

Ziegler-Nichols Based Proportional-Integral-Derivative Controller for a Line Tracking Robot

Kim Seng Chia

Faculty of Electrical and Electronic Engineering,
Universiti Tun Hussein Onn Malaysia, 86400 Parit Raja, Batu Pahat, Malaysia

Article Info

Article history:

Received July 19, 2017

Revised Oct 20, 2017

Accepted Nov 7, 2017

Keywords:

Differential Wheeled Robot

Line Tracking

Proportional-Integral-

Derivative

Ziegler Nichols

ABSTRACT

Line tracking robots have been widely implemented in various applications. Among various control strategies, a proportional-integral-derivative (PID) algorithm has been widely proposed to optimize the performance of a line tracking robot. However, the motivation of using a PID controller, instead of a proportional (P) or a proportional-integral (PI) controller, in a line tracking task has seldom been discussed. Particularly, the use of a systematic tuning approach e.g. closed loop Ziegler Nichols rule to optimize the parameters of a PID controller has rarely been investigated. Thus, this paper investigates the performance of P, PI, and PID controllers in a line tracking task, and the ability of Ziegler Nichols rule to optimize the parameters of the P, PI, and PID controllers. First, the ultimate gain value, K_u and ultimate period of oscillation, P_u were estimated using a proposed approach. Second, the values of K_p , K_i and K_d were estimated using the Ziegler Nichols formulae. The performance of a differential wheeled robot in the line tracking task was evaluated using three different speeds. Results indicate that the Ziegler Nichols rule coupled with the proposed method is able to identify the parameters of the P, PI, and PID controllers systematically in the line tracking task. Findings indicate that the mobile robot coupled with a proportional controller achieved the best performance compared to PI and PID controllers in the line tracking process when the estimated initial parameters were used.

Copyright © 2018 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

Kim Seng Chia,

Faculty of Electrical and Electronic Engineering,

Universiti Tun Hussein Onn Malaysia, 86400 Parit Raja, Batu Pahat, Malaysia.

Email: kschia@uthm.edu.my

1. INTRODUCTION

Line follower or line tracking robots have been widely implemented in various applications e.g. delivery services, transportation systems, blind assistive, and educational [1] applications. This could be due to its simplicity and reliability in terms of its design and performance. A line follower robot is a self-operating robot that is designed to work on a given line. Generally, the speed and direction of a line follower robot can be controlled using a simple logic that based on the state of sensors [2, 3]. However, this might be less flexible due to the limited states that are available for a combination of a few sensors.

A proportional-integral-derivative (PID) controller is one of popular classical controllers in numerous applications. One of the challenges for a PID controller is to obtain the optimal values of its proportional, integral, and derivative parameters. Even though abundant strategies have been proposed to automatically tune these parameters [4], e.g. Artificial Bee Colony [5]. However, these strategies are rarely to be applied due to their complexity. Consequently, classical PID tuning methods, e.g. Ziegler Nichols rule, are still widely applied in industries due to their robustness and simplicity [6, 7].

Among various control strategies, a proportional-integral-derivative (PID) controller has been widely proposed to optimize the performance of a line tracking robot [8, 9]. However, the motivation of using a PID controller, instead of a proportional (P) or proportional-integral (PI) controller, in a line tracking task has seldom been discussed. Besides, the tuning process of PID parameters has not been fully understood. This is because a trial-and-error practice has been used in most cases.

The closed loop Ziegler Nichols rule is a systematic tuning approach that has been widely used in industries to optimize the parameters of a PID controller. In fact, Ziegler Nichols rule has been widely used as a benchmark to tune PID parameters in numerous studies [10, 11]. This could be due to the fact that the implementation of the closed loop Ziegler Nichols rule does not need the information of a model, and it is likely to achieve satisfactory performance. Consequently, various GUI or simulator have been developed to observe the effect of Ziegler Nichols tuning method [7, 12].

Despite the simplicity and robustness of Ziegler-Nichols rule, the use of Ziegler-Nichols rule to find the parameters of a PID controller in a line tracking robot has rarely been reported and investigated. One of the challenges of applying the closed loop Ziegler-Nichols rule in a line tracking robot is to figure out the ultimate gain, K_u and ultimate period of oscillation, P_u . The ultimate gain, K_u and ultimate period of oscillation, P_u are two crucial parameters that can only be computed when a system is performing a simple harmonic oscillation. Thus, a strategy is needed to obtain the K_u and P_u so that Ziegler Nichols rule can be correctly implemented to estimate the parameters of a PID controller for a line tracking robot.

Thus, this paper aims to investigate the ability of the Ziegler Nichols rule to estimate the initial parameters of the P, PI, and PID controllers, and to evaluate the performances of P, PI, and PID controllers in controlling a mobile robot for a line tracking task when the estimated parameters are used.

2. RESEARCH METHOD

2.1. Differential Wheeled 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); and is powered by four AA rechargeable batteries.

2.2. Testing Field

A 50cm × 75cm white cardboard was used as the background of the testing field. A black masking tape with a width of 2cm was used to construct a 164cm continuous tracking line as illustrated in Figure 1. A continuous tracking line was used so that the performance of the line tracking robot can be observed continuously in a limited space.

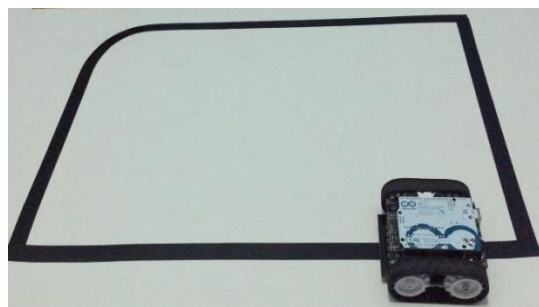


Figure 1. The mobile robot on the testing field

2.3. Line Tracking Algorithm

Matlab (R2013b) Simulink coupled with ZumoBotLib (adafruit) was used to tune the parameters of P, PI, and PID controllers of the mobile robot. The signals of the reflectance sensor array were calibrated with a value from zero to 5000. The reading would approach to zero, 2500, and 5000 when the black line was on the left, the middle, and the right of the reflectance sensor array. In order to track a given line, the aim of the robot is to maintain its desired position (i.e. the value of 2500) with a minimum deviation by adjusting the speeds of its left and right DC motors according to the Equation 1 and Equation 2.

$$V_L = V - \omega L/2 \tag{1}$$

$$V_R = V + \omega L/2 \tag{2}$$

Where, V_L and V_R are the speeds of the left and right motors, respectively; V denotes the desired speed; L is the axial length of the robot; and ω is the output of the P, PI, or PID controller that is used to adjust the speeds of the DC motors. A good controller should estimate a suitable ω continuously so that the speed of both left and right motors can be adjusted properly in such ways that the robot can follow the given line with a minimum deviation.

2.4. Ziegler Nichols

In this study, a closed-loop Ziegler Nichols rule was used to find the parameters of a P, PI, or PID controller for the mobile robot in line tracking process. First, the reflectance sensor array was calibrated to differentiate the black line and white background. Second, both integral, K_I and derivative, K_D parameters were set to zero so that only the proportional parameter, K_P was used to perform a proportional controller for the line tracking process. Third, the speed of the mobile robot was set to 25 cm/s. Then, the value of K_P was gradually increased until the mobile robot was oscillating with an approximately consistent amplitude as illustrated in Figure 2.

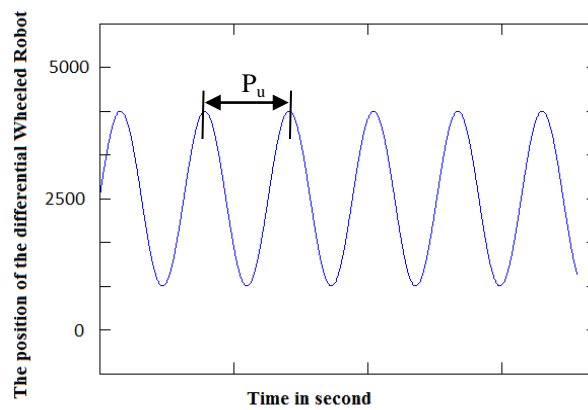


Figure 2. The position of the differential wheeled robot versus the time

After that, the movement of the mobile robot was captured using a video camera. From the captured video, the time taken for the mobile robot to complete 20 stable oscillations continuously was recorded. After that, the ultimate period, P_u can be approximated by dividing the time taken by 20. This strategy was used to estimate the P_u because the time taken to complete an oscillation is less than two seconds. The ultimate gain K_c , on the other hand, is the same as the K_p that caused the consistent oscillation. The above procedure was repeated using two different speeds of 50 cm/s and 75 cm/s.

After completing the experiment with three different speeds, the estimated P_u and K_u values were used to estimate the initial parameters of P, PI, and PID controllers for each speed using the classical Ziegler Nichols formulae as that tabulated in Table 1.

Table 1. The formulae used to estimate the values of K_p , K_I and K_D for P, PI, and PID controllers using K_u and P_u

Controller	K_p	K_I	K_D
P	$K_u \div 2$	0	0
PI	$K_u \div 2.2$	$1.2K_p \div P_u$	0
PID	$K_u \div 1.7$	$2K_p \div P_u$	$K_p P_u \div 8$

3. RESULTS AND ANALYSIS

3.1. Different Speeds

Table 2 tabulates the values of K_u and P_u for the mobile robot with different speeds. The ultimate gain, K_u was increased when the speed of the mobile robot was increased. This suggests that the mobile robot with a higher speed requires a higher K_p value to maintain a consistent oscillating movement.

Ultimate period, P_u , on the other hand, did not indicate a particular pattern as K_u . For example, P_u was the lowest when the speed of 50 cm/s was used; but higher P_u was needed for either speed of 25 or 75 cm/s. This could be due to the fact that the consistent oscillating paths for each speed was different in terms of total distance and amplitude. Consequently, different period was needed to complete one circle for a combination of a different speed and a different ultimate gain.

Table 2. The values of K_u and P_u for the mobile robot with different speeds

Speed (cm/s)	Ultimate gain, K_u	Ultimate period, P_u (second)
25	0.056	1.40
50	0.060	1.25
75	0.070	1.71

3.2. P, PI, and PID Controllers

Table 3 shows that P, PI, and PID controllers with different parameters were estimated because the mobile robot with different speeds had different values of the ultimate gain and ultimate period. This should be expected because different K_p and P_u were needed for the mobile robot to perform a stable oscillation when the speed of the robot was different.

Table 3. The parameters of P, PI, and PID controllers for the mobile robot with different speeds

Speed (cm/s)	Controller	K_p	K_i	K_D
25	P	0.028	0	0
	PI	0.025	2	0
	PID	0.033	46	6
50	P	0.030	0	0
	PI	0.027	08	0
	PID	0.035	65	6
75	P	0.035	0	0
	PI	0.032	25	0
	PID	0.041	55	4

Table 4 tabulates the performance of the mobile robot for completing a circle of the line tracking application when different speeds and controllers were applied. By inspection, the mobile robot that used a proportional controller achieved the best performance among other controllers for the three different speeds. Interestingly, the mobile robot that used the initial estimated PID controller produced the worst performance for the three different speeds. This suggests that the Ziegler Nichols rule has successfully estimated a good value for proportional controller only for the application. Manual optimization based on a rule of thumb or a heuristic approach is needed to optimize the performance for PI and PID controllers.

It is worth to highlight that an integral controller is commonly used to remove an existing offset. However, an offset might not exist in this line tracking process. In other words, adding an integral controller could be redundant, and consequently it could degrade the performance of the mobile robot. This is possible because an integral controller tends to slow down the response of a controller. For example, the mobile robot that used a PI controller is only able to achieve similar or worse performance compared to that used P controller for the three. However, only the mobile robot that applied P and PI controllers demonstrated this expected performance. The mobile robot that used PID for a speed of 75 cm/s, in contrast, achieved the worse performance than that applied PID for a speed of 50 cm/s. This is because the mobile robot with the

speed of 75cm/s had the higher rising time and significant damping oscillations when the robot turned 90 degrees. Consequently, more time was needed to complete the given task.

Table 4. The time taken for the mobile robot to complete a cycle in the testing field

Speed (cm/s)	Controller	Time taken (s)
25	P	9.33
	PI	9.33
	PID	11.00
50	P	5.00
	PI	5.33
	PID	6.67
75	P	4.33
	PI	4.33
	PID	8.00

The main challenge of a closed-loop Ziegler Nichols tuning rule is to accurately estimate the ultimate period, P_u and the ultimate gain K_c . Recent study uses a visual inspection based on whether the mobile robot performed a noticeable oscillation without wild ones during line tracking [13]. However, this approach trends to be less reliable and subjective. On the other hand, the proposed approach by means of a video camera that records the observation can eliminate the unwanted uncertainty in estimating the P_u and K_c . Consequently, the repeatability and reproducibility of a study can be improved.

4. CONCLUSION

Findings indicate that the Ziegler-Nichols rule is a systematic approach to estimate the parameters of P, PI, and PID controllers for the line tracking process. This can be accomplished by using the approximated ultimate gain, K_u and ultimate period of oscillation, P_u to estimate these parameters based on the Ziegler-Nichols' formulae. In other words, a trial-and-error tuning method to estimate the parameters of the controllers can be avoided. Nevertheless, background knowledge is needed to identify a suitable controller to suit particular applications. The best controller for the line tracking process was the P controller among PI and PID controllers.

The higher the speed of a mobile robot, the higher the K_p value was required to maintain a consistent oscillating movement. The best performance was achieved by P controllers for the three different speeds. Findings also show that the selection of the type of controllers is crucial to achieve the best performance. Particularly, the use of a complex controller, i.e. PID controller, did not achieve the best performance when the estimated parameters by means of the Ziegler-Nichols rule. Thus, the selection of a suitable controller should be made according to the nature of a given system. Besides, a simple controller, e.g. P controller, should be considered first before more complex controllers are implemented.

ACKNOWLEDGMENTS

The author would like to acknowledge Research and Innovation Fund provided by the Office for Research, Innovation, Commercialization and Consultancy Management (ORICC), RMC, Universiti Tun Hussein Onn Malaysia (UTHM) for providing financial support, and Faculty of Electrical and Electronic Engineering, UTHM for providing facilities for this study; and Mr Amir Yasser for acquiring the data needed in this study.

REFERENCES

- [1] Ibrahim, D. and Alshangleh, T. An undergraduate fuzzy logic control lab using a line following robot. *Computer Applications in Engineering Education*, 2011; 19(4): 639-646.
- [2] Pakdaman, M. and Sanaatiyan, M.M. *Design and Implementation of Line Follower Robot*. Second International Conference on Computer and Electrical Engineering. 2009; 2: 585-590.
- [3] Hasan, K.M., Al-Nahid, A., and Al Mamun, A. *Implementation of autonomous line follower robot*. International Conference on Informatics, Electronics & Vision (ICIEV). 2012: 865-869.
- [4] Kiam Heong, A., Chong, G., and Yun, L. PID control system analysis, design, and technology. *IEEE Transactions on Control Systems Technology*, 2005; 13(4): 559-576.
- [5] Elsis, M., Soliman, M., Aboelela, M., and Mansour, W. ABC Based Design of PID Controller for Two Area Load Frequency Control with Nonlinearities. *Indonesian Journal of Electrical Engineering and Computer Science*, 2015; 16(1): 58-64.

- [6] Mallesham, G., Mishra, S., and Jha, A.N. *Ziegler-Nichols based controller parameters tuning for load frequency control in a microgrid*. International Conference on Energy, Automation, and Signal (ICEAS). 2011: 1-8.
- [7] Abdulameer, A., Sulaiman, M., Aras, M., and Saleem, D. Tuning Methods of PID Controller for DC Motor Speed Control. *Indonesian Journal of Electrical Engineering and Computer Science*, 2016; 3(2): 343-349.
- [8] Fu-hua, J. and Bao Trung, M. *Building an autonomous line tracing car with PID algorithm*. 10th World Congress on Intelligent Control and Automation (WCICA). 2012: 4478-4483.
- [9] Chia, K.S. and Yap, X.Y. A Portable PID Control Learning Tool by Means of a Mobile Robot. *International Journal of Online Engineering*, 2016; 12(6): 54-57.
- [10] Korkmaz, M., Aydogdu, O., and Dogan, H. *Design and performance comparison of variable parameter nonlinear PID controller and genetic algorithm based PID controller*. International Symposium on Innovations in Intelligent Systems and Applications (INISTA). 2012: 1-5.
- [11] Kanojiya, R.G. and Meshram, P.M. *Optimal tuning of PI controller for speed control of DC motor drive using particle swarm optimization*. International Conference on Advances in Power Conversion and Energy Technologies (APCET). 2012: 1-6.
- [12] Abdulameer, A., Sulaiman, M., Aras, M., and Saleem, D. GUI Based Control System Analysis using PID Controller for Education. *Indonesian Journal of Electrical Engineering and Computer Science*, 2016; 3(1): 91-101.
- [13] Ding, J., Li, Z., and Pan, T. Control System Teaching and Experiment Using LEGO MINDSTORMS NXT Robot. *International Journal of Information and Education Technology*, 2017; 7(4): 309-313.