Contents list available at IJMEC

International Journal of Mechatronics, Electrical and Computer Technology (IJMEC)

Journal Homepage: www.aeuso.org

PISSN: 2411-6173 - EISSN: 2305-0543

Design and Implementation a Ball Balancing System for Control Theory Course

Zhen Gao*, Sahan Wijesinghe, Thushan Pathinathanpillai, Eric Dyer, Ishwar Singh

School of Engineering Technology, McMaster University, Hamilton, Ontario, Canada

***Corresponding Author's E-mail:** gaozhen@mcmaster.ca*

Abstract

The main purpose of the project is to develop one ball balancing system to keep a ball balanced on a beam using a microcontroller and related control algorithm to adjust the angle of the beam with real-time sensory feedback. A constant angle of a beam causes the ball to glissade in the axial direction due to the gravity. Based on the closed loop real-time control system and well-tuned parameters, it will necessarily adjust the angle of the beam to minimize the error, namely the distance between the actual position and anticipated/setpoint position of the ball. During this process, three prototypes are made and compared with each other especially in aspects of reasonableness and cost. The entire system could be easily designed with the integration of one servo motor, one distance sensor, one microcontroller board, and one mechanical platform. This system can be used for the undergraduate students to learn and understand the fundamental of proportional-integral-derivative (PID) control. Besides, they will comprehend and implement the control algorithm through microcontroller programming. In the situation that if the students have no much knowledge about MATLAB. The interface program is developed to plot the real-time position curve in MATLAB. This simplified system will serve as a bridge to connect the theoretical knowledge and hands-on skill. The proposed method is also applicable to more complicated system.

***Keywords:** Ball Balancing System, PID Control, Pedagogical Research, Real-time Sensory Feedback, Microcontroller*

1. Introduction

The initial reason to design a ball balancing system is to create new lab for the control theory course. Control theory is a traditional and fundamental course for automation education. The control theory course is offered in the background that the students know Laplace transform, programmable logic controller, instrument and measurement, and a little bit information of PID control [18]. Basically, they have no knowledge about MATLAB and microcontroller. Besides, the students learnt programming languages such as Visual Basic and C++ from the general technology courses. However, they got limited chance to enhance their programming skills in Year-2 and Year-3 technical courses. Thus, in this course, microcontroller will be applied as a tool to improve the students' programming ability and help them understand the principle of closed-loop control, feedback system, and PID algorithm in a vivid and visual way. Another purpose of this lab is that the students just start to learn about MATLAB. What they can do is to use it to conduct some mathematic calculation such as matrix manipulation and apply the self-carried control tool in MATLAB to build the block diagram of control model with continuous signal. They get no idea about how to obtain and plot the real-time data from external control system. Therefore, this work will get them familiar with MATLAB programming and application of MATLAB in a mechatronic system.

The sensitive ball and beam system naturally is unstable even with a restricted beam angle which is nearly horizontal [5]. Without a real time feedback based closed loop system, the ball will

Article History:

Received Date: Jul. 02, 2015
Accepted Date: Sep. 07, 2015
Available Online: Sep. 10, 2015

eventually roll off to either end of the beam due to the force of gravity and external disturbance such as mechanical vibration. Many control problems in the world are aimed to conquer the issue of instability by designing a closed loop feedback control system. Although the proposed prototype is simple, the ball balancing system is relevant to many applications in today's industry [6] [7] [8] [11] [14] [15]. For example, when a chemical reaction generates heat and the reaction speeds up as the temperature increases, it may create a continuous unstable loop of the reaction. The temperature of the reaction should be controlled and stabilized to avoid equipment damage and accident. The system balancing issues were also commonly seen for robot control [12] [13].

2. Prototypes and final model of a ball balancing platform

2.1 First generation model

The first step for obtaining a primary model for the project was to determine the structure for the ball beam balancer. This involved determining:

(a) The location to mount the servo motor (center or corners)

(b) The location and orientation of the ultrasonic sensor

After careful planning, the first prototype was designed in an A-Frame structure as shown in the attached figure). This prototype was crafted using basic materials which involved wood and bamboo sticks.

The advantage of this model is that the A-frame structure requires minimal rotational gears because the servo motor is mounted directly to the beam.

Its disadvantages are concluded as follows:

(a) Due to the poor choice of materials, the first prototype would consistently vibrate causing the ball to fall off the beam. This occurred due to the fact that the servo motor could not be securely mounted to the bamboo beam.

(b) This prototype also did not include a proper mounting platform for the ultrasonic sensor. Thereby, the only option was to mount the ultrasonic senor to the beam using a zip tie, which in-turn caused the ultrasonic sensor to also vibrate constantly.

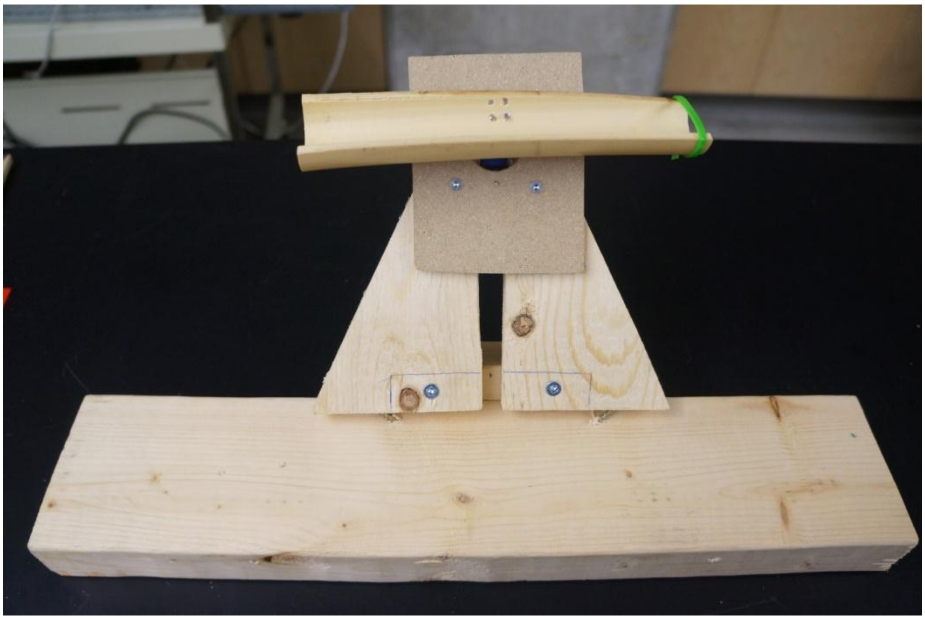


Figure 1: The first prototype of the ball beam balancer

2.2 Second generation model

The next step was to take the previous model and improve upon the structure, to basically minimize the vibration and design a more solid and highly secure model. For the second prototype (as shown in the following figure), the same 'servo-motor in the center' structure was used for a minimalistic design. But the beam was designed and printed in PVC plastic using a 3D printer for a much stable structure. The 3D printed model also includes a slot to mount the servo motor securely, and a platform to hold the ultrasonic sensor.

The authors observed the following advantage of this design:

(a) This second model is even more minimalistic than the first prototype because the A-frame structure was dropped and only a flat 2x2 wood piece is used for the base structure.

(b) Much more secure with absolutely no vibration because the servo-motor can be securely mounted in the slot.

(c) Secure platform to mount the ultrasonic sensor and hold it in place when the beam is moving.

The disadvantages cannot to be ignored either:

(a) Although this model was much more secure over the first, the main problem with this model is the fact that the beam is not perfectly straight. This issue occurred due to the limitations of not being able to print the entire beam in one piece; but rather having to print two separate pieces of the beam and joining them together. This problem made it hard for the ball to balance directly in the center.

(b) Furthermore, this second prototype is rather bulky and much heavier than the previous model, making it harder to handle and reduces mobility.

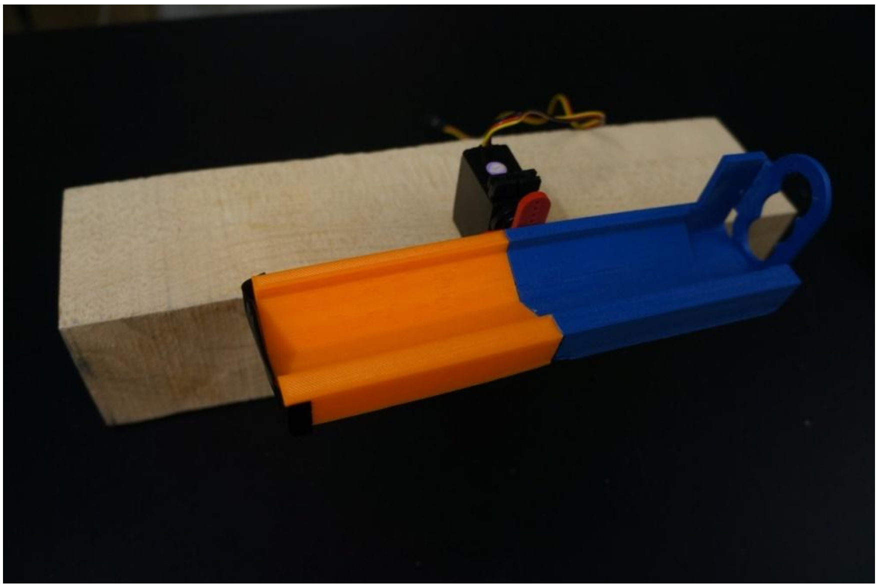


Figure 2: The second prototype of the ball beam balancer

2.3 Final model

The main design objective of the final model was to create a structure that is not only sturdy and well balanced, but is also light-weight and portable to carry. Thus, it was determined that the best option was to go back to using wood because of low cost and time factor. Although, the 3D printed prototype was ideal and stable, the biggest issue was that it took time to manufacture several models and also used a lot of material (plastic).

As seen in the following figure, the final model cuts down on the amount of materials used for production. And thus a much more portable and efficient product was designed.

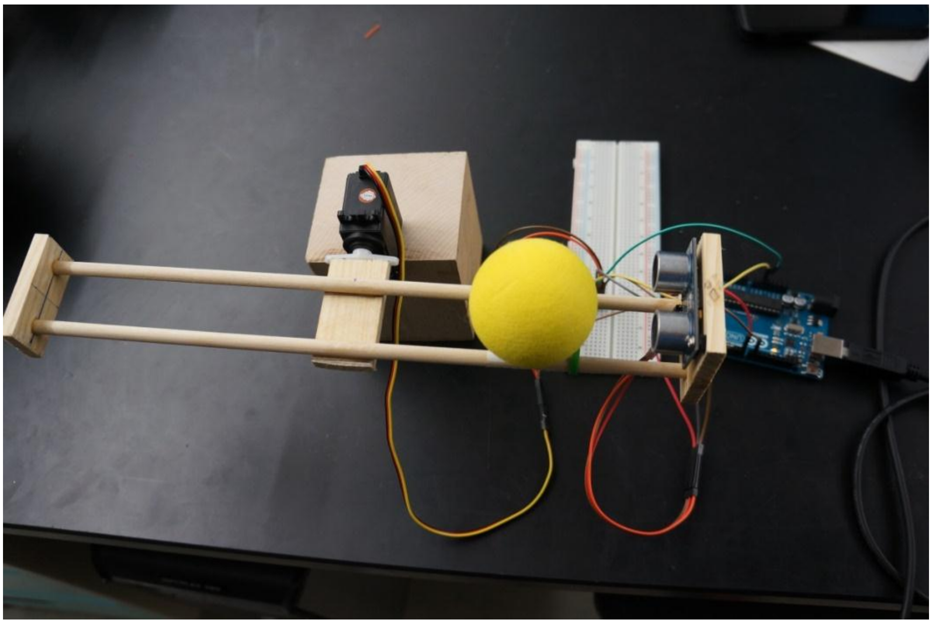


Figure 3: The final model of the ball beam balancer

Beam:

The previous beam idea was dropped, and a much more efficient design was created, in which the beam consisted of two rods carefully placed apart for the ball to fit nicely. This helped to reduce the overall weight of the model and also design a well-balanced product for the ball to travel smoothly.

Servo motor:

Like the previous two prototypes, the best way to mount the motor is directly to the beam, positioned exactly in the center. As seen on the picture below, the motor is attached to a platform that acts as a base for the two rods.

Base:

Once again, the base was made to look minimalistic. Therefore a simple block was used as a base, but this time the block size was reduced to make it more portable.

Ultrasonic Sensor:

A platform was designed to mount the ultrasonic sensor to the two rods. The advantage of this platform is that, it allows us to mount the sensor horizontally, unlike the previous two models where the sensor was mounted vertically.

Protoboard and microcontroller:

The microcontroller used for this model is an Arduino Uno board [1] [3], with connections to Digital Pins 7 and 9 for the ultrasonic sensor and servo motor. The final design idea is to create a platform to mount the ball beam balancer, the protoboard, and the Arduino microcontroller all together.

Therefore, after careful planning and an extensive design process, the final model of the ball beam balancer was created as shown in the following figure. Thus the goal for making a well-balanced, sturdy, and an easily portable model was achieved.

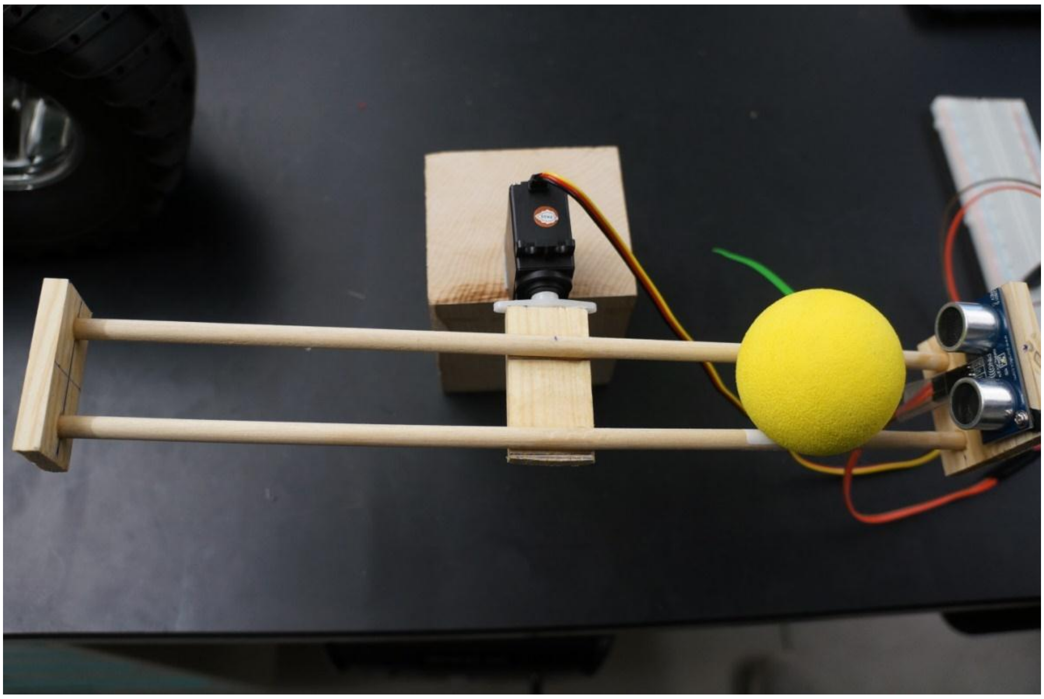


Figure 4: The final model of the Ball Beam Balancer

3. Control algorithm

Because of its popularity and affordability, Arduino microcontroller is chosen to execute the control algorithm. Based on the integration of Arduino (UNO Rev 3.0), ultrasonic sensor (Parallax Ping) and servo motor (HS-645MG), the ball on the beam could be maintained close to the desired setpoint by creating a control algorithm to control the angle of the beam. Having the sensor to monitor and keep track of the ball will feed the data to the microcontroller in real time to adjust the beam angle based on the deviation between the setpoint and the actual position of the rolling ball.

PID was widely used as a liner controller [10]. According to the deviation of the given value and the anticipated value, the theoretical PID algorithm is given as follow [2] [4]:

$$u(t) = K_p \left[e(t) + \frac{1}{T_I} \int_0^t e(t) d(t) + \frac{T_D de(t)}{d(t)} \right] \tag{1}$$

where K_P is the tunable proportional gain, T_I is the tunable integral gain, and the T_D is the tunable derivative gain.

The following diagram shows the closed loop control diagram of the proposed single axis motion balancing system. A desired position is used as the reference of the system. The feedback signal is taken from the distance sensor and fed to the microcontroller to compare with the setpoint. The error signal is calculated and used as the input of PID controller. Then the controller adjusts the level of the beam by changing the servo angle.

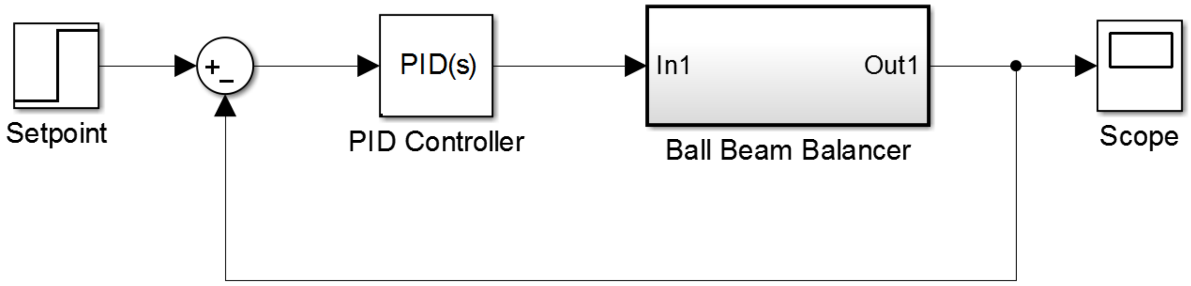


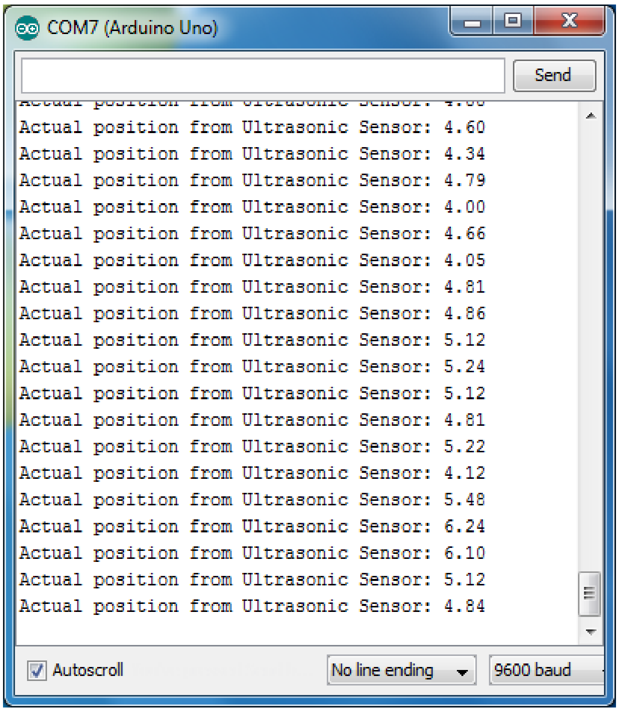
Figure 5: The block diagram of the control system

The platform itself should be lightweight to support instantaneous motion of the servo motor. Use millisecond value readings from the ultrasonic distance sensor as a basis of measurement to achieve real time control with minimal time delay.

The pseudo code of the proposed algorithm is given as follows:

```
Define the servo objective
Define the previous position
Define the current position
Define difference in time between two adjacent measurements
Define the velocity
Define the home position for the servo motor
Serial communication setup/initializing
Control loop starts:
    previous position = current position
    current position = current sensor reading
    velocity = (current position - previous position) / time difference between two adjacent measurements
    calculate the coefficient of proportion
    calculate the coefficient of differential
    adjust the feed of the motor based on the different positon of the ball and the direction of its movement
    check the current position of the ball to judge if the while loop should terminate or continue
Control loop ends
```

The actual position of the distance sensor can be monitored in real-time from the serial communication port as shown in the following figure.



```
Actual position from Ultrasonic Sensor: 4.60
Actual position from Ultrasonic Sensor: 4.34
Actual position from Ultrasonic Sensor: 4.79
Actual position from Ultrasonic Sensor: 4.00
Actual position from Ultrasonic Sensor: 4.66
Actual position from Ultrasonic Sensor: 4.05
Actual position from Ultrasonic Sensor: 4.81
Actual position from Ultrasonic Sensor: 4.86
Actual position from Ultrasonic Sensor: 5.12
Actual position from Ultrasonic Sensor: 5.24
Actual position from Ultrasonic Sensor: 5.12
Actual position from Ultrasonic Sensor: 4.81
Actual position from Ultrasonic Sensor: 5.22
Actual position from Ultrasonic Sensor: 4.12
Actual position from Ultrasonic Sensor: 5.48
Actual position from Ultrasonic Sensor: 6.24
Actual position from Ultrasonic Sensor: 6.10
Actual position from Ultrasonic Sensor: 5.12
Actual position from Ultrasonic Sensor: 4.84
```

Figure 6: The actual position of the distance sensor monitored from the serial port

For the classic PID algorithm, the coefficients are constant. When the error is significant, the fixed coefficients will affect the control efficiency. Similarly, when the error is tiny, the less dramatic control gain can make the convergence more smoothly. This can be achieved by adaptive PID control scheme whose control gains of proportion, differential and integral are changeable according to the real-time control situation especially the features of error signal [9] [17].

For the traditional control theory course, adaptive control usually was not covered. The project allows the students to apply the idea of adaptive control when programming in microcontroller. For example, if the position of the ball is far away from the setpoint, the rotation angle of the beam can be accordingly enlarged more than the classic PID algorithm. If the position of the ball is close to the setpoint, the rotation angle of the beam can be accordingly reduced more than the classic PID algorithm.

4. Real-time data acquisition in MATLAB

MATLAB is a very powerful tool for scientific computation including control simulation, signal processing, and function optimization, to name a few of them. MATLAB is like engineers' and scientists' calculator. In addition to academia, MATLAB nowadays is also useful in industry. Although as a programming language, it is much less popular and commercial than Java and C++. In some special fields, e.g. control, the importance of MATLAB cannot be replaced by others.

MATLAB carries a graphical programming environment called as Simulink, which can be regarded as a visual library of MATLAB. In Simulink, there are various build-in toolboxes including control system toolbox. Like Python, MATLAB is also a scripting language. It is suitable for self-learning by googling and by typing "help + build-in function" in the "command window". For example, MATLAB has a self-carried function called pid. If one wants to understand how to use it, she/he can type "help pid" in command window and then the related information will pop up.

Basically, MATLAB is a programming and computation tool. The students should know some fundamentals such as transfer function then they can build the control model based on block diagram in Simulink. Since MATLAB is not open source software, the students are expected to fully take advantage of the lab hours to get familiar with it based on sufficient practice.

MATLAB quickly visualizes a function and draws it with 2D and 3D based on the number of variables. For example, to generate the following 3D heart shape function, the students can use the attached sample code.

```
steplength = 0.04;
[X,Y,Z] = meshgrid(-1:steplength:1, -1:steplength:1, -1:steplength:1);
Coutour = (-(X.^2).*(Z.^3)-(9/20).*(Y.^2).*(Z.^3))+((X.^2)+(12).*(Y.^2)+(Z.^2)-.6).^3;
patch(isosurface(X,Y,Z,Coutour,0), 'FaceColor','r', 'EdgeColor','b')
daspect([.5 .5 .5])
view(3)
```

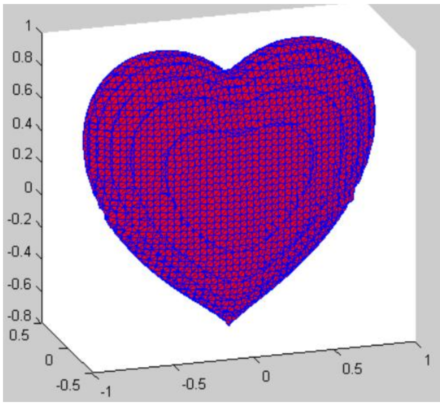


Figure 7: 3D heart shape function

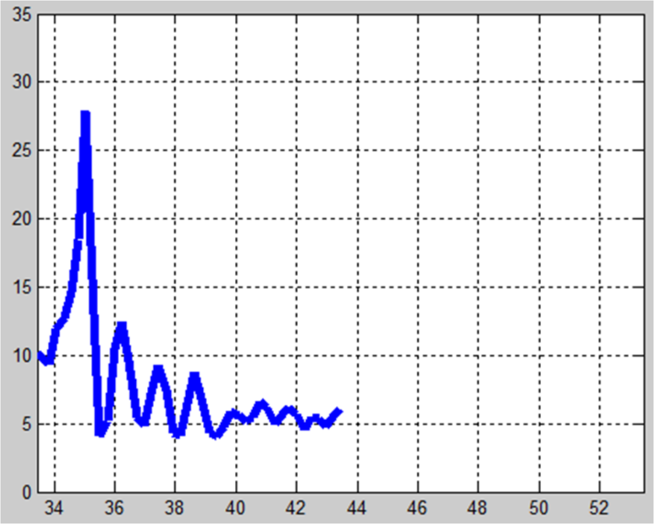
The interface between Arduino and MALAB can be achieved by the following code:

```
Arduinodataforposition = serial ('COM7', 'BaudRate', 9600);
fopen (Arduinodataforposition);
fclose (Arduinodataforposition); % end communication with arduino
```

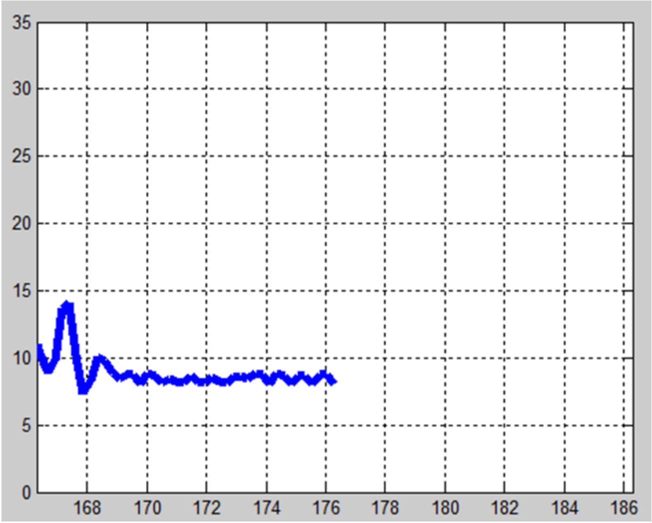
The first line implements the serial communication assuming the serial port on the host computer is 7. The second line is to initiate the Arduino communication. The last line will terminate the serial communication between Arduino and MATLAB. A 'while loop' can be created between line 2 and line 3 to display/record the real-time date which is named as 'Arduinodataforposition' which represents the actual position in real time of the ball on the beam.

5. Experiments

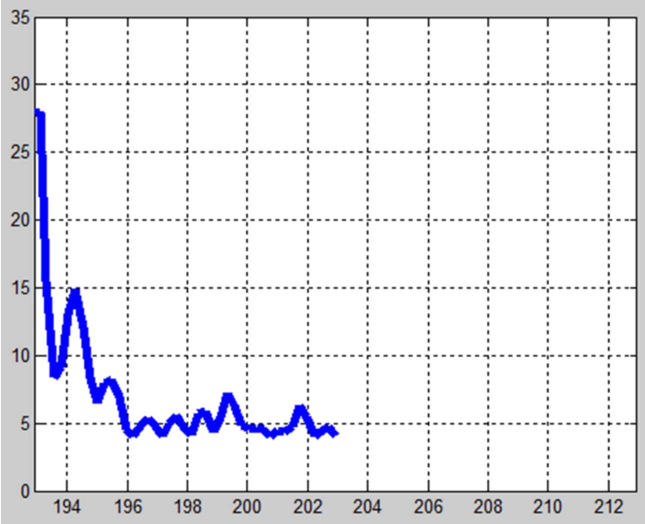
The following figure shows the sample results of the experiment. The actual position is measured by the sensor and is transferred to the serial port, namely COM7 in this case. MATALB acquires the data from this serial port. From the subfigure (a)-(c), it can be observed that the error is largely reduced when time elapses. However, the shaking still exists. The problem probably is caused by multiple reasons. One reason is that the plastic table tennis balls which used in the experiment are very light. It is difficult to completely stop it because of a lack of sufficient friction. Another reason is that the parameters may be not properly tuned. The last subfigure shows a decent result with an expected quality. However, the repeatability of the proposed system still needs to be improved.


35
30
25
20
15
10
5
0
34
36
38
40
42
44
46
48
50
52

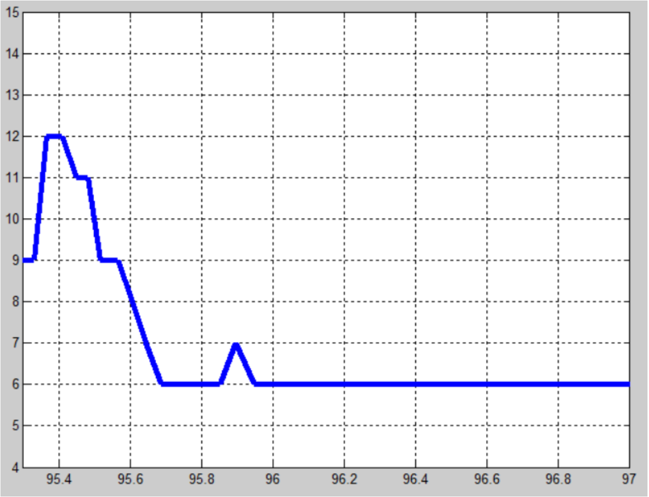
(a)


35
30
25
20
15
10
5
0
168
170
172
174
176
178
180
182
184
186

(b)


35
30
25
20
15
10
5
0
194
196
198
200
202
204
206
208
210
212

(c)



(d)

Figure 8: (a) result of first experiment, (b) result of second experiment, (c) result of third experiment, (d) result of fourth experiment

From the curve, it can tell the closed loop system is 2nd order or higher. Usually, for a closed loop system whose system order is higher than 2, it can be approximated as a 2nd order system. The general expression for a 2nd order system is [16]:

$$m\frac{d^2x}{dt^2} + b\frac{dx}{dt} + kx = 0 \tag{2}$$

Besides, a 2nd order system can be rewritten with only two coefficients as [16]:

$$\frac{d^2x}{dt^2} + 2\zeta w_n\frac{dx}{dt} + w_n^2 x = 0 \tag{3}$$

where, the natural frequency w_n can be defined as:

$$w_n = \sqrt{k/m} \tag{4}$$

In this case, $m = 0.041kg$ and $T \cong 0.5s$.

Thus, $\mathrm{w_n} = \frac{2\pi}{T} \cong 12.56\, rad/s$ and $k \cong 157.75\, N/m$.

The damping ratio can be approximately written as

$$\zeta \cong \ln\frac{previous\ peak\ of\ output}{current\ peak\ of\ output} \tag{5}$$

Therefore, in this case, $\zeta \cong 0.58$.

Besides, it has,

$$b \cong 2\zeta\sqrt{km} = 2.95kg/s \tag{6}$$

Various balls are tried in the experiments. Table 1 and figure 9 show the results and experiment setup, respectively. Due to the difference of size, mass, and friction between ball and rods, the coefficients vary. In practice, there are many uncertain factors affect the performance of the actual system, the try and trial results of PID coefficients usually will be different with the optimal theoretical results.

Table 1: PID coefficients

	Proportional coefficient	Integral coefficient	Derivative coefficient
Table tennis ball	0.25	0	0.2
Yellow sponge ball	0.17	0	0.2

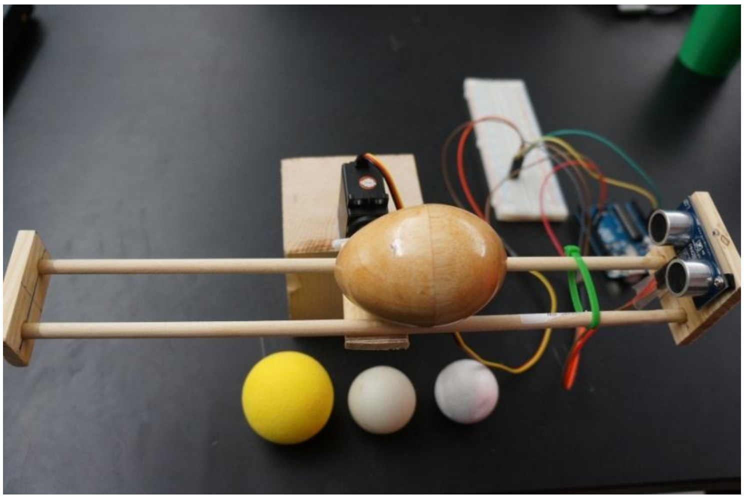


Figure 9: The setup of experiments with different balls

Conclusions

The ball balancing system is an inexpensive project that is designed as an educational tool for teaching control theory applications since the experiment involves all the representative components of a classic control system. This work allows the students to apply what they learnt to execute a sensor and microcontroller based control platform which is relevant to the closed loop feedback theory. Overall, the ball on beam system is effective for developing and demonstrating the related feedback control algorithm.

Acknowledgements

The authors sincerely thank Mr. Omar Danta for his great technical support. The authors sincerely thank Mr. Brandon Anderson and Mr. James Hutton for 3D printing the beams and sensor holder. The authors also thank Mr. Salman Nanji and Mr. Kim Ocampo for their initial work in the capstone design project.

References

[1] A. Ibrahim, 'Mechatronics Tutorials: Balancing of a Ball on Beam using Arduino as a PID controller', Mechatronicstutorials.blogspot.ca, 2014. Available: http://mechatronicstutorials.blogspot.ca/2014/07/balancing-of-ball-on-beam-using-arduino.html.

[2] B. B. Ghosh, B. K. Sarkar, R. Saha, "Realtime performance analysis of different combinations of fuzzy–PID and bias controllers for a two degree of freedom electrohydraulic parallel manipulator," Robotics and Computer-Integrated Manufacturing, Volume 34, August 2015, pp. 62-69

[3] Blogs.furman.edu, 'Arduino-controlled Ball-on-Beam with Interactive PID Interface « Furman Physics STEM Initiative', 2015. http://blogs.furman.edu/physicssteminitiative/2012/08/19/arduino-controlled-ball-on-beam-with-interactive-pid-interface

[4] C. Liaud, N.T. Nguyen, R. Nasreddine, S. Le Calvé, "Experimental performances study of a transportable GC-PID and two thermo-desorption based methods coupled to FID and MS detection to assess BTEX exposure at sub-ppb level in air," Talanta, Volume 127, 1 September 2014, pp. 33-42

[5] C. Rajalingham, R.B. Bhat, S. Rakheja, "Automatic balancing of flexible vertical rotors using a guided ball," International Journal of Mechanical Sciences, Volume 40, Issue 9, 1 September 1998, pp. 825-834

[6] F. Sartori, G. de Tommasi and F. Piccolo, "The Joint European Torus", IEEE Control Systems Magazine, vol. 26, no. 2, pp. 64-78, 2006.

[7] H. Lee, S. Jung, "Balancing and navigation control of a mobile inverted pendulum robot using sensor fusion of low cost sensors," Mechatronics, Volume 22, Issue 1, February 2012, pp. 95-105

[8] K. Sukvichai, M. Parnichkun, "Double-level ball-riding robot balancing: From system design, modeling, controller synthesis, to performance evaluation," Mechatronics, Volume 24, Issue 5, August 2014, pp. 519-532

[9] M. Abedini, M. A. Nojoumian, H. Salarieh, A. Meghdari, "Model reference adaptive control in fractional order systems using discrete-time approximation methods," Communications in Nonlinear Science and Numerical Simulation, Volume 25, Issues 1–3, August 2015, pp. 27-40

[10] M. N. Anwar, S. Pan, "A new PID load frequency controller design method in frequency domain through direct synthesis approach," International Journal of Electrical Power & Energy Systems, Volume 67, May 2015, pp. 560-569

[11] P.C.-P. Chao, Cheng-Kuo Sung, Chun-Lung Huang, Jeng-Sheng Huang, "Precision repositioning of the balancing ball in an auto-balancer system via a fuzzy speed regulator equipped with a sliding-mode observer," IEEE Transactions on Control Systems Technology, 2005, Volume: 13, Issue: 6, pp. 1107 – 1118

[12] P. Henaff, V. Scesa, F. B. Ouezdou, O. Bruneau, "Real time implementation of CTRNN and BPTT algorithm to learn on-line biped robot balance: Experiments on the standing posture," Control Engineering Practice, Volume 19, Issue 1, January 2011, pp. 89-99

[13] P. R. Vundavilli, D. K. Pratihar, "Soft computing-based gait planners for a dynamically balanced biped robot negotiating sloping surfaces," Applied Soft Computing, Volume 9, Issue 1, January 2009, pp. 191-208

[14] Q. Yang, E. Ong, J. Sun, G. Guo, S. Lim, "Study on the influence of friction in an automatic ball balancing system," Journal of Sound and Vibration, Volume 285, Issues 1–2, 6 July 2005, pp. 73-99

[15] S. Awtar, C. Bernard, N. Boklund, A. Master, D. Ueda, Kevin Craig, "Mechatronic design of a ball-on-plate balancing system," Mechatronics, Volume 12, Issue 2, March 2002, pp. 217-228

[16] Wikipedia.org/wiki/Damping_ratio

[17] Z. Gan, A. J. Hillis, J. Darling, "Adaptive control of an active seat for occupant vibration reduction, Journal of Sound and Vibration," Volume 349, August 2015, pp. 39-55

[18] Z. Gao, T. Wanyama, I. Singh, "Design a practice-intensive course of programmable logic controller and automation," International Journal of Mechatronics, Electrical and Computer Technology, Vol. 5(16) Jul. 2015, pp. 2207-2216