

A Portable PID Control Learning Tool by Means of a Mobile Robot

<http://dx.doi.org/10.3991/ijoe.v12i06.5716>

K. S. Chia and X. Y. Yap
Universiti Tun Hussein Onn Malaysia, Batu Pahat, Malaysia

Abstract—A proportional-integral-derivative (PID) controller is a classical controller that has been applied in numerous applications. One learning lesson of PID control theory is to tune its proportional, integral, and derivative parameters so that the performance of system is optimal. Besides, teaching PID control theory verbally is challenging especially when transient response characteristics e.g. overshoot, rise time, and settling time are introduced. Thus, this study investigates the feasibility of a low cost mobile robot in conveying the knowledge of PID control theory. First, an inexpensive open-source mobile robot was modified so that the position of the robot can be recorded and visualized wirelessly. Second, a graphical user interface was built to visualize the movement of the robot. Lastly, the PID parameters were tuned and their effects were recorded and analyzed quantitatively. Findings show that the proposed method is capable of demonstrating the effects of P and D parameters correctly.

Index Terms—PID tuning; mobile robot; control; education;

I. INTRODUCTION

One effective way to teach an abstract theory to engineering learners is using laboratory experiments in which the learners could link the abstract theory to the real world after completing a given experiment in the laboratory [1]. However, this conventional approach requires expensive resources e.g. space and equipment. Remote and virtual laboratory may be an effective affordable alternative in providing educative experiences to learners [2], the learning atmosphere by means of a computer display unit may not be exciting and interesting to some learners.

In order to attract the interest of learners, innovative education tools e.g. mobile robots have been proposed and investigated in engineering education at university level. This is because hands-on experience can make the learning process more exciting [3]. For instance, a self-balancing human transportation vehicle has been proposed as an active and inexpensive pedagogical tool that is capable of conveying feedback control theory and practice experiences to learners via various hands-on experiments [4]. Besides, a line following robot has been proposed in teaching fuzzy logic control theory that has enhanced the interest of learners [5]. This is in agreement with the findings in which more than 200 university students satisfied the use of mobile robots as teaching tools for 3 years [6]. Additionally, Chin et al. found that overall learning interest and motivation was better when robot-based learning systems were implemented, compared to a PowerPoint-based learning system [7]. Mc Lurkin et al. concluded that robots are effective inexpensive tools in recruiting, retaining, and training the next generation of engineers in sci-

ence, technology, engineering, and mathematics (STEM) topics after a three semesters of experiments [8].

A proportional-integral-derivative (PID) controller is one of popular classical controllers that has been applied in numerous applications e.g. temperature control [9]. One learning lesson of PID control theory is to tune its proportional, integral, and derivative parameters so that the performance of system is optimal. Teaching PID control theory verbally is challenging especially when transient response characteristics e.g. overshoot, rise time, and settling time are introduced. A better way to convey the knowledge of PID control theory and to improve the understanding of learners is worthy to be developed and investigated. Thus, this study aims to investigate the feasibility of a low mobile robot in conveying the knowledge of PID control theory.

II. METHODOLOGY

A. Demonstration Field

Figure 1 illustrates the demonstration field that used in this study with a continuous tracking line. The tracking line was a black line tape with a width of 1.7 cm. The background of the tracking line was a white tape with a width of 4.9 cm. The 10 bit analog reading from the IR sensors on the white and the black were zero and 800 analog values, respectively. The total length of the tracking line was 252 cm. The distance between the border and the tracking line was 8 cm. The challenge of this study was that the robot must control the direction and the speed of its motors in ways that the robot can continuously follow the given line that contains straight lines and 90 degree lines.

B. Mobile Robot

An Arduino-controllable tracked robot platform (Pololu Zumo Robot), also known as a differential wheeled robot, was used in this study. This robot consists of a Zumo shield, an Arduino UNO R3 microcontroller, two high speed 30:1 high power micro metal gear DC brushed motors, and a reflectance sensor array (six pairs of reflectance sensors). The robot was powered by a 7.4 V rechargeable lithium battery. The height, width, and length of the robot are 5, 7, and 7 cm, respectively, as that illustrated in Figure 2.

C. The reflectance Sensor Array

The six reflectance infrared (IR) sensors of the reflectance sensor array was directly connected to the analog pins of the microcontroller so that the position of the robot on a given line can be estimated more precisely. The height between the IR sensors and the demonstration field

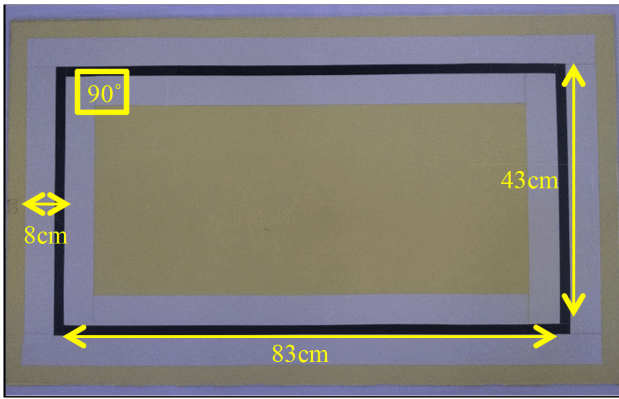


Figure 1. Top view of the demonstration field.

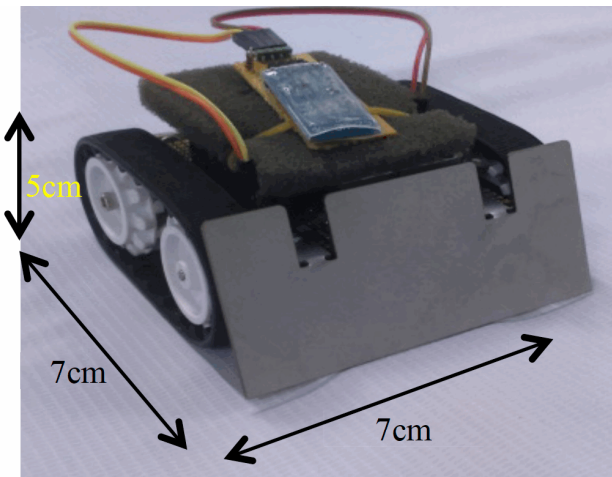


Figure 2. Robot Dimension.

is 0.7 cm. The distance between each IR sensor is 1.2 cm. The distance between the IR receiver and the IR transmitter is 0.3 cm. As a result, the black line would only be partially detected by two IR sensors simultaneously when the robot is moving along the given straight line.

D. Position of the Robot

From six pairs of reflectance sensors (i.e. R_0 , R_1 , R_2 , R_3 , R_4 , and R_5), the position of the line tracking robot on a given line can be modelled using the following formula.

$$\text{Position} = \frac{0 \times R_0 + 1000 \times R_1 + 2000 \times R_2 + 3000 \times R_3 + 4000 \times R_4 + 5000 \times R_5}{R_0 + R_1 + R_2 + R_3 + R_4 + R_5}$$

When the line was on the middle of the reflectance sensor array, the position value would be 2500, also known as the desired position. When the position of the line was approached to the left and the right of the reflectance sensor array, the position value would approach to zero and 5000, respectively.

The performance of the mobile robot was investigated on the demonstration field (Figure 1). The aim of the PID controller in this study was to ensure the robot would follow the given black line smoothly. The disturbance was introduced when the robot reached the 90 degree corner in which its position would be instantly varied significantly from its desired position. Consequently, the robot must counter this error properly so that the robot can tracking the line continuously on the demonstration field.

E. Data Acquisition

A Bluetooth module (HC-05) was integrated with the mobile robot as a wireless transceiver for sending and receiving data to and from a laptop computer via UART communication protocol. Another Bluetooth module (HC-05) was connected to a TTL-USB module that was plugged-in the USB port of the laptop computer so that the data received from the mobile robot would be recorded for analysis. The performance of the robot was recorded wirelessly in each 15 ms.

F. Proportional, Integration, and differentiation controller

Theoretically, the rise time of a system can be reduced by increasing the proportional parameter, K_p ; the overshoot and the settling time can be decreased by increasing the differentiation parameter, K_d , and the steady-state error can be removed using a suitable integral parameter, K_i . Thus, these transient response characteristics were investigated using the line tracking mobile robot in this study with the rise time from 0% to 100%, and the settling time of $\pm 2\%$.

A general PID manual tuning method was used in this study. First, K_p , K_i , and K_d were set to zero. After that, K_p was increased until the robot performed an oscillation when the robot was trying to correct back to its desired position immediately after the 90 degree corner. Next, the K_p was halved, and K_i was tuned to eliminate any steady-state error. Lastly, the K_d was tuned until a satisfactory overshoot and settling time were achieved.

III. RESULTS AND DISCUSSION

A. Effects of K_p to the performance of the mobile robot

Figure 3 illustrates the performance of the mobile robot that acquired wireless using the Bluetooth module for different K_p values when $K_i = 0$ and $K_d = 0$. When the robot reached the 90 degree corner, the value of the position was less than 1000 points. In order to reach its desired position of 2500 points, the robot with a good control system should be able to return to this desired position steadily.

Table 1 tabulates the transient response of the robot during K_p tuning. The results show that the overshoot of the mobile robot was increased from 486 units to 1117 units when the K_p value was increased from 0.15 to 0.30. However, the robot would be unstable and oscillating along the line when the K_p was more than 0.25. This observation is expected because a system will response faster when higher K_p is used, and a system will be unstable when its response is too fast.

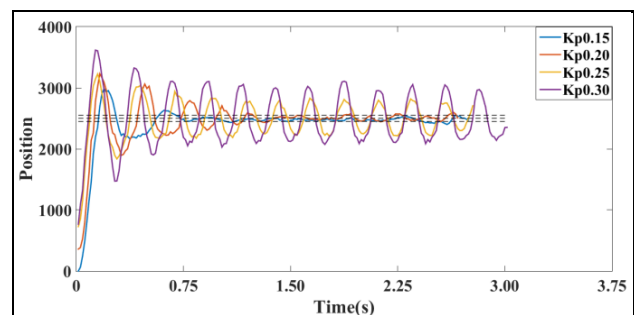


Figure 3. The performance of the mobile robot during K_d tuning when $K_i = 0$ and $K_p = 0.15$.

It is worth to highlight that the overshoot was slightly higher when K_p was 0.10, compared to that was 0.15. This was because one of the motors of the robot was not moved when the error was small. Consequently, the trajectory of mobile robot would be different and more time was needed when only one of its motors was moving while another motor was stopped to correct its position back to 2500 points.

The rise time of the robot was reduced when the K_p was increased. This is because the robot with higher K_p has higher rate of reaction toward its feedback error. The settling time (i.e. $\pm 2\%$ of the desired position) of the mobile robot, on the other hand, was fluctuated when different K_p was used between 0.100 and 0.225. After that, the performance of the robot was unstable due to its overreaction.

Since the robot performed a constant oscillation when K_p was 0.30, K_p of 0.15 (i.e. K_p equals 0.30 multiplied by 0.5) was used for K_d tuning. K_i was set to zero because the offset was zero.

B. Effects of K_d to the performance of the mobile robot

Figure 4 illustrates that the overshoot and settling time of the mobile robot were gradually reduced when the K_d was increased from 0.01 to 0.15. This indicates that the performance of the robot could be gradually improved by using a suitable K_d value. The rise time, on the other hand, was progressively increased when the K_d was increased. This demonstrates the side effect of higher K_d value that causes a system responses slower.

Table II tabulates the performance of the mobile robot during K_d tuning when $K_i = 0$ and $K_p = 0.15$. When the K_d was increased from 0.01 to 0.15, the overshoot and the settling time was reduced. This indicates that the robot was able to achieve its desired position faster when the K_d was increased. However, the robot that used K_d more than 0.15 was taking too much time to reach its desired position. Thus, the mobile robot that used K_p of 0.15 and K_d of 0.15 achieved the best performance with zero overshoot and a settling of 0.392 seconds in this study.

IV. CONCLUSION

This paper presents an alternative way to demonstrate the effects of K_p and K_d in PID tuning by means of the mobile robot. The performance of the mobile robot could be directly observed in-situ when different K_p and K_d values were applied. This could provide real experience to learners by observing the movement of the mobile robot when it was tracking the given line with specific K_p and K_d values. Besides, the results show that the effects of different K_p and K_d values toward the mobile robot are in-line with the control theory, i.e. the higher K_p value, the higher the overshoot and shorter the rise time; and the overshoot of the system can be reduced using an optimal K_d value. Thus, the proposed system is promising to be an alternative in conveying the knowledge of PID control theory.

Additionally, the performance of the mobile robot could be analyzed quantitatively using the wirelessly acquired data graphically, in which, the rise time, settling time and the amount of the overshoot could be visualized and computed. However, the proposed learning tool is unable to demonstrate the effect of K_i because the proposed system does not contain an offset.

TABLE I.
THE OVERSHOT, SETTLING TIME, AND RISE TIME WITH DIFFERENT K_p VALUE WHEN $K_i = 0$ AND $K_D = 0$.

K_p	Overshoot, M_p	Settling Time, T_s (s)	Rise Time, T_r (s)
0.100	597	0.810	0.210
0.150	486	0.690	0.135
0.175	594	0.650	0.108
0.200	730	1.329	0.103
0.225	699	0.963	0.093
0.250	726	N/A	0.080
0.275	891	N/A	0.077
0.300	1117	N/A	0.071
0.500	N/A	N/A	N/A

TABLE II.
THE OVERSHOT, SETTLING TIME, AND RISE TIME WITH DURING K_d TUNING WHEN $K_i = 0$ AND $K_p = 0.15$.

K_d	Overshoot, M_p	Settling Time, T_s (s)	Rise Time, T_r (s)
0.01	697	0.725	0.124
0.05	344	0.485	0.136
0.10	76	0.405	0.153
0.15	0	0.392	0.165
0.20	N/A	0.596	0.307
0.50	N/A	0.750	0.334

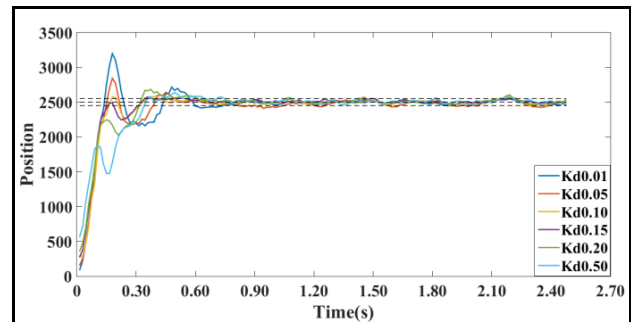


Figure 4. The performance of the mobile robot during K_d tuning when $K_p = 0.15$ and $K_i = 0$.

In future, a lab sheet will be developed so that the designed PID learning tool by means of the mobile robot will be evaluated based on the learning performance of students.

ACKNOWLEDGMENTS

The author would like to acknowledge Universiti Tun Hussein Onn Malaysia (UTHM) for providing financial support (Vot U351), and Advanced Mechatronic Research Group (AdMiRe), Faculty of Electrical and Electronic Engineering, UTHM for providing facilities for this study.

REFERENCES

- [1] E. d. Klerk and I. K. Craig, "A laboratory experiment to teach closed-loop system identification," *IEEE Transactions on Education*, vol. 47, pp. 276-283, 2004. <http://dx.doi.org/10.1109/TE.2004.825524>
- [2] Y. Kuchirka, M. Petriv, S. Baran, D. Ursutiu, and C. Samoila, "Complementing Education via Virtual Labs: Implementation and Deployment of Remote Laboratories and Usage Analysis in South Indian Villages," *International Journal of Online Engineering*, vol. 12, pp. 4-7, 2016. <http://dx.doi.org/10.3991/ijoe.v12i03.5233>

SHORT PAPER
A PORTABLE PID CONTROL LEARNING TOOL BY MEANS OF A MOBILE ROBOT

- [3] J. Vaughan, J. Fortgang, W. Singhose, J. Donnell, and T. Kurfess, "Using mechatronics to teach mechanical design and technical communication," *Mechatronics*, vol. 18, pp. 179-186, 5// 2008.
- [4] S. C. Lin and C. C. Tsai, "Development of a Self-Balancing Human Transportation Vehicle for the Teaching of Feedback Control," *IEEE Transactions on Education*, vol. 52, pp. 157-168, 2009. <http://dx.doi.org/10.1109/TE.2008.921799>
- [5] D. Ibrahim and T. Alshanableh, "An undergraduate fuzzy logic control lab using a line following robot," *Computer Applications in Engineering Education*, vol. 19, pp. 639-646, 2011. <http://dx.doi.org/10.1002/cae.20347>
- [6] F. Mondada, M. Bonani, X. Raemy, J. Pugh, C. Cianci, A. Klaptocz, et al., "The e-puck, a Robot Designed for Education in Engineering," presented at the 9th Conference on Autonomous Robot Systems and Competitions, Castelo Branco, Portugal, 2009.
- [7] K. Y. Chin, Z. W. Hong, and Y. L. Chen, "Impact of Using an Educational Robot-Based Learning System on Students' Motivation in Elementary Education," *IEEE Transactions on Learning Technologies*, vol. 7, pp. 333-345, 2014. <http://dx.doi.org/10.1109/TLT.2014.2346756>
- [8] J. McLurkin, J. Rykowski, M. John, Q. Kaseman, and A. J. Lynch, "Using Multi-Robot Systems for Engineering Education: Teaching and Outreach With Large Numbers of an Advanced, Low-Cost Robot," *IEEE Transactions on Education*, vol. 56, pp. 24-33, 2013. <http://dx.doi.org/10.1109/TE.2012.2222646>
- [9] X.-I. Xie and Z. Long, "Fuzzy PID Temperature Control System Design Based on Single Chip Microcomputer," *International Journal of Online Engineering*, vol. 11, 2015.

AUTHORS

K. S. Chia obtained his PhD and BEng from Universiti Teknologi Malaysia in Feb. 2014 and Sept. 2010, respectively. He was an assistant professor at Southern University College, Skudai, Malaysia from March 2014 to August 2014. He has served as a senior lecturer in the Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia, 86400 Batu Pahat, Malaysia since September 2014. His research interests are Machine Learning, Artificial Intelligence, Microcontroller, Soft Modeling, and Near Infrared Spectroscopic Analysis (e-mail: kschia@uthm.edu.my, url: <https://sites.google.com/site/kschia86/>).

X. Y. Yap is currently pursuing the BEng in Mechatronic and Robotic Engineering in the Faculty of Electrical and Electronic Engineering, Universiti Tun Hussein Onn Malaysia, 86400 Batu Pahat, Malaysia. His research interests are data analysis and machine learning (e-mail: wymvernyp@gmail.com).

This work was financially supported by Universiti Tun Hussein Onn Malaysia (UTHM) (Vot U351). The author would like to acknowledge Universiti Tun Hussein Onn Malaysia (UTHM) and Advanced Mechatronic Research Group (AdMiRe), Faculty of Electrical and Electronic Engineering, UTHM for providing facilities for this work.

Submitted, 01 April 2016. Published as resubmitted by the authors on 10 May 2016.