 2, 2004.

[9] Jeff Lieberman, A Robotic Ball Balancing Beam. *MIT Media Lab Publications*. February 2004.

[10] Wen Yu, Floriberto Ortiz, Stability Analysis of PD Regulation for Ball and Beam System, *Proceedings of the 2005 IEEE Conference on Control Applications Toronto, Canada*. August 28-31, 2005.