PID Parameter Tuning of a Low-Cost DC Motor Speed Control for Mobile Robot Application

Munkh-Erdene Ayurzana
Mongolian University of Science and Technology, 8th khoroo, Baga toiruu 34, Sukhbaatar district Ulaanbaatar, Mongolia 14191

Erkhembayar Gankhuyag
Mongolian University of Science and Technology, 8th khoroo, Baga toiruu 34, Sukhbaatar district Ulaanbaatar, Mongolia 14191

Dondogjamts Batbaatar
Mongolian University of Science and Technology, 8th khoroo, Baga toiruu 34, Sukhbaatar district Ulaanbaatar, Mongolia 14191

Naranbaatar Erdenesuren
Mongolian University of Science and Technology, 8th khoroo, Baga toiruu 34, Sukhbaatar district Ulaanbaatar, Mongolia 14191

Email: B232340036@must.edu.mn

Abstract

Precise control of DC motors is vital for robotics, industrial automation, and mechatronics. Traditional PID (Proportional-Integral-Derivative) control methods, while widely used, often require offline parameter tuning, which can be time-consuming and suboptimal for real-time applications. This paper proposes a GUI (Graphical User Interface) based approach for PID tuning of a DC motor model. The proposed method utilizes a MATLAB-based parameter estimation model with GUI to continuously monitor and update PID controller parameters based on real-time data from the Arduino-controlled DC motor setup.

Keywords: PID parameter tuning, Graphical user interface (GUI), DC motor control

1. Introduction

Despite its simplicity, Proportional-Integral-Derivative (PID) control remains remarkably efficient, offering a three-term solution to address both transient and steady-state responses in numerous real-world problems. Since its invention in 1910, largely attributed to Elmer Sperry's ship autopilot, and the development of straightforward tuning methods by Ziegler and Nichols in 1942 [1], [2]. PID control has witnessed widespread adoption [3]. DC motors are widely employed as actuators for wheeled mobile robots, robots controlled by programmed microcontrollers, or computers that navigate autonomously based on sensor information. These robots typically have two, three, or four wheels, each driven by a DC motor for smooth maneuverability [4], [5], [6]. Two primary control applications for DC motors are speed and position control. Speed controllers regulate robot movement speed, while position controllers ensure accurate movement to predetermined positions. [7], [8]. Recent research on DC motor speed control focuses on stabilization, with many works combining PID controllers with other methods like fuzzy logic control, genetic algorithms [1],

and particle swarm optimization. Similarly, PID-based position controllers have been combined with artificial intelligence techniques like fuzzy logic and genetic algorithms to enhance accuracy [9], [10], [11]. While PID control exhibits remarkable effectiveness, optimal tuning remains a challenge, particularly in dynamic environments encountered by wheeled mobile robots. However, these approaches often lack user-friendliness and adaptivity.

This study utilizes one of the DC motors from a two-wheeled mobile robot as the plant. The DC motor's mathematical model is developed by acquiring input and output signals from an open-loop experiment [10]. The applied working voltage serves as the input signal, and the electronically measured speed is the output signal. The resulting model's step response is compared to the original motors to validate its accuracy. A PID-based speed controller is proposed, with initial parameters obtained from the relay feedback experiment and MATLAB based PID tuning integrated with GUI. These parameters are then manually fine-tuned based on practical PID tuning knowledge to achieve optimal performance without overshoot or steady-state error. The paper is structured as follows: Section 1 presents the background and related work, Section 2 outlines the proposed approach and methodology, Section 3 details the implementation and results, and Section 4 concludes the study and highlights its potential impact.

2. Methodology

2.1. *Mathematical model of DC motor*

Before we create a GUI interface, we need to start off with a Simulink model of the DC motor system with PI controller. The DC motor has to be modeled first, in Simulink and the mathematical equations behind the functioning of the motor has to be understood before advancing further

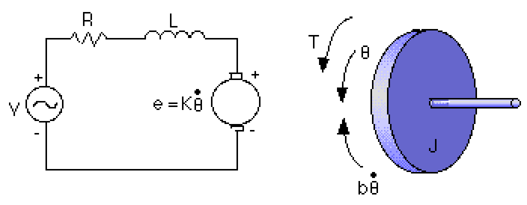

Fig.1 DC Motor model

In order to model a DC Motor in Simulink, we have to define the mathematical equations on which it functions, The DC Motor functioning diagram is given in Fig.1.

2.2. *Electrical Characteristics*

By using Kirchoff's voltage law in the DC-motor model according to Fig. 1 DC motor model following equation is derived

$$V_{in} - V_{Ra} - V_{La} - V_{emf} = 0 \quad (1)$$

Where, V_{in} , V_{Ra} , V_{La} and V_{emf} is the input-voltage, voltage over the armature resistance, voltage drop over the armature inductance and voltage induced by the coil respectively. The equations for the voltage parts in the DC-motor are,

$$V_{Ra} = i_a \cdot R_a \quad (2)$$

$$V_{La} = L_a \frac{d}{dt} i_a \quad (3)$$

$$V_{emf} = K_e \cdot \dot{\theta}_a \quad (4)$$

2.3. *Mechanical Characteristics* [2]

By torque balance (energy balance) in the system the mechanical equations can be stated.

$$T_e - T_{\ddot{\theta}a} - T_b - T_L = 0 \quad (5)$$

Where T_e is the electromagnetic torque, $T_{\ddot{\theta}_a}$, is torque generated from the rotational acceleration of rotor. T_b, is the torque due to the friction and angular velocity in the motor. T_L is the torque of mechanical load (external load), equations for the first three parts are

$$\boldsymbol{T_e = K_t \cdot i_a} \quad (6)$$

$$\boldsymbol{T_{\ddot{\theta}_a} = J_m \cdot \frac{d}{dt}\dot{\theta}_a} \quad (7)$$

$$\boldsymbol{T_b = b_m \cdot \dot{\theta}_a} \quad (8)$$

Substituting equation (5) with (6), (7) and (8) yields following differential equation,

$$V_{in} - i_a \cdot R_a - L_a \cdot \frac{d}{dt} i_a - K_e \cdot \dot{\theta}_a = 0 \quad (9)$$

$$\boldsymbol{\frac{d}{dt} i_a = \frac{1}{L_a} \cdot (i_a \cdot R_a + K_e \cdot \dot{\theta}_a - V_{in})}$$

(10)

3. Results and Discussion

The experiment's foundation lay in meticulous data collection, enabling the analysis of the intricate interplay between voltage and RPM. A 140 second data acquisition process captured the voltage profile, characterized by a 20-second ramp-up to 24V followed by a 20-second minimum, and this pattern mirrored the observed RPM variations with relay feedback experiment was shown in Fig. 2.

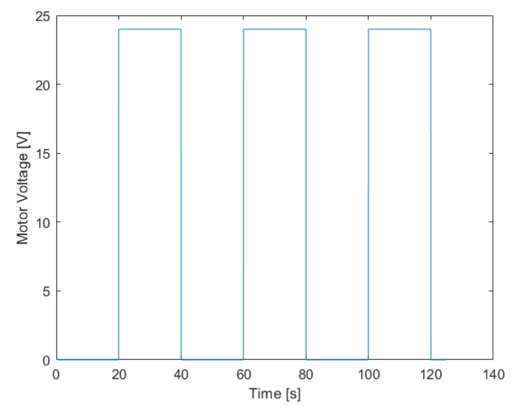

Fig.2 Relay output

From the relay feedback experiment, the following result was obtained.

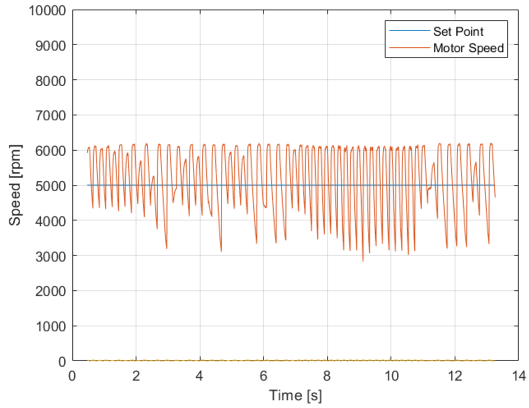

Fig.3 System without PID tuning

The open-loop experiment Fig. 3 revealed significant overshoot and sustained oscillations in the DC motor speed due to suboptimal PID settings. Manually configuring the parameters or using the time-consuming Ziegler-Nichols method might not be ideal for real-time applications.

3.1 *Parameter estimation and tuning*

From the mathematical model of DC motor following initial parameters are to be known such as moment of inertia of the rotor J_m, damping ratio of the mechanical system b_m, electric resistance R_m, electric inductance L_m, armature constant K_t, before being tuned with a PID controller. While manufacturers may provide values for some of these DC motor parameters, they are only approximate. Its necessary to estimate these parameters as precisely as possible for our model to ascertain whether it is an accurate representation of the actual DC motor system.

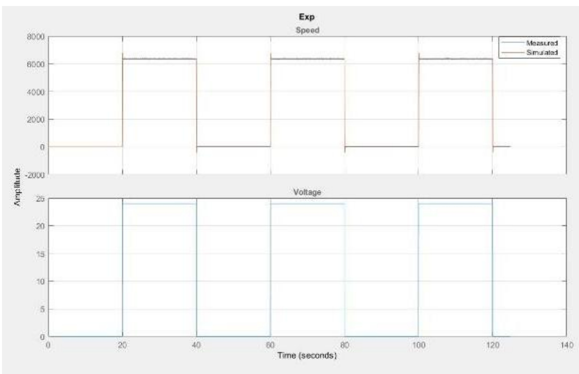

Fig.4 Parameter estimation plot

Again, these values are of important for designing and tuning a PID controller for the DC motor. The PID controller will use these parameters to calculate the motor's input voltage in order to achieve the desired speed. Furthermore, without knowing the specific values of the parameters, it is difficult to say much about the specific performance of the DC motor. However, the fact that these are estimated parameters (Fig. 5) suggests that the motor has not been precisely modeled or characterized [12], [13]. This could lead to difficulties in tuning the PID controller effectively.

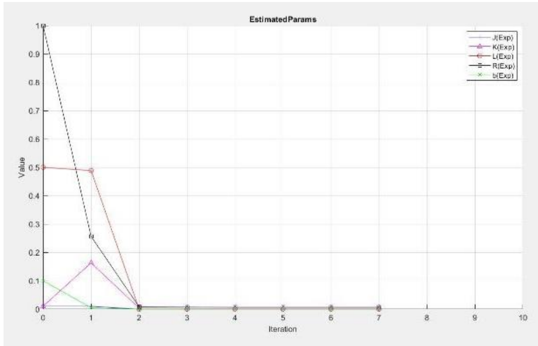

Fig. 5 Iteration for finding motor parameters

As shown in Fig. 4 the measured data overlaid with the simulated data. The simulated data comes from the model with the estimated parameters listed in Table 1. Comparing the response of the system before and after the estimation process clearly shows that the estimation successfully identified the model parameters and the simulated response accurately matches the experimental data.

Table 1. DC Motor parameters

Constants	Value	Unit
K_t	0.0035401	$[N \cdot m/A]$
R_m	0.0065388	$[Ohm]$
L_m	8.758e-05	$[H]$
b_m	0.0001263	$[-]$
J_m	4.3114e-05	$[kg \cdot m^2]$

While from PID tuning result following initial PID parameters are determined in Table 2.

Table 2. Parameters used in the numerical simulation.

Controller	PID Parameters		
	K_p	K_i	K_d
PID	0.01	0.061	0

After the fine-tuning process in MATLAB, it was found that the final K_p = 0.01, K_i = 0.061 and K_d = 0. The initial value of Kp = 0.9 gives big oscillation output. The fine tuning is performed to adjust the K_p, such that the P controller output results in only one overshoot and no oscillation, although steady state error is still not zero. This final value of Kp is 0.01.

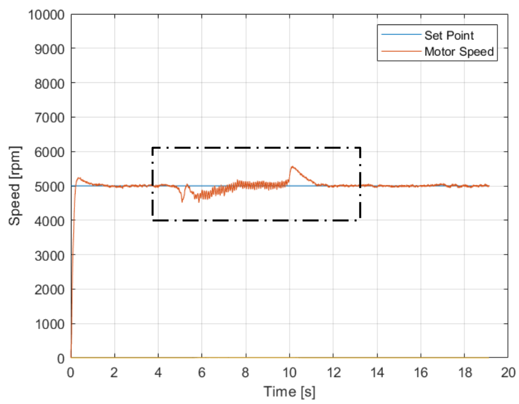

Fig.6 System with PID tuning

The Fig. 6 illustrates the response of a DC motor speed control performance after it has been tuned with PID controller. The blue line shows the reference input, the red line shows the actual output speed of the motor, which initially lags behind the reference input and then abruptly rise to meet it. There are some oscillations in the motor speed, which indicates that the system is damped due external disturbances or externally given mechanical load. The system is returned to stable state approximately after 1 second under the varying operating conditions.

3.2. GUI for motor control

In order to simplify the dc motor control process, a GUI is developed that allows a user to quickly identify model, tune controller coefficients and send reference data was created. Two types of GUIs were created.

a) GUI for identification

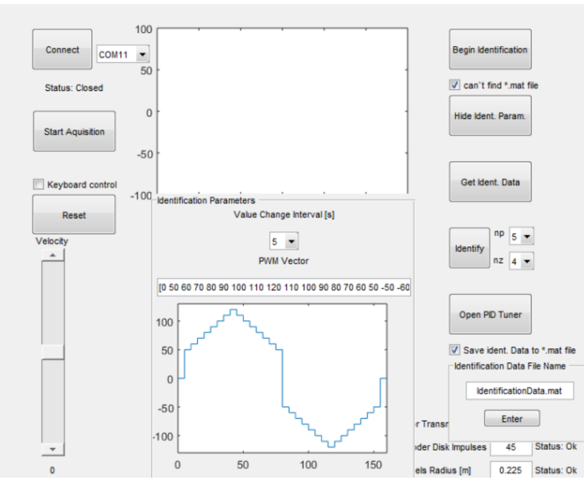

Fig.7 Identification GUI MATLAB

This GUI (presented on Fig. 7) allows user to control motor speed manually, send identification data, identify model, tune PID, set object constants and deploy model with new coefficients and constants to Arduino. After uploading a model to Arduino, user can check the quality of tuned controller by reconnecting to Arduino, setting reference velocity manually and checking the response. The quality of controlling depends on several factors. First of all, getting a proper data is one of the most important steps. Based on the collected data, picking the correct model is directly affects the PID tuning response and therefore directly affects the possibilities of tuning. At last, figuring the optimal response through MATLAB PID tuner directly affects the quality of controlling.

b) GUI for standalone application

Once identification and tests are complete, user can shut down MATLAB and use a standalone application as shown in Fig. 8 to control the DC motor. Standalone application requires less resources to run compared to multistep manual identification.

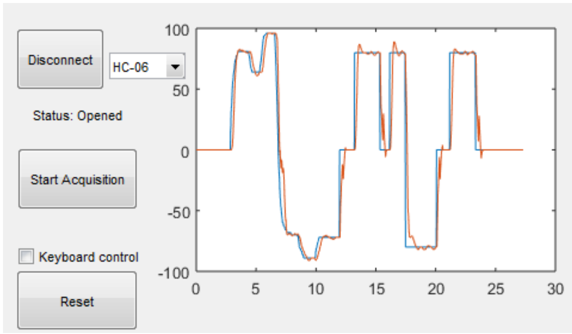

Fig.8 Speed controller performance at final stage with standalone application

In Fig. 8 graph depicts the reference tracking response of the DC motor after it has been tuned with the PID controller. The blue line shows the reference input, which is direction of motor changes with time. The red line shows the actual output of a motor, which closely tracks the reference input with minimal overshoot, and settling time, and there is no steady-state error. Finally, an analysis was performed to ensure that the controller was well-tuned with GUI of standalone application.

4. Conclusion

This project developed an interactive GUI to optimize PID controller parameters for a DC motor SIMULINK model in real-time. This approach enables continuous adaptation to varying operating conditions, achieving significant improvements compared to traditional offline or manual tuning methods. Experimental results demonstrate enhanced control accuracy and system stability, confirming the effectiveness of the proposed method. The PID controller, adjusted through the GUI, optimizes the motor's input voltage to minimize the error between the desired and actual output. This is achieved by utilizing a parameter estimation tool based on the DC motor's mathematical model. Careful tuning of the PID gains ensures a system that is both responsive and stable. The developed GUI serves as a valuable tool for robot designers and researchers, enabling real-time control optimization and hands-on learning of control systems concepts. It offers undergraduate students a practical learning experience in the classroom, fostering their understanding of PID control and its applications.

Acknowledgements

This work was supported in part by Mongolia-Japan higher Engineering Education Development (MJEED-JICA) through the joint research project "Development of industrial, service and intelligent systems based on advanced technologies" (J23A16).

References

1. Astrom, K. J. and T., Hagglund, PID Controllers: Theory, Design and Tuning, ISA, Research Triangle, Par, NC, (1995)
2. S. K. Suman and V. K. Giri, "Speed control of DC motor using optimization techniques based PID Controller," *2016 IEEE International Conference on Engineering and Technology (ICETECH)*, Coimbatore, India, 2016, pp. 581-587
3. Visioli, Antonio. (2012). Research Trends for PID Controllers. Acta Polytechnica. 52. 144-150.
4. Supriyo, B., Suharjono, A., & Atmaja, K. W. (2021). Modelling and Control of DC Motor Speed and Position for Wheel Mobile Robot Application. *Proceedings of the International Conference on Innovation in Science and Technology (ICIST 2020)*, 80–84.
5. Fu, Yuheng & Zhou, Qinyou. (2023). Analysis and application research of mobile robot navigation related technologies. Applied and Computational Engineering. 9. 92-96.
6. Khan, H.; Khatoon, S.; Gaur, P.; Abbas, M.; Saleel, C.A.; Khan, S.A. Speed Control of Wheeled Mobile Robot by Nature-Inspired Social Spider Algorithm-Based PID Controller. *Processes* **2023**, *11*, 1202.
7. K. Ogata (2009) "Modern Control Motorering, 5th Edition", Dorling Kindersley,".
8. Kiam Heong Ang, G. Chong and Yun Li, "PID control system analysis, design, and technology," in *IEEE Transactions on Control Systems Technology*, vol. 13, no. 4, pp. 559-576, July 2005
9. Vikhe, Pratap & Punjabi, Neelam & Kadu, C B. (2015). DC Motor Speed Control Using PID Controller In Lab View. IJISME. 3. 38-41.
10. Vikhe, Pratap & Punjabi, Neelam & Kadu, C B. (2014). Real Time DC Motor Speed Control using PID Controller in LabVIEW. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering. 03. 12162-12167.
11. Messaadi, Nassim & Amroun, Abdelkader. (2021). Speed Control of DC Motor Using Fuzzy PID Controller.
12. Nilawar, Vaishnavi & Sonare, Aniket & Samrutwar, Shrirup. (2022). Implement PID using MATLAB for motion control application.
13. Abdul-Kareem Mohammed, J. (2011). Modeling, Analysis and Speed Control Design Methods of a DC Motor. *Engineering and Technology Journal*.

Authors Introduction

Mr. Munkh-Erdene Ayurzana

He is currently studying for his Bachelor's degree in Mechanical Engineering at the Mongolian University of Science and Technology (MUST). His research interests include control system and robotics.

Mr. Erkhembayar Gankhuyag

He is currently studying for his Bachelor's degree in Mechatronics Engineering at the Mongolian University of Science and Technology (MUST). His research interests include control system and mechatronics.

Dr. Dondogjamts Batbaatar

He received his M.S. in the field of mechatronics from Mongolian University of Science and Technology (MUST), Mongolia and Ph.D. degree from Kyushu Institute of Technology, Japan in 2015 and 2021. He is currently a vice dean of research affair in School of Mechanical Engineering Transportation in MUST. His research interests include computational non-linear dynamics and bio-inspired robotics.

Dr. Erdenesuren Naranbaatar

He received his M.S. in the field of mechatronics from Mongolian University of Science and Technology (MUST), Mongolia and Ph.D. degree from University of Ulsan, South Korea in 2002 and 2013. he is currently an Associate Professor of the Graduate department in School of Mechanical Engineering Transportation in MUST. His research interests include computer vision and robotics.