Redesign and control of a Ball and Beam plant

Lucas Dubois Camacho, Carlos Elizondo Rojas, Cristofher Solís Jiménez
lucasduboisc@estudiantec.cr, carlos.ezrj@estudiantec.cr, cristofhersj@estudiantec.cr.
EL5405: Automatic Control Laboratory
Academic Area of Mechatronics Engineering
Costa Rican Institute of Technology

***Abstract*—This paper shows the steps taken to construct, obtain an empirical model and design a control system for the plant known as Ball and Beam. Using primarily PSoC's design tool for the programming of the function of the plant and Matlab's multiple tools for control design in order to obtain the best control possible using an $I + PD$ system with the help of an Integral State Feedback by pole placement. The control method proved to be useful to make the plant worked accordingly to the design parameters, but with undesirable stationary state error, high overshoot and a long settling time.**

***Keywords*—Ball and Beam, Integral state feedback, pole location, PSoC, automatic control.**

I. Introduction

The ball and beam system can usually be found in most university control labs, since it is relatively easy to build, model and control theoretically. The system includes a ball, a beam, a motor, distance sensors and a controller. The basic idea is to use the torque generated by the motor to control the position of the ball on the beam. The ball rolls on the beam freely. By employing linear sensing techniques, the data from the sensor can be taken and compared with desired positions values. The difference is fed back to the Integral-Proportional-Derivative ($I + PD$) controller to obtain the desired position. The mathematical model for this system is inherently nonlinear but may be linearized around the horizontal region. This simplified linearized model, however, still represents many typical real systems, such as horizontally stabilizing an airplane during landing and in turbulent airflow. By considering real plant problems such as the sensor noise and actuator saturation, the controllers of the system become more efficient and robust [1].

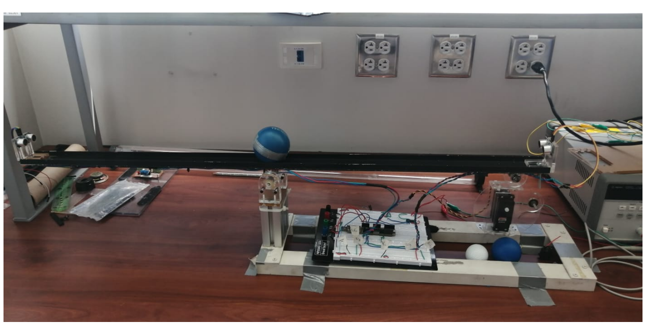

Fig. 1: Ball and Beam plant.

Regarding the control used, it should be noted that an Integral State Feedback was used. Originally, integral action was employed in controller design to overcome the problem of steady state errors. In many cases, it is difficult to obtain an accurate value for the plant gain, in part, because plants are typically nonlinear and the plant model is linearized at a particular point. Therefore, the steady state errors will result even though the model is sufficiently accurate for good feedback controller design. The solution is to include an integral term in the controller design, then, a state space integral controller can be designed with the purpose of achieving satisfactory dynamic response in terms of rise-time, overshoot, settling time or other measures of transient response [2].

Proportional-Integral-Derivative (PID) controllers are the most common type of controllers for controlling of many systems. The main reason of this situation is that PID controllers have simple structure and can give satisfactory results in many industrial control applications. There are lots of PID controller design methods, however, in PID controller application, proportional, integral and derivative components are located on the forward path and affect the error between the reference input and output of the closed loop control system and implementation of the PID controller in this way causes the derivative kick which is an undesirable situation in the system output for a step reference input, to overcome these difficulties, it is generally suggested the use of PI-PD controller to control unstable, integrating and resonant processes more effectively. Since PI-PD controllers have four parameters to be tuned, it is not easy to design these controllers. A similar controller structure to PI-PD controller is I-PD controller which has one less tuning parameter than PI-PD controller. Although I-PD controller has one less parameter to be adjusted, it gives comparable responses to PI-PD controller and allows the control of systems with time delay [3].

II. Procedure and results

A. Main components

In this project, the initial mechanical system of the plant has been already built. However, it had problems that needed to be fixed in order to provide a proper functioning. The first step to improve the functioning was to add a second proximity sensor. This way, the control system can be more robust since if one of the sensor has a bad reading of some sort, the second one can have a correct reading, making the control more solid overall.

The chosen sensor was the HC-SR04 which is an ultrasonic sensor.

The ultrasonic sensor has four pins, two for the electrical power supply, one for the trigger and one for the echo (output). Once the trigger input receives a positive value, the sensor send an ultrasonic wave, which travels trough the air and once it bounces with an object and the sensor receives it, the echo outputs a PWM signal, which pulse width has a direct relationship with the distance between the sensor and the object. Another important component for the plant is the servomotor, this part of the plant is needed to provide torque needed to move the beam and allow the control system to move the ball to the center of the beam. This position would be the point of reference of the control system. The chosen servomotor was a HS-805.

One of the requirements for the design of the plan is to use the micro-controller known as PSoC 5LP to control the motor, get the readings of the sensors and design the overall control system. This micro-controller is capable to manage the PWM signals that control both the sensors and the servomotor, as well as saving the sensors readings. The Infineon company (developer of the PSoC 5LP) provides the software known as PSoC Creator which allows the user to program the functioning of the micro-controller. In this case, functions like analogical to digital converters, PWM generators and interrupts were needed for the design of the plant.

1) Programming of the PSoC 5LP: To program the PSoC 5LP, the code used as well as the flow charts that describe its behavior are shown in the appendix section.

B. Reconstruction of the plant

As mentioned, it was necessary to add an extra sensor to the plant and the previous one was replaced by a new one, to do this it was required to build a small support structure for the new sensor, it was created as similar as possible to the one that was already in the plant, as shown in appendix.

In addition, a breadboard was used to place the PSoC and it's respective connections, an RC filter and a follower amplifier to isolate impedances were also placed on the breadboard. The circuit can be seen in the appendix section.

C. Obtaining the model of the plant

Once the main construction was rebuilt, it is necessary to obtain a model that is capable to approximate the physical behavior of the plant. The first approximation used was to obtain the measurements of the sensors when the plant receives an input of a positive step, then a negative step (as shown in the fig. 2). Those data were used in the MatLab's identification tool, obtaining the model of the plant described by the transfer function shown in equation 1.

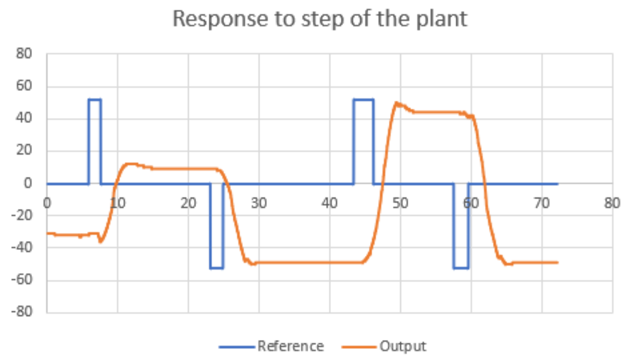

Fig. 2: Response of the plant to step signals vs time (s).

$$G(s)_1 = e^{-2.54s}\frac{2.1684*10^5 s - 716.3}{5.118*10^5 s^2 + 1459s + 1} \tag{1}$$

As shown, the first model, while having a 94 percent of accuracy with the data. Is more complex than needed to obtain a proper control. With the recommendation of the supervising teacher. The model of the plant should be as similar as possible to a gain with a double integrator, as shown in the equation 2.

$$G(s)_{desired} = \frac{K}{s^2} \tag{2}$$

In order to obtain a model similar to the desired one. A new experiment was needed, now the dataset used to generate the model was using only one step of input, also the ball started in the center of the beam and the sensors read the position of the ball moving through the beam as the step is applied. However, MatLab's identification tool does not have an option to generate a model in the desired form. The new strategy to obtain it then was to cut the dataset obtained to only have the data of the exact time when the ball moves, then using Microsoft's Excel graphics functions, it was obtained a polynomial equation (Eq. 3) that has a similar form to the graphed data as shown in fig. 3.

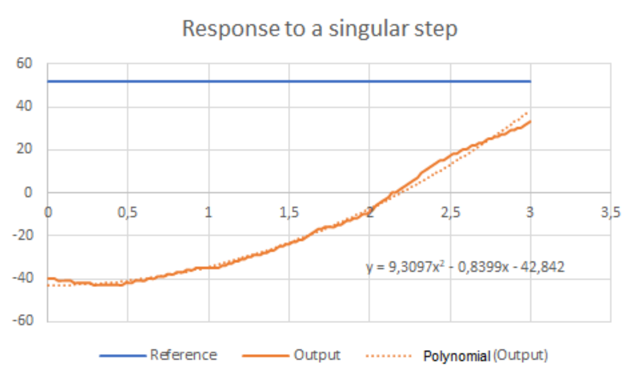

Fig. 3: Polynomial equation obtained through graphing the measures of the plant responding to a single step vs time (s).

$$Y = 9.3097t^2 - 0.8399t - 42.842 \tag{3}$$

Once the equation was gathered, it is still needed to get the transfer function equivalent to it. It was obtained with the equations 4 and 5. Where C is the last coefficient of the polynomial equation (9.3097), K is the gain of the model and A is the value of the input. In this case the input was the reference signal that in this experiment has the value of 52. (52 means a step to move the beam upwards, -52 means a step in the opposite direction and 0 means to keep the beam in the original position). That way it was possible to get the final model used to describe the plant, shown in the equation 6

$$Ct^2 = (A * K * t^2)/2 = L^{-1}(K/s^2 * A/s) \tag{4}$$

$$K = \frac{2C}{A} \tag{5}$$

$$G(s) = \frac{0.19395}{s^2} \tag{6}$$

D. *Designing the structure of the control system*

As for the design of the control system structure, a first iteration of a typical PID compensator was created, for which, the integral, derivative and proportional constants were obtained using the SISO tool provided by Matlab. Choosing the desired characteristics (settling time of 30 seconds and less than 3%) and then placing poles and zeros in the locations that lead to simulations with the expected result. Nonetheless, when implementing the PID control to the plant, the results where not even close to the expected from the simulation. Considering the designed control was not able to successfully control the plant, a more robust method was required.
For this, an integral state feedback control was thought of and then implemented through an I+PD topology. This method showed better results and these are discussed in the next section.

E. *Results*

Once the final control system got online. Three different tests were design to verify the effectiveness of the control system, the one shown in the Fig. 4 was taken by just putting the ball on one of the ends of the beam without adding any force or disturbance in any way. While the Fig. 5 presents what happens if a force is applied to the ball, therefore generating an agressive disturbance to the control system.

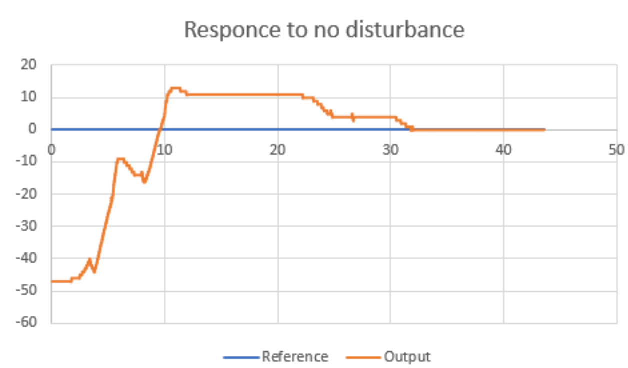

Fig. 4: Results obtained without applying any disturbance to the plant vs time (s).

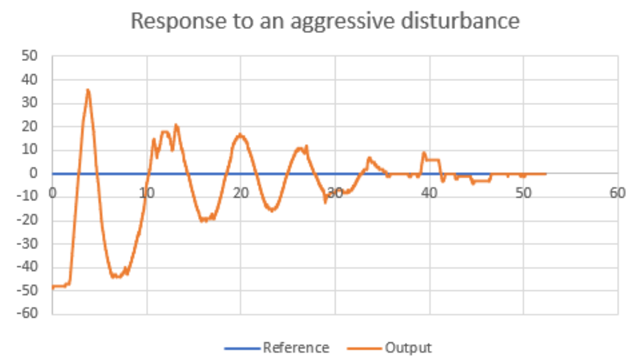

Fig. 5: Results obtained applying an aggressive disturbance to the plant vs time (s).

Since the control system got results expected for the design. Ten different test were run in order to get an average stabilization time of the control system, the results are shown in the table I.

TABLE I: Results of stabilization time to multiple instances of the control system working

Number of test	Stabilization time(s)
1	41.34
2	58.62
3	24.51
4	18.34
5	22.07
6	35.46
7	40.45
8	50.23
9	29.35
10	53.02
Average	37.34

III. Results Analysis

As mentioned, three experiments were performed to analyze the implemented control, the first one was performed by placing the ball at one of the ends of the beam and tilting the beam downward until reaching the limit allowed by the stop, Fig. 4 allows to observe how the output is initially at a maximum (it should be noted that this value has a direct relationship with the subtraction between the sensor measurements), ideally, once the control code is executed the ball should pass through the center in less than 5 seconds but in the graph it is observed that this happens until 10 seconds (the center corresponds to the reference value 0), the reason for this event is related to a false measurement of any of the sensors, because if the ball has not reached the reference, the beam should remain with the same angle of inclination to make it reach the center, but the figure shows that shortly after 5 seconds there is a small unwanted irregularity that falsely indicate that the ball had already passed through the center and the beam should change its angular position in the opposite direction, such errors delay the control process, although the code has a method to avoid incorrect measurements sometimes this method is not 100% effective and that is reflected in this type of anomalies in the control.

Once the reference value is reached, by inertia the ball will continue in the opposite direction but this time due to friction and the control that varies the angle of the beam, the ball does not reach the other end but moves a few centimeters allowing to repeat this process until the desired reference is reached. Note that after 12 seconds (approximately) the output signal remains constant for at least 10 seconds, this phenomenon is a product of the static friction between the ball and the beam that, although the beam tilt angle is different from zero, the ball remains in the same position, clearly this is a great limitation for the control of the system and some methods were tried to reduce this friction, however there were no noticeable improvements and it was defined as a possible future change to improve the plant. Returning to the graph, after 30 seconds it can be seen how the output value reaches the desired reference and remains constant in that position.

For the next experiment an aggressive perturbation was applied to the plant, that is, from one of the ends of the beam the ball was manually propelled until it reached the other end, this impulse can be observed in the results of Fig. 5 where it crosses the center in less than 5 seconds and a quite oscillatory behavior can be appreciated in comparison with the previous experiment, for example, in the first experiment the ball was only in the center in two occasions while in this case this happened in more than 10 occasions. This can be understood due to the difference in the speed of the ball in both cases, since in the case where the ball does not move very fast the measurements will be more similar to each other, which allows the response time of the implemented control to be reflected in a better way, while for the case where the ball moves very fast, there will be measurements with quite noticeable differences and the control will take longer to be effective, this is seen in the stabilization time, where it can be noted that for this second experiment is close to 46 seconds. Here, again, an improvement could be implemented in the plant, in this case the response time so that it responds more adequately to this type of sudden disturbances.

The last experiment allowed obtaining an average stabilization time of the plant, it should be noted that the measurements presented a quite notorious deviation as shown in table 1, this indicates that the results do not have the expected consistency, it is clear that one of the main reasons why this problem occurs is the friction already mentioned, which randomly affects each repetition, since in each case the ball will reach a different position with different angles and different measurements. Despite this, in all the repetitions a stabilization time of less than one minute was obtained, this fulfills the objectives initially proposed and the average time was 37.34 seconds. However, there were some cases (outside the experiment) where some stabilization times reached even two minutes, this is due to erroneous measurements of the sensors, the system self-induced disturbances that made practically the control process to start again, although they are atypical cases, it is important to consider them in future improvements to the system.

IV. Conclusions

The Ball and Beam plant got successfully redesigned, having an average stabilization time of 37.34s, but a high steady state error which is reflected in the vibration of the motor when the plant stabilizes. The overshoot can be interpreted as the difficulty that the system has in placing the ball in the center of the beam. This has a direct influence with the stabilization time of the plant. The use of a simple model to approximate the operation of the plant proved to facilitate the development of a control system that works correctly.

Out of the multiple control systems that were used, the I+PD with integral state feedback constants turned out to have the most stable results.

The time constant used in the derivative filter of the control system has a high importance to the steady state error of the plant, reducing drastically the frequency of the vibrations of the motor.

The mechanical construction of the plant introduces some limits to the control system in the form of the degrees that the beam can be moved.

V. Recommendations

The connections to the servomotor should be as isolated as possible in order to avoid any kind of noise in the PWM signal that controls it. The physical position of the servomotor should be closer to the center of the beam, by doing this, the control system can have a more precise control of the degrees than the beam can move.

If possible, the sensors used should be either a better quality or a different type of proximity sensor, since using two ultrasonic sensors on both sides of the beam can generate problems of a sensor getting the read from the ultrasonic signal of the other sensor if both of them are working at the same time, also the used sensor has been shown to provide erroneous measurements with a greater frequency than desired.

The longitude of the beam should be shorter than the one used, being one meter long made more problems to the control system than needed since that longitude restricts the degrees of movement of the beam. For future iterations of this plant, there should be a better treatment of the system delay reacting to an input, as this can drastically improve control performance.

VI. References

[1] A. Mohammed, A. Abdallah, O. Ahmed, and A. Abass, *Design and implementation of ball and beam control system using pid controller*, Sudan University of Science and Technology, Oct. 2015.

[2] R. Gao, A. O'Dwyer, S. McLoone, and E. Coyle, *State feedback integral control by velocity-based multiple model networks*, IEEE Xplore, 2004.

[3] F. Peker and I. Kaya, *Integral-proportional derivative (i-pd) controller tuning for pure integrating processes with time delay*, IEEE Xplore, 2019.