Comparison of Machine Learning Algorithms for DC Motor PID Control with Genetic Algorithm

Ethan Deng[1], Leya Joykutty[#] and Juliana Carvalho De Arruda Caulkins[#]

[1]American Heritage School
[#]Advisor

ABSTRACT

Proportional-Integral-Derivative (PID) is a closed loop control system based on dynamic system feedback commonly employed in industrial control systems that require continuous modulated control. The purpose of this project was to improve the efficacy of PID control, a branch of classical control theory, to allow for the application of PID controllers in more complex and non-linear environments. The research compares the results of different machine learning algorithms with genetic algorithms to optimize PID control parameters to achieve more precise control of 12V DC motors. Genetic Algorithm (GA) is a metaheuristic algorithm used to solve search and optimization problems in machine learning. GAs are developed based on the process of natural selection and genetics, relying on biologically inspired operators. After a series of selections and crossovers, the genetic algorithm selected the fittest generation of PID constants that would result in the optimal PID controller which minimizes system error value and linearizes the system behavior. Optimizing PID parameters to enhance the efficacy of classical control theory could increase machining precision in industrial production and improve the robustness of integrated PID controllers in complex environments such as the human body, enabling PID controllers to be an adaptive, simple, and viable option to be implemented in implantable artificial organs and motorized prosthetics. The results of the study suggest that GA optimized PID parameters are effective in regulating the system behavior of DC motors and the enhanced PID control system could be implemented in a wide range of rapidly changing and non-linear applications.

Introduction

Problem Statement

Proportional-Integral-Derivative (PID) controllers are used in industrial processes to regulate temperature, pressure, speed, and many other production variables. Although PID control systems are proven to be robust in linear and symmetric systems, they are not suitable for applications with rapidly changing environment that introduces non-linearities to the system. However, adaptive control systems with modern control theory using state-space representation require intricate implementation and non-linear environments such as the human body are very complicated to be represented by a mathematical model. Enhancing the efficacy of classical control theory such as PID control would improve its robustness and effectiveness as an adaptive control system, allowing PID control to be a viable control system in rapidly changing environments and non-linear applications.

Overview of PID Control

Proportional-Integral-Derivative (PID) is a closed loop control system based on dynamic system feedback that provides continuously modulated control, and it is commonly employed in industrial control systems, mobile robotics systems, and automation. A PID control system continuously calculates the error value $e(t)$ of the system, which is the difference between the desired setpoint (SP) and a measured process variable (PV) and applies a correction based on the proportional, integral, and derivative term. The SP is essentially the desired output for the system, and the PV is the measured output of a system from a sensor or encoder. The purpose of a PID controller is to regulate the system input to minimize system error value to achieve the desired system behavior. The error value of a system can be obtained by subtracting the measured process variable from the desired setpoint:

$$e(t) = SP - PV$$

After obtaining the error value of the system, the PID controller applies a correction to the input of the system using the PID variable to try to achieve the desired output to gradually move towards the desired system behavior. The PID control variable is calculated using the differential equation with relation to error value and the P, I, and D constants with the following equation:

$$u(t) = K_p e(t) + K_i \int_0^t e(t)dt + K_p \frac{de}{dt}$$

In daily scenarios, PID control is implemented in the cruise control system of modern automobiles, which utilizes PID controllers to regulate the vehicle speed at a linear rate of change; PID applications in robotics systems improves the accuracy of motion planning with odometry and allows for more precise control of actuators.

Principles of PID Control Theory

PID control is a linear combination of the proportion (P), integral (I), and differential (D) of deviations in a closed loop feedback system. Ther Proportional (P) term increases the open loop gains of the system and change their signal amplitudes without impacting their phases in controlling changes to input signal $e(t)$. The Differential (D) term determines the differential for input signals, which reflects the rate of changes to a system, leads to general mode of predictive regulation, forecasts system variations, increases system damping, and enhances phase margin, thus improving system performances. The Integral (I) term, in addition to the P and D terms, records history of system changes, and the integral control applies correction to the current systems based on the history of system behavior (Yi Zhou, 2022).

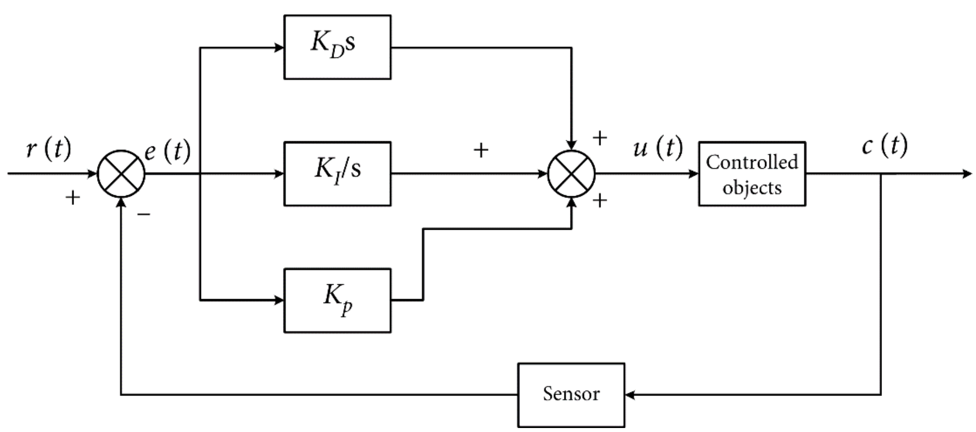

Figure 1. Schematic diagram of PID control (Yi Zhou, 2022).

Machine Learning

Machine Learning (ML) is defined as the ability of a program to learn and adapt from data to improve its performance on a certain task without explicitly being programmed to do so. Machine Learning algorithms build a model based on the sample data to gradually improve its performance on a certain task over time. ML has emerged across various industries in the business world as it enables enterprises to predict selected customer behavior and operational patterns. Likewise, machine learning assisted PID tuning algorithms would greatly improve accuracy of motion planning in robotics systems, allowing machines to perform finer tasks that require precise motion control.

Genetic Algorithm

Genetic Algorithm (GA), a subset of evolutionary algorithms, is a metaheuristic algorithm used to solve search and optimization problems in machine learning. Genetic algorithms are based on the process of natural selection and genetics, relying on biologically inspired operators such as mutation, crossover, and selection. Genetic algorithms are direct search algorithms into a region of higher performance in the solution space. In addition, unlike traditional artificial intelligence or machine learning algorithms, genetic algorithms have been proven to be robust when noise is introduced in the system. Genetic algorithms are developed based on the following principles:

1. Individuals in population compete for resources and mate.
2. The fittest individual mate to produce more offspring.
3. Genes from the fittest parents propagate throughout the generation.
4. Each successive generation should be more suited to their environment.

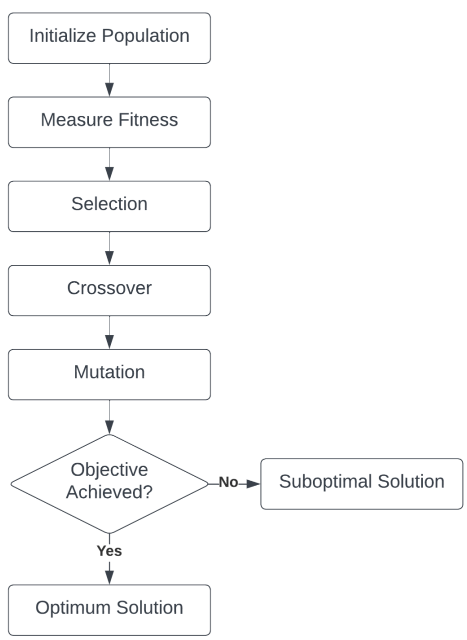

Figure 2. Schematic diagram of genetic algorithm.

Design Statement

The objective of the project is to create a PID tuning algorithm in combination with genetic algorithm and machine learning that selects the fittest generation of PID parameters that would result in the optimal PID controller. The construction of a prototype is also required to collect data and test the efficacy of the PID control parameters, which would involve the use of 12V DC motors that represent the regulated system. The algorithm would compare machine learning algorithms in PID control efficiency by examining their respective error value, and oscillations in system error value and non-linear system behaviors are considered as indicators of a poor performing PID controller.

Materials and Procedures

Materials List

-12V 463RPM DC Motor with Encoder, Gear Ratio 9.28:1
-Raspberry Pi 3 Model B Quad Core 64 Bit Wi-Fi Bluetooth
-L298N Dual H-Bridge Motor Driver
-Solderless Breadboard
-22 AWG Jumper Wires with Dupont Connectors

Prototype Construction

The prototype is constructed using a Raspberry Pi 3 Model B and a 12V DC motor with a magnetic quadrature encoder that records the position of the motor shaft. The collection of data and model testing was performed on a Raspberry Pi 3 Model B microcontroller, which communicates with the L298N Dual Channel H-Bridge motor driver that allows both the speed and direction control of the motor. The quadrature encoders on the motor are connected to the Raspberry Pi's GPIO pins through a solderless breadboard to record the incremental rotation of the motor shaft. The construction of the prototype began by designing a stabilizer or stand for the motor to dampen the motion of the motor to ensure minimum outside influences during the data collection process. The motor stabilizer was designed using Computer-Aided Design (CAD) software, which is then exported to an STL file and 3D printed with Polylactic Acid (PLA) material (Figure 3).

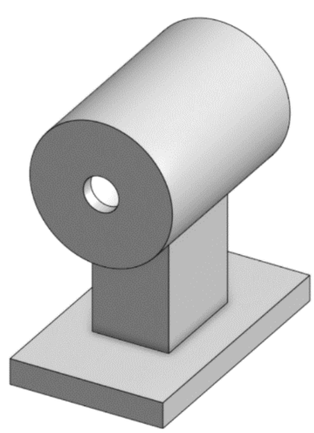

Figure 3. CAD design of the motor stabilizer.

Furthermore, the wiring of the prototype was determined and organized based on the schematic diagram of the L298N H-Bridge motor controller, which allowed the motor to be supplied with an external 12V DC power source (Figure 4). The implementation of the motor controller also enabled the use of Pulse-Width Modulation (PWM) control on the DC motors by varying the Duty Cycle output on each channel, and PWM control was utilized during the early phases of prototype construction to test the basic functionality of the DC motors.

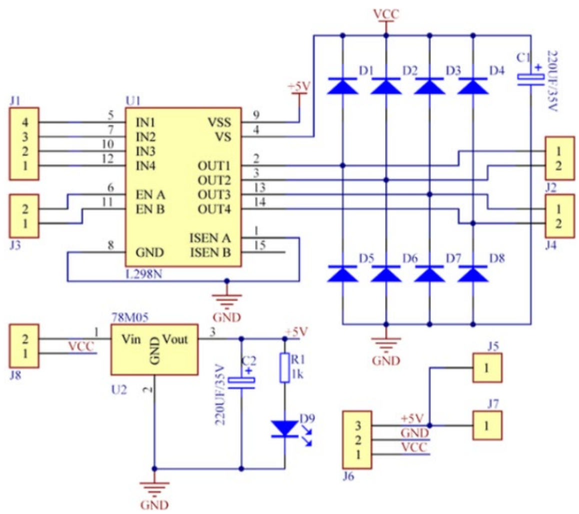

Figure 4. L298N Dual Channel H-Bridge Motor Controller Schematic Diagram (Handson Technology.).

After gathering all the physical materials and wiring the electrical components, the final prototype has been constructed for DC motor PID tuning (Figure 5). The Raspberry Pi Microcontroller essentially serves as a PID controller that regulates the system behavior of the DC motor, and the desired setpoint (SP) of the experiment is the desired angular velocity of the motor, and the measured process variable (PV) would be the actual measured angular velocity of the motor. The project has also specifically chosen a 12V DC motor because they are commonly used in industrial systems and robotics system, which ensures that the results of the experiment can be generally applied to other systems, not just the controlled system demonstrated in the research.

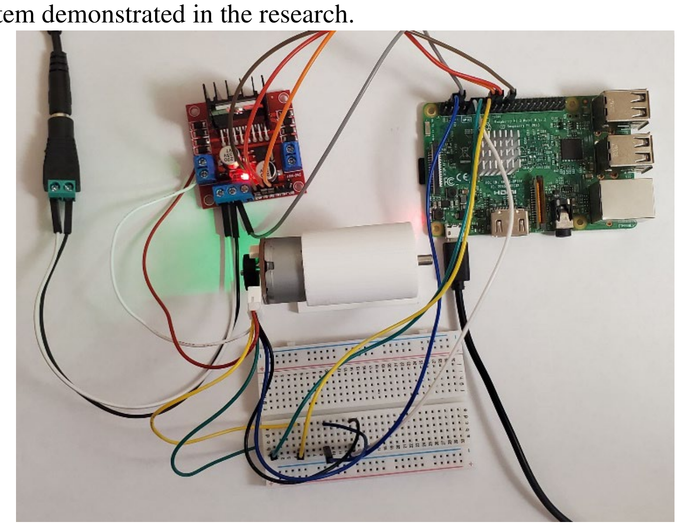

Figure 5. Picture of the final prototype used for DC motor PID tuning.

Data Collection

A Python PID control script was implemented to utilize the Raspberry Pi as a PID controller. The program also implements an automatic data collection condition loop where the motor will run to the desired speed and the error value and the fluctuation of the PID controller variable will be recorded and formulated into a CSV file. The data collection script allows a large amount of data to be recorded and processed to test with different machine learning algorithms to reduce the error value of the system. The PID controller also allows the test trial to specify the total duration of execution and the data sampling period. Afterwards, the collected test trials will be organized into a Python dictionary dataset and transformed into a Python Pandas DataFrame object to allow for further preprocessing. At preliminary stages, the experiment selected three common machine learning models such as Linear Regression, Random Forest Regression, and Recurrent Neural Network.

Genetic Algorithm and Training

The collected data of the PID controlled DC motor is transformed into a Pandas DataFrame object to allow for the ease of processing and machine learning. The data was trained using genetic algorithm based on PyGAD, the Python Genetic Algorithm library, in combination with different machine learning models. In addition, the correlation between variables was examined using the feature correlation heatmap to determine which variables in the dataset are valuable in the machine learning process and to calculate the fitness of a dataset (Figure 6). The correlation heatmap also shows non-correlated data that is not useful for the purpose of this research, thus they can be excluded during the training process to reduce inaccuracy.

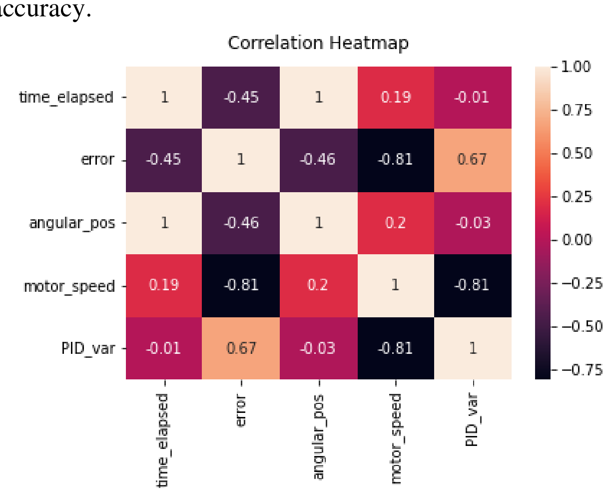

Figure 6. Feature correlation heatmap of the dataset generated using Matplotlib.

Data and Analysis

The selected datasets were data generated with the final PID parameters selected by GA, and the data collection script was executed with a total duration of 10.0 seconds, a sampling rate of 0.1 second, resulting in a total of 100 trials. The generated parameters of the PID controller, the proportional gain, integral gain, and the derivative gain, are set to 0.5, 0.35, and 0.01, respectively. The Python script also specified the target speed as 10.0 rad/s, which is the desired angular velocity of the motor shaft, and the measured angular velocity is recorded in the dataset. The first 10 trials of each dataset are displayed for closer examination for the purpose of data analysis.

Final Generation Dataset #1

K_p: 0.5 K_i: 0.35 K_d: 0.01
Total Period: 10.00s
Sampling Period: 0.1s
Target Speed: 10.0 rad/s

Table 1. The first 10 rows of data in dataset #1 were generated using the final PID parameters.

Time Elapsed (s)	Error Value e(t)	Angular Position (radians)	Angular Velocity (radians/s)	PID Variable u(t)
0.1	10	0	0	1
0.2	5.04	57.45	9.92	1
0.3	0.56	140.03	13.92	0
0.41	3.25	163.96	4.06	1
0.51	1.16	227.39	10.93	0.41
0.61	1.58	275.27	8	0.87
0.72	0.16	342.29	11.26	0.01
0.82	3.36	362.63	3.43	1
0.92	0.95	429.65	11.47	0.46
1.03	1.05	483.51	8.86	0.73

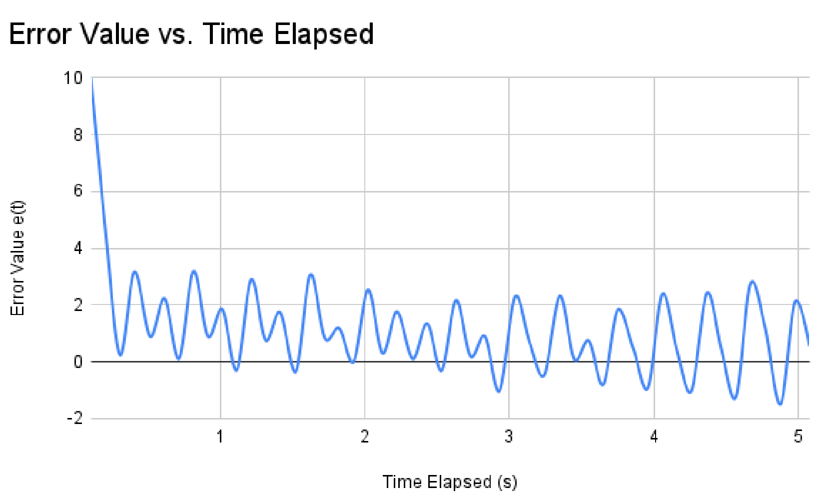

Figure 7. The graph of the system error value with the final generation PID parameters recorded in dataset #1.

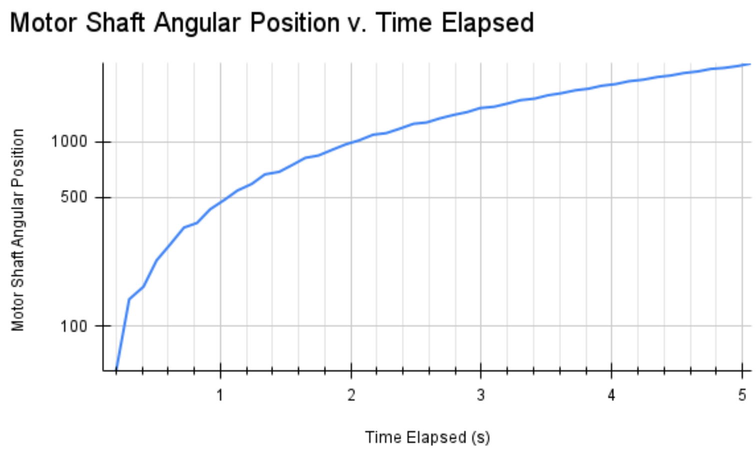

Figure 8. The graph models the angular position of the motor shaft on a log scale to examine the change in position of the motor shaft in dataset #1.

Final Generation Dataset #2

K_p: 0.5 K_i: 0.35 K_d: 0.01

Total Period: 5.00s

Sampling Period: 0.1s

Target Speed: 10.0 rad/s

Table 2. The first 10 rows of data in dataset #2 generated using the final PID parameters.

Time Elapsed (s)	Error Value e(t)	Angular Position (radians)	Angular Velocity (rad/s)	PID Variable u(t)
0.1	10	0	0	1
0.2	4.62	62.23	10.77	1
0.3	0.24	144.81	14.14	0
0.4	3.17	167.55	3.9	1
0.51	0.88	233.38	11.4	0.25
0.61	2.23	270.48	6.43	1
0.71	0.11	339.89	12.02	0
0.81	3.19	361.44	3.73	1
0.91	0.89	427.26	11.4	0.34
1.01	1.85	469.15	7.19	1

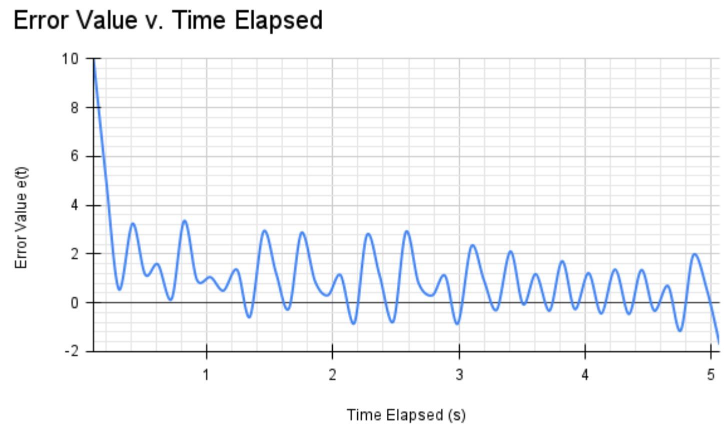

Figure 9. The graph models the system error value over time based on the target speed in dataset #2.

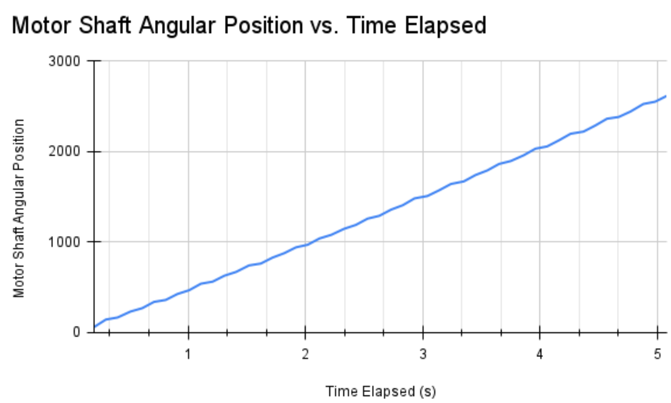

Figure 10. The angular position of the motor shaft recorded in dataset #2.

After retrieving the initial testing data from the prototype, a correlation heatmap was generated to allow for closer examination of the relationship between these measured variables (Figure 6). The PID variable exhibits strong correlation with both the error value and the motor velocity, which demonstrates the effectiveness of the PID controller because a strong correlation indicates the ability of the PID controller to provide modulated control of the motor behavior. Examining the first dataset generated with the final generation of PID parameters, as time elapsed during testing, the angular position of the motor shaft steadily increased with minimum oscillation, and the data is graphed on a log scale to visually amplify non-linear system behavior (Figure 8). Although the PID controller could not provide perfectly consistent modulated control, the error value of the system constantly fluctuates around the value of zero, and the relatively small scale of fluctuation exhibits proficient regulation of the motor behavior, which indicates that the combination of PID gains resulted in an efficacious PID controller variable (Figure 7). In the second dataset of the final generation, the error value over time exhibits similar behavior as the first dataset, where the error value constantly oscillates around the value of zero, indicating the PID controller in effect to regulate the motor velocity (Figure 9). The angular position of the motor shaft in the second dataset is graphed on a regular scale, and the linear correlation of the angular position and time elapsed demonstrates the ability of the PID controller to regulate system behavior towards a linear state (Figure 10). In addition, an example of a poorly tuned PID control system exhibits sudden shifts in the rate of change of the error value of the system, indicated by the sharp spikes on the graph, which demonstrates a discontinuous regulation of the DC motor with the PID controller variable. The example graph demonstrates a suboptimal combination of PID constants that results in sudden changes of the PID controller variable and poorly regulated error value that would lead to non-linear system behavior (Figure 11).

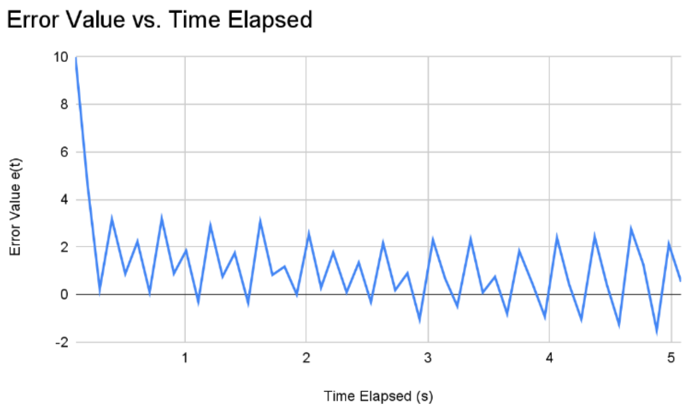

Figure 11. The error value of an example of a poorly tuned PID control system.

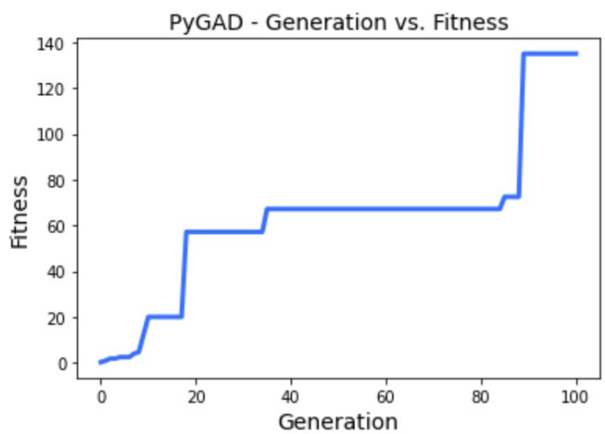

Figure 12. Graph of the progression of the fitness of each generation using PyGAD.

Overall, the dataset generated using the final generation of PID parameters selected by GA demonstrates the ability of the PID controller to effectively regulate system behavior of the DC motor, where the angular position of the motor shaft exhibits the linear behavior of the system. The sum of the error value over time also indicates the efficacy of the PID controller as the error value constantly oscillates around the x-axis with regulated fluctuations in the error value, which shows that the PID controller is trying to regulate the DC motor velocity towards the desired system behavior. Lastly, the progression of the calculated fitness of each generation of PID parameters also shows that the GA has chosen more successful generations of PID parameters over time, which will result in a more effective PID controller that regulate the system behavior towards a linear state (Figure 12).

Discussion

Conclusions

The results of the experiment demonstrate that the final PID control parameters selected by the genetic algorithm are efficacious in their ability to regulate system behavior and minimize fluctuations in the system error value. The results are expected as the PID control theory is considered a robust industrial closed-loop control system, and the application of PID control in automation and other industrial processes proves its reliability and effectiveness. Although the PID control parameters did not demonstrate perfect consistency in regulating the motor velocity and angular position in relation to the target speed, the error values in the two datasets only exhibit minimal oscillations around the zero value and the angular position of the motor shaft also illustrates the overall linearized system behavior. In conclusion, the results suggest that genetic algorithms are generally effective in PID controller tuning.

Applications

The results of this project demonstrate that GA enhanced PID control systems are more effective in linear applications, and the selected PID parameters generally improve the efficacy of the PID controller. The improved PID control system also shows the potential of achieving the effects of an adaptive modern control system such as state-space control using classical control theory, which simply focuses on capturing and regulating the inputs and outputs of the system. In addition, the use of genetic algorithms with machine learning algorithms enables PID controllers to be viable options for control systems in non-linear environments like the human body, with the implementation of integrated PID controllers in artificial organs and motorized prosthetics. The enhanced PID controllers would also improve its performance in the traditional applications, improving machining precision and regulation of production variables in industrial processes.

Limitations

There are natural limitations to the project itself since there are inevitable noises in the circuitry that might have interfered with the quality of the data, and the DC motors utilized in the experiment might have mal-performing rotors, thus these limitations are hard to eliminate from the experimental environment. In addition, due to the limited time and scope of this project, a limited variety of motors are tested and the improved PID controllers were not stress tested under load, which would generally demonstrate the true performance of a PID control system. Therefore, the combination of different limitations would result in the limited applications of the conclusions drawn from the results of the experiment.

Future Research

In the future, further testing can be conducted with other combinations of PID parameters and additional machine learning algorithms, and it is also ideal to stress test the PID control system under load, which would further improve the efficacy of GA enabled PID control systems. Additionally, the implementation of integrated PID controllers in artificial organs and Brain-Computer Interface (BCI) enabled motorized prosthesis can also be used to conduct further testing with GA selected PID control parameters to examine its effectiveness in the real world.

Acknowledgements

I would like to thank my research mentors, Leya M. Joykutty, and Juliana Caulkins, for assisting and guiding me throughout the course of this project.

References

Grelewicz, P., Khuat, T.T., Czeczot, J., Klopot, T., & Gabrys, B. (2021). Application of Machine Learning to Performance Assessment for a class of PID-based Control Systems. *ArXiv, abs/2101.02939.*

Handson Technology. (n.d.). *L298N Dual H-Bridge Motor Driver* [PDF]. http://www.handsontec.com/dataspecs/L298N%20Motor%20Driver.pdf

Hernández-Alvarado, R., García-Valdovinos, L. G., Salgado-Jiménez, T., Gómez-Espinosa, A., & Fonseca-Navarro, F. (2016). Neural Network-Based Self-Tuning PID Control for Underwater Vehicles. *Sensors (Basel, Switzerland)*, *16*(9), 1429. https://doi.org/10.3390/s16091429

I W Widhiada *et al*. (2019). *IOP Conf. Ser.: Mater. Sci. Eng*. 673 012109 https://doi.org/10.1088/1757-899X/673/1/012109

Jali, Mohd & Azam, N & Ghazali, Rozaimi & Jaafar, Hazriq Izzuan & Bohari, Z. (2016). PID control tuning using PSO for prosthetic hand system. International Journal of Applied Engineering Research. 11. 10274-10280.

Jayachitra, A., & Vinodha, R. (2014). Genetic algorithm based PID controller tuning approach for continuous stirred tank reactor. *Advances in Artificial Intelligence*, *2014*, 1-8. https://doi.org/10.1155/2014/791230

Mirzal, A., Yoshii, S., & Furukawa, M. (2012). PID Parameters Optimization by Using Genetic Algorithm. *ArXiv, abs/1204.0885*.

Nyberg, A. (2017). Optimizing PID parameters with machine learning. *arXiv preprint arXiv:1709.09227.*

Tanvir Ahmmed, Irin Akhter, S. M. Rezaul Karim, F. A. Sabbir Ahamed, Genetic Algorithm Based PID Parameter Optimization, *American Journal of Intelligent Systems*, Vol. 10 No. 1, 2020, pp. 8-13. doi: 10.5923/j.ajis.20201001.02.

Yi Zhou, "A Summary of PID Control Algorithms Based on AI-Enabled Embedded Systems", *Security and Communication Networks*, vol. 2022, Article ID 7156713, 7 pages, 2022. https://doi.org/10.1155/2022/7156713