IJRCS

International Journal of Robotics and Control Systems
Vol. 3, No. 3, 2023, pp. 485-500
ISSN 2775-2658
http://pubs2.ascee.org/index.php/ijrcs

ASCEE

Integrated Modelling and Control of Linear Actuator Based Automatic Pedal Pressing Mechanism for Low-Speed Driving in a Road Traffic Delay

Azrul Azim Abdullah Hashim [a,1], Salmiah Ahmad [b,2,*], Nor Maniha Abdul Ghani [a,3], Ahmad Nor Kasruddin Nasir [a,4]

[a] Faculty of Electrical & Electronics Engineering, Universiti Malaysia Pahang
[b] Department of Mechanical and Aerospace Engineering, Kulliyyah of Engineering, International Islamic University Malaysia (IIUM)
[1] azimbaazim@gmail.com; [2] salmiah@iium.edu.my; [3] normaniha@ump.edu.my; [4] kasruddin@ump.edu.my
* Corresponding Author

ARTICLE INFO

Article history
Received June 06, 2023
Revised July 12, 2023
Accepted July 18, 2023

Keywords
Automatic Braking;
PID Controller;
Linear Actuator;
Pedal Pressing;
Road Traffic Delay;
Low Speed Control

ABSTRACT

Sitting in traffic congestion for hours in a posture that requires recurrent actions of manually pressing the pedal and braking excessively can result in fatigue, especially on the driver's leg and back. This fatigue can have long-term implications and adversely affect the driver's health. Thus, this paper aims to model and develop a control system that utilizes a linear actuator to replace the leg activities involved in pressing and releasing the brake pedal. This approach, combined with the implementation of a PID controller, offers a novel solution to control the vehicle speed by integration with the linear actuator that focus on low-speed driving condition. The design process begins with creating a 3D model using SolidWorks to visualize the movement of the linear actuator and Pedal subsystem. This model is then connected to Matlab-Simulink, where a PID controller is implemented and integrated into the electrical circuit to control the actuator's movement. Integration with the vehicle dynamic model enables a comprehensive analysis of the system's behavior on the vehicle dynamics. This research compares the trial and error method with the Matlab tuner for implementing the PID controller. The performance of the system will be evaluated based on the steady state error, overshoot, rise time, and settling time. The results demonstrate that the Matlab tuner outperforms trial and error method by achieving a faster response and significantly reducing steady state error during robustness testing. With the integration of the linear actuator, the system is capable of tracking the desired speed and has the potential to replace the leg activities involved in pressing and releasing the brake pedal. For future work, validating the proposed mechanism with a physical prototype of the linear actuator and pedal using hardware-in-the-loop techniques poses a challenge, as hardware constraints may vary with different environments.

1. Introduction

Traffic congestion poses a significant challenge as extended periods of sitting in traffic result in frequent and excessive manual pedal pressing and braking, negatively impacting drivers' physical health. Research indicates that monotonous driving and repetitive actions can lead to muscle stiffness and fatigue [1]-[3]. In a study on Malaysian commercial vehicle drivers, a high prevalence of low back pain (60.4%) was found, with fatigue identified as a significant risk factor [4]. The stress of being stuck in traffic can disrupt emotions and productivity, creating a negative work atmosphere [5]. Prolonged sitting in a car also contributes to fatigue and back pain [6]. Therefore, effective solutions are essential to address these challenges associated with road traffic delays.

The objective of this paper is to develop a control system that automates the pedal pressing process using a linear actuator. By incorporating a PID controller, the system can regulate the force exerted by the actuator to its position, enabling control over the pedal motion and ultimately the speed of the vehicle. This approach eliminates the need for manual leg activities such as pressing and releasing the pedal, which the linear actuator takes over to operate the pedal brake, reducing driver fatigue by allowing the driver to rest their leg activities. The electric linear actuator provides compact size, high efficiency, and easy installation, making it popular in automotive, industrial, medical, and home automation applications. Its ability to convert electrical energy into linear motion makes it ideal for automating pedal pressing, enabling precise control over vehicle speed in the proposed system.

In the early studies, researchers primarily focused on controlling the position of the actuator. Specifically, researchers, including [7], have explored the application of classical control laws, such as I-PD, in the control of pneumatic linear peristaltic actuators. The main aim of these investigations was to minimize positioning errors and evaluate the performance of these actuators in comparison to conventional piston actuators. Additionally, in another study conducted by [8], the authors propose a model-based robust force control approach specifically designed for pneumatic actuation systems. They apply a sliding mode control technique to a double-acting cylinder in order to achieve effective force control. Actuators find significant applications, such as in humanoid robots as demonstrated in [9], where a direct-drive linear motor is utilized. The authors evaluate the performance of a PID controller for precise positioning, acceleration, and impact force on the foot. Another research in [10] the author proposed a cost-effective sensor less position control method for a solenoid actuator using current difference measurement in a PWM supply. The study's findings revealed an impressive level of accuracy in the online position estimation, with minimal relative error and deviation observed between the measured and estimated positions. These results show the precision achieved in determining the actuator's position. Additionally, in [11], focuses on the analysis of a hybrid linear actuator, specifically a pneumatic-electric actuator. The researchers address the challenges of position tracking and input allocation through the application of convex optimization with PD predictive control, where the tuning process was performed manually.

Recent studies have demonstrated an increasing emphasis on developing advanced positioning control techniques. Researchers are exploring various control algorithms, such as Fuzzy logic controller [12]–[17],Sliding mode control [18]–[20] and Neural Network. In [21]–[23], to achieve precise positioning control. However, implementing these advanced control techniques can be challenging, as it requires a deep understanding of the theoretical foundations and may have certain disadvantages. Consequently, the PID (Proportional-Integral-Derivative) controller remains one of the most widely applied techniques in industrial settings due to its simplicity, robustness, and effectiveness in achieving accurate positioning control. Studies have shown that PID controllers offer effective design and deployment for motor drives and precise motion control in piezoelectric actuators [24].

The common challenge in implementing PID controllers is tuning the parameters to achieve optimal system performance. The traditional approach for tuning PID parameters is through trial and error, which involves iteratively adjusting the parameters and observing the system's response until the desired performance is achieved [25]. This process can be time-consuming and often requires multiple tuning iterations [26]. Consequently, researchers have explored alternative tuning methods to improve the efficiency and effectiveness of PID controller tuning such as Ziegler Nicholas [27], [28], Genetic Algorithms [29]–[31], Particle Swarm Optimization [32]–[34] n, Fuzzy Tuning [35], [36] and Matlab Tuner [37]. Matlab Tuner stands out as a straightforward and highly efficient tuning method among various alternatives. Its simplicity lies in the fact that it is an integrated feature of Matlab, eliminating the need for extra programming efforts. Additionally, it offers a comprehensive analysis of system responses, providing detailed estimations. The exceptional user interface of Matlab tuner further enhances its usability, making it an ideal choice for tuning purposes.

Thus, in this research the primary objective is to achieve precise control over the linear actuator to effectively control the speed of the vehicle By integrating the actuator with the vehicle's dynamics, the control system ensures that the actuator exerts the necessary force on the attached pedal at each specific position, thereby controlling the vehicle's speed. The main advantage of this approach is that it eliminates the need for drivers to manually press the pedals during traffic congestion, which can lead to fatigue. In the following section, the proposed concept and the chosen control strategy will be discussed in more detail. This research makes significant contributions by developing an automated pedal pressing system design, designing a practical PID controller for precise control the linear actuator, and providing a detail of the system's performance comparison of trial-and-error tuning method and Matlab tuner.

2. Proposed Design Process

2.1. Mechanical Modelling

The proposed design utilizes a linear actuator to automate the process of engaging and disengaging the pedal brake during low-speed driving in traffic delays. The actuator is attached to the back of the pedal and moves the pedal brake forward or backward depending on its direction as shown in Fig. 1. When the actuator retracts, the leadscrew moves forward, applying force to the pedal brake and causing it to engage the brakes. Conversely, when the actuator extends, the leadscrew moves backward, releasing the force on the pedal brake and causing it to disengage the brakes.

The prolonged usage of the actuator can lead to increased friction and temperature within its components, resulting in a decline in performance. This decrease in performance can lead to inaccuracies in the force exerted by the actuator, thereby affecting the control of the vehicle's speed. Consequently, the utilization of this linear actuator is not suitable for situations involving heavy traffic congestion, where the actuator would need to operate continuously for extended periods, such as 4 or 5 hours. In such conditions, the actuator may experience significant wear and decreased efficiency, making it unreliable for maintaining precise speed control. Therefore, alternative solutions or actuator systems that are specifically designed to withstand long operating times would be more appropriate for such scenarios.

Fig. 1. Proposed design pedal brake with linear actuator

The chosen linear actuator was selected based on various factors, including its ability to provide force feedback for precise control of the pedal brake. This ensures accurate engagement and disengagement, improving the braking system's reliability during low-speed driving. Additionally, the compatibility of linear electric actuators with control systems enables seamless integration into the research setup, allowing for precise and responsive control over the actuator's motion. Furthermore, the decision to use an existing actuator from the industrial market was influenced by its extensive testing and established safety record in diverse applications, ensuring reliability and enhancing overall system safety.

The primary objective was to synchronize the parameters of the actuator used in simulation with the real hardware prototype. For future progress, a validation process is intended to be conducted using hardware in the loop technique. This validation step aims to assess the system's behavior and performance under realistic conditions, facilitating the bridging of the gap between simulation and physical implementation. Moreover, selecting an actuator with appropriate stroke length is essential to minimizing space utilization and preventing any interference with other vehicle components or occupants.

In the following section, mathematical model that represents the working of the linear actuator is presented. The relationship between voltage signal, Rotational motion and Linear force and position will be extensively discussed.

2.2. Mathematical Model of Linear Actuator

The actuator converts electrical signal in form of voltage to linear force. Fig. 2 shows the circuit representation of linear DC motor.

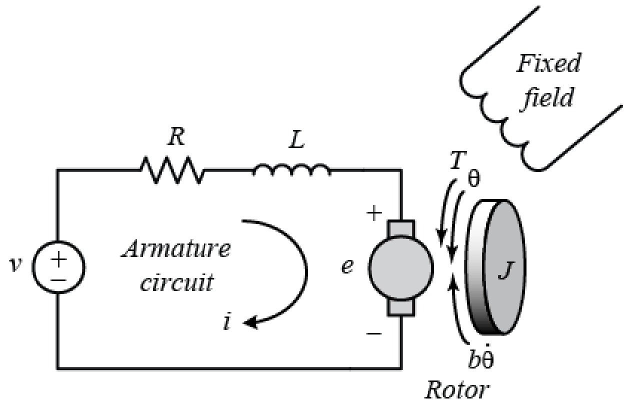

Fig. 2. Schematic of a DC motor system

In Fig. 2 the rotational torque T_{motor} is directly proportional with current i,

$$T_{motor} = K_{motor} i \tag{1}$$

where K_{motor} is the motor's constant, and it depends on physical dimension of the motor. By applying the Kirchhoff Voltage Law (KVL) and the Laplace transform on circuit in Fig. 2 yields the relationship between input voltage V and output rotational torque T where the relationship is demonstrated by (2).

$$\frac{T_{motor}(S)}{V(s)} = \frac{K_{motor}}{sL + R} \tag{2}$$

On the mechanical shaft, the same process applies in which KVL and the Laplace transformation is performed and yields the relationship between the shaft's angular speed W_{shaft} and input voltage V.

$$\frac{W_{shaft}(S)}{\mathrm{V}(S)} = \frac{{}^{1}/_{b}}{s(J/\mathrm{b}) + 1} \tag{3}$$

where b and J are shaft's rotational friction constant and moment of inertia, respectively.

2.3. Conversion Linear Force and Linear Position

The linear force and linear speed are required to apply a direct pressing on braking pedal. In actual DC motor, the gearbox and leadscrew are connected with rotational shaft to convert an angular torque to linear force. By assuming the leadscrew and gear friction are negligible, the relationship between input and output torque with gear ratio is

$$G = \frac{\tau_o}{\tau_{in}} \tag{4}$$

Where G is the gear ratio, τ_o is output angular torque and τ_{in} is the input angular torque from rotating shaft. Thus, the linear force, F is produced as a result from rotating leadscrew.

$$F = G\,\frac{2\pi}{l_{\Upsilon}} \tag{5}$$

Where l is the length of one pitch on the leadscrew. As for linear position x_{linear} of the leadscrew can be expressed by the following equation.

$$X_{linear} = \theta_{linear}\,\frac{l}{2\pi} \tag{6}$$

The linear actuator model is then simulated by employing the Matlab-Simulink platform, as shown in Fig. 4. The parameters used for the actuator taken from the industry-available actuator, is summarized in Table 1.

Table 1. Linear actuator parameters used in simulation

No	Parameter	Value
1	Torque Constant	0.4 N.m
2	Winding Resistance	1 Ω
3	Winding Inductance	0.05 H
4	Screw lead pitch	0.008 m
5	Gearbox ratio	2
6	Lead screw efficiency	40 %
7	Shaft moment of inertia	0.01 Kg.m^2
8	Rotational friction coefficient	0.1 N.m. s

2.4. Modelling of Proposed Idea in Solidworks

The SolidWorks CAD model of the pedal subsystem, including components such as the pedal brake, linear actuator nut, and pedal base, is utilized as the initial design representation as shown in. To further analyses the system's dynamics and behavior, the SolidWorks model is exported to MATLAB and imported into Sims cape Multibody, enabling the creation of a simulation environment.

In Sims cape Multibody, the components from the SolidWorks model are accurately replicated and interconnected using joints, which simulate the physical connections and constraints among the mechanical parts. This enables the precise representation of forces, motions, and interactions within the pedal subsystem.

By conducting simulations in this virtual environment, insights into the behavior and performance of the pedal subsystem can be obtained. Factors such as the applied force to the pedal and the characteristics of the actuator can be studied to understand their effects on the system's response. This information aids in optimizing the design, identifying potential issues, and making informed decisions prior to physical prototyping and testing.

3. Simulation System Design

3.1. Modelling a Linear Electric Actuator in Simulink

Fig. 3 illustrates the overall block diagram of the system design. The actuator was composed of various components, including a DC motor, gear box, and lead screw. These components were chosen based on their specifications outlined in Table 1. In this diagram, the PID controller is directly connected to the electric components of the linear actuator, such as the DC motor, lead screw, and gearbox, as shown in Fig. 4. The PID controller generates a control signal that is supplied to the DC motor. The DC motor, along with the lead screw and gearbox, converts this electrical energy into linear motion.

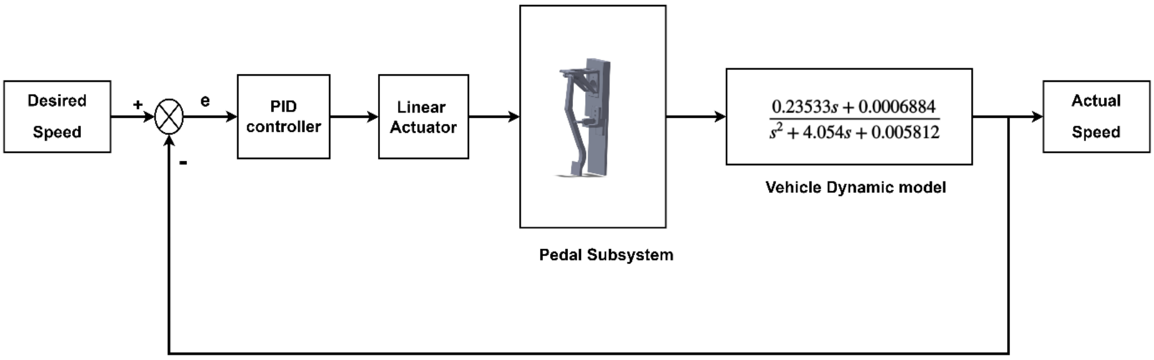

Fig. 3. Block diagram of the system

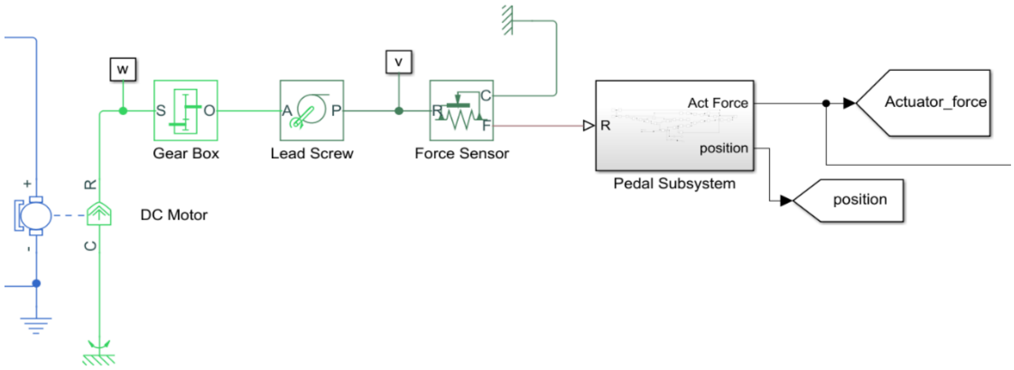

Fig. 4. Electrical actuator design in Matlab-Simulink

To determine the position of the actuator, a force sensor is incorporated into the system. This force sensor measures the force exerted by the actuator, providing feedback on its position. The force produced by the actuator is then directly integrated with the vehicle dynamics of the car to achieve the desired speed. The pedal subsystem's representation in the form of a detailed 3D model provides an accurate overview of the pedal's movement and position.

Table 2 presents information about the parameters necessary for configuring the component in a linear electric actuator. This step is crucial to achieve a good system response. Properly setting up the component ensures the actuator system's optimal performance and functionality.

3.2. PID Controller Design and Tuning Method

PID controllers are the most extensively used controllers in the motion control, process control, power electronics, hydraulics, pneumatics, and manufacturing industries etc. [38],due to their simple structure, easy implementation and maintenance. They are also prevalent in modern applications, like self-driving cars, Unmanned aircraft vehicle, and autonomous robots, for the same purpose [39] e. In most of the control system applications 90-95 % of control loops are of PID form.

Table 2. Linear actuator parameters used in simulation

Components	Parameter
DC motor	The block represents a mathematical model of a direct current (DC) motor. It captures the motor's behavior and characteristics, such as speed and torque, and allows for analysis and control design within the Simulink environment. Therefore, a specific parameter needs to be set to accurately represent the characteristics of the motor being modelled, such as torque, armature resistance and inductance
Gear box	The gear ratio in a linear actuator determines how much linear displacement is achieved for each rotation of the motor or gear mechanism. By selecting the appropriate gear ratio, the actuator can achieve the desired force and speed based on the motor's capabilities. Higher gear ratios provide higher force but lower speed, while lower gear ratios offer higher speed but lower force. The gear ratio choice depends on the specific requirements of the application, balancing force, speed, and precision
Lead screw	A lead screw is a mechanical component often used in actuators to convert rotary motion into linear motion. It consists of a threaded shaft (screw) and a nut that engages with the threads. When the screw rotates, the nut moves linearly along the shaft, resulting in the desired linear displacement The appropriate size of a lead screw for a specific application depends on factors such as the required linear travel distance, the load to be lifted or moved, and the desired resolution or precision.
Force sensor	The Ideal Force Sensor block represents a device that converts a variable passing through the sensor into a control signal proportional to the force. The sensor is ideal since it does not account for inertia, friction, delays, energy consumption, and so on.
Prismatic joint in Pedal subsystem	In the pedal subsystem, a prismatic joint is utilized to represent the movement behavior of the pedal. To accurately model this behavior, certain adjustments are necessary. At the input of the prismatic joint, the required force for the pedal needs to be specified. On the other hand, at the output of the prismatic joint, the force and position of the actuator need to be calculated based on the system requirements and desired pedal movement. These adjustments ensure that the prismatic joint accurately simulates the behavior of the pedal and enables precise control over its movement.

The PID controller block consists of three components: the proportional (P) term, the integral (I) term, and the derivative (D) term as show in Fig. 5. These terms are responsible for adjusting the control signal based on the error $e(t)$, between the setpoint and the actual output of the system. The output of the PID controller is then connected to the plant or process block. The plant represents the physical system or process being controlled. It can be any system that responds to the control signal, such as a motor or actuator. The output of the plant represents the actual output $y(t)$ of the system, which is then fed back to the input of the PID controller. This feedback creates a closed-loop system, where the PID controller continuously adjusts the control signal based on the difference between the setpoint $r(t)$ and the actual output $y(t)$. The purpose of the PID controller is to minimize the error $e(t)$ between the setpoint and the actual output by dynamically adjusting the control signal and generate a control signal $u(t)$ to drive the system that being controlled. The proportional term provides immediate correction based on the current error, the integral term corrects for steady-state errors over time, and the derivative term anticipates the future trend of the error.

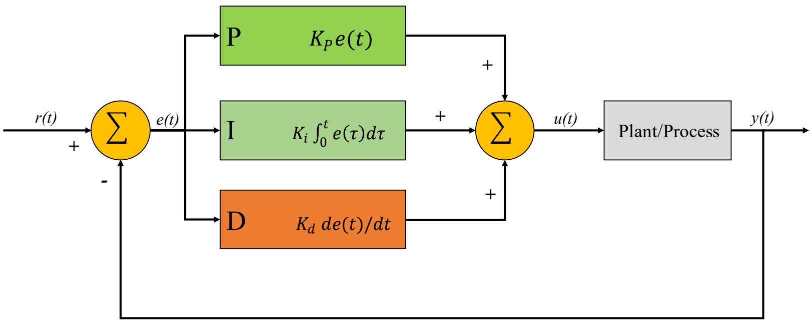

Fig. 5. Block diagram for PID controller

Thus, in this study, a PID controller is employed to regulate the linear actuator, which serves as a mechanism to replace leg activities in pedal pressing and releasing, as well as controlling the vehicle's speed. The PID controller continuously adjusts the actuator's force output based on the difference between the desired speed and the actual speed, as shown in Fig. 6. By modulating the actuator's force output, the PID controller enables precise control over the vehicle's speed.

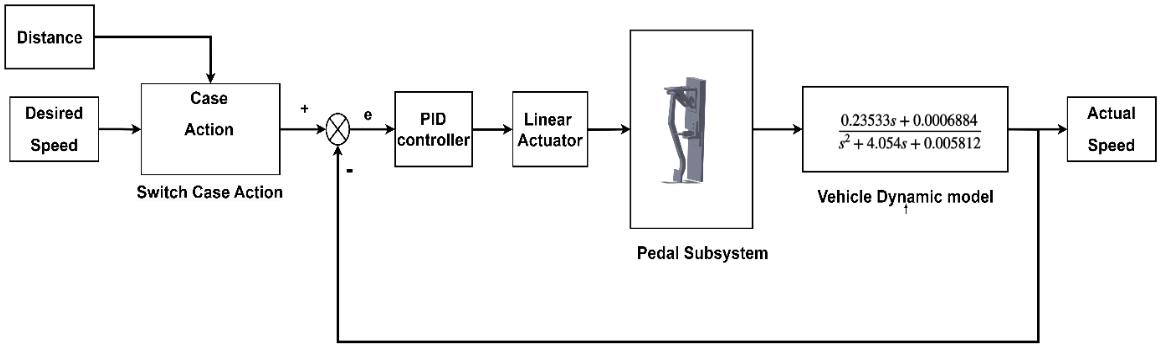

Fig. 6. Block diagram with distance

The process of adjusting the proportional (P), integral (I), and derivative (D) gains, to achieve the desired control performance is commonly referred to as the tuning process. Through tuning, the gains are optimized to balance the system's response characteristics, such as response speed, stability, and elimination of steady-state error. Finding the optimal values for the PID terms is crucial to achieve the desired control behavior and maximize the system's performance. The tuning process involves iterative adjustments of the gains, considering factors like overshoot, oscillations, settling time, and steady-state error, until the desired control performance is achieved. In this study, a comparison will be made between two tuning methods, manual tuning and the use of a MATLAB tuner, to evaluate their effectiveness in optimizing the PID controller. The subsequent chapter will provide a detailed discussion of these tuning methods.

4. Results and Discussion

The PID controller will test with linear actuator and dynamic model of vehicle by using simulink Matlab in simulation environment. A comparison is made between two tuning methods for the PID controller: Manual, and the Matlab PID Tuner tool. The primary aim of this analysis is to verify the ability of the system with the existing of linear actuator to accurately track the desired speed of the vehicle based on the proposed tuning method. This analysis focuses on evaluating the system's performance in terms of vehicle speed control with the integration of linear actuator, specifically in low-speed driving condition in a traffic congestion. By assessing the results obtained from the system performance analysis, it can be determined if the force exerted by the linear actuator effectively translates into the desired speed of the vehicle.

4.1. Manual Tuning

The trial-and-error method, also commonly known as manual tuning, is a popular approach for adjusting the parameters of a PID controller [40]. This method involves iteratively testing different combinations of proportional, integral, and derivative gains by manually adjusting the values and observing the system's response. The controller's performance is evaluated, and based on the observed behavior, further adjustments are made to refine the parameter values in subsequent trials. This iterative process continues until the desired system response or performance specifications are achieved. The trial-and-error method is widely used due to its practicality and intuitive nature, allowing for effective PID controller parameter optimization.

To evaluate the performance of the control system, data is collected from five different sets of PID parameters while keeping the setpoint constant. Table 3 provides a summary of the PID parameter settings used in the manual tuning process, with variations in proportional (P), integral (I), and derivative (D) gains for a setpoint of 4 km/h. Fig. 7 illustrates the system response analysis graph resulting from the manual tuning process, demonstrating the behavior of the system under different

PID parameter combinations. Fig. 8 displays the force exerted by the actuator throughout the system response analysis, providing a visual representation of how the force varies over time for different PID parameter combinations. Additionally, Table 4 presents the results of the system performance, including metrics such as steady-state error, overshoot, rise time, and settling time. These metrics serve as indicators of the system's performance and help evaluate the effectiveness of different PID parameter combinations.

Table 3. PID Parameter for Manual Tuning

No	Setpoint	PID Parameter		
		P	I	D
1	4 km/h	20	0	0.5
2	4 km/h	10	5	1
3	4 km/h	30	10	0.8
4	4 km/h	10	8	1.2
5	4 km/h	7	4	1.4

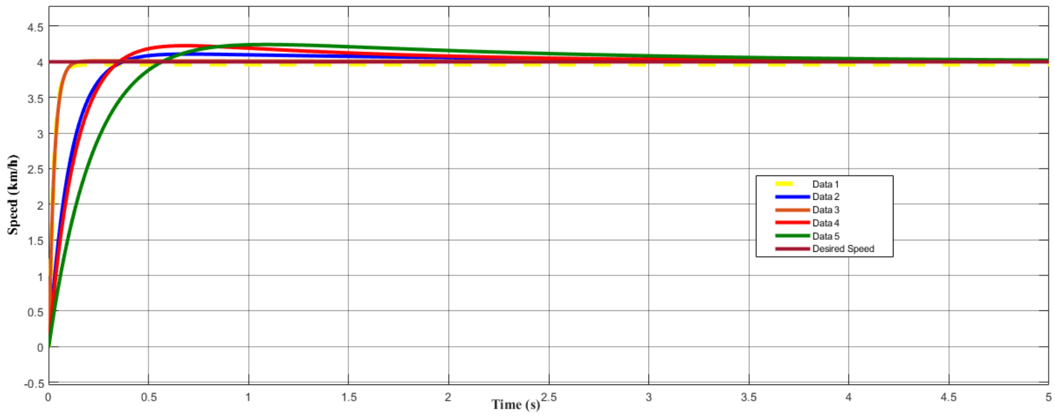

Fig. 7. System response by applying Manual Tuning

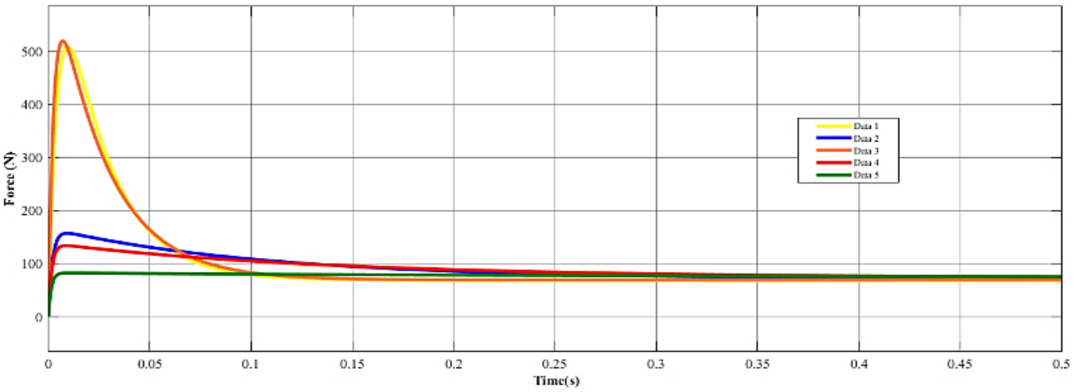

Fig. 8. Force response by applying Manual Tuning

Table 4. System Performance Analysis for Manual Tuning

No	Setpoint	System Performance Analysis				
		Steady State Error	Overshoot (%)	Rise Time(s)	Settling Time(s)	Force (N)
1	Data 1	0.0336	0	0.0540	0.1090	67.9
2	Data2	-0.0012	2.7102	0.2077	1.4061	68.1
3	Data3	-5.4e-4	0.3478	0.0567	0.1009	68.2
4	Data4	-2.8e-4	5.6385	0.2259	2.2259	68.1
5	Data5	-0.0012	6.0640	0.3698	3.0800	68
	Average	-6.076e-3	2.9521	0.1828	1.3843	68.06

Vol. 3, No. 3, 2023, pp. 485-500

Based on the system performance analysis, the average steady-state error, overshoot, rise time, and settling time of -6.076e-3, 2.9521, 0.1828, and 1.3843, respectively, along with the average force output of the linear actuator at 68.06 N, demonstrate that the trial and error tuning method has provide acceptable results for the PID controller. This indicates the controller's ability to effectively maintain the desired output speed of the vehicle.

4.2. Robustness Analysis

The evaluation of the system's robustness involved testing its performance in handling various distances and corresponding speeds during traffic delay situations. This was accomplished using a switch case block in Matlab-Simulink, where different speed values were assigned based on the distance range from 0 to 10 meters. From the observation based on Fig. 9 and Fig. 10, the system exhibited a slow response in tracking the desired speed and had a significant gap in achieving the desired speed at a steady state, indicating a larger steady-state error.

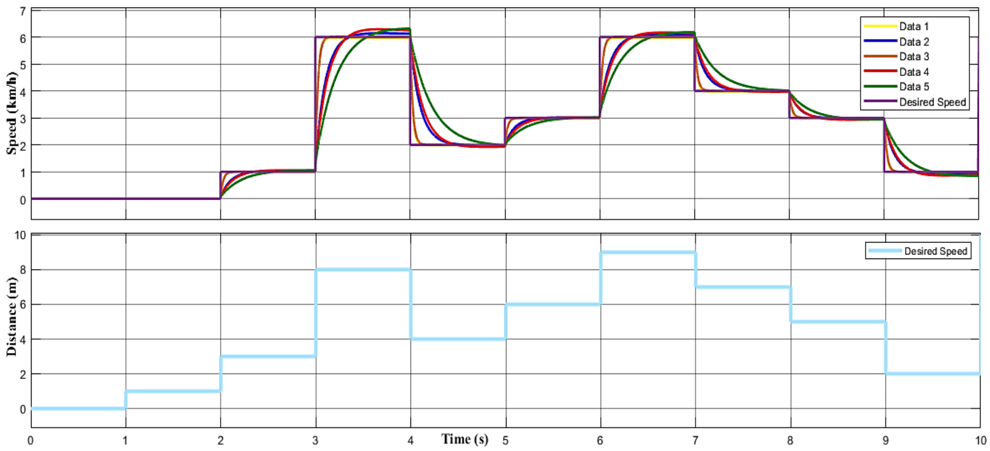

Fig. 9. System response with variation speed and distance

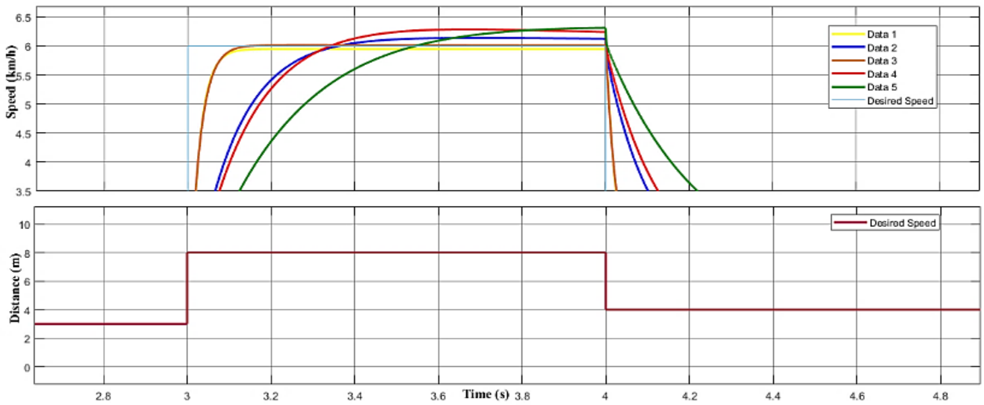

Fig. 10. Zoom view of the system response from 3s – 4s

4.3. Matlab Tuner

The Matlab PID Tuner is a powerful tool that facilitates the tuning of PID controllers in control systems. It provides an interactive and graphical interface for tuning PID parameters based on system response data. The PID Tuner allows users to analyze the system's response and adjust the proportional, integral, and derivative gains in real-time. By adjusting the parameters and observing the resulting system behavior, users can iteratively refine the PID controller's performance to meet desired specifications. The tool provides a streamlined and efficient workflow for PID controller tuning, enabling users to achieve optimal control system performance.

In a similar manner to the previous test, the experiment was conducted five times, maintaining a constant setpoint while varying the control parameters. The resulting combinations of controller gains are summarized in Table 5. The system response for each combination was observed and analyzed to evaluate the performance of the controller, as depicted in Fig. 11. Fig. 12 displays the force exerted by the actuator throughout the system response analysis, providing a visual representation of how the force varies over time for different PID parameter combinations. The vehicle speed performance was assessed using the PID controller in conjunction with the Matlab Tuner, and the results are presented in Table 6.

Based on the system performance analysis, the average steady-state error, overshoot, rise time, and settling time of -1.93e-3, 23.06, 0.01674, and 0.146, the system performance indicates that the PID controller implemented using the Matlab PID Tuner provides better results for the speed trajectory compared to manual tuning. It achieves a faster response and a lower steady-state error.

Table 5. PID parameter for Matlab Tuner

No	Setpoint	PID Parameter		
		P	I	D
1	4 km/h	10.3921	71.7148	0.0572
2	4 km/h	6.4003	9.701	0.1364
3	4 km/h	5.4597	43.6498	0.14528
4	4 km/h	5.3044	43.7026	0.15261
5	4 km/h	7.7638	55.5875	0.2537

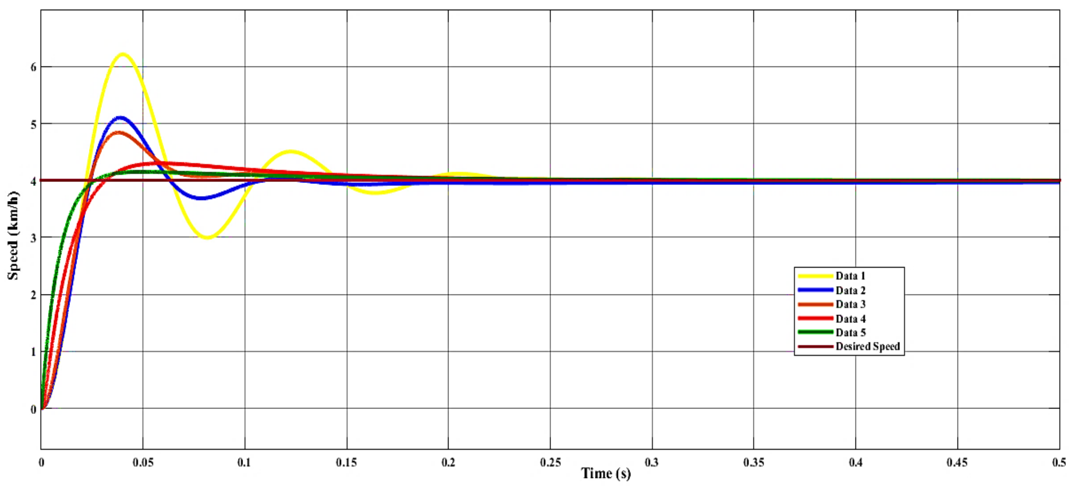

Fig. 11. Force response by applying Matlab Tuner.

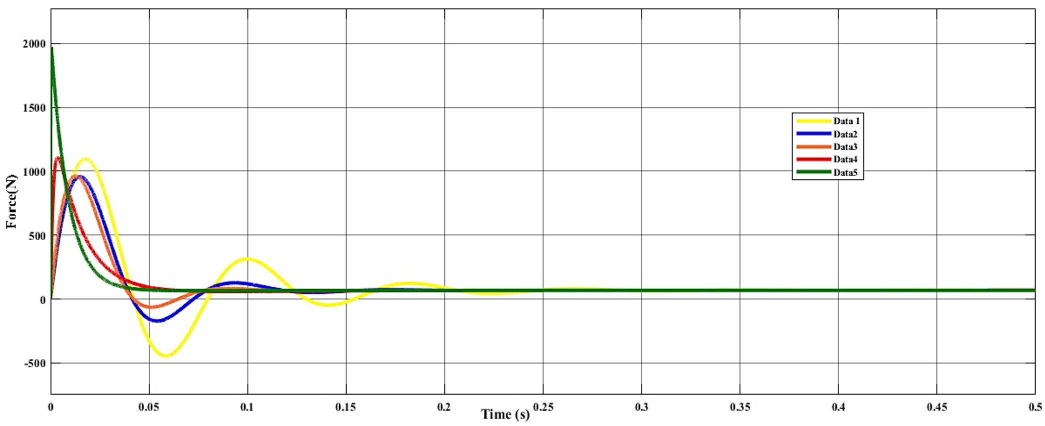

Fig. 12. Force response by applying Matlab Tuner.

Table 6. System Performance Analysis for Matlab Tuner

No	Setpoint	System Performance Analysis				
		Steady State Error	**Overshoot (%)**	**Rise Time(s)**	**Settling Time(s)**	**Force (N)**
1	Data1	-1.39e-5	55.3	0.0147	0.217	68
2	Data2	-1.03e-4	27.6	0.0163	0.0998	68
3	Data3	-2.28e-5	21.1	0.0162	0.14	68
4	Data4	-8.25e-4	7.52	0.0206	0.1562	68
5	Data5	-2.10e-6	3.78	0.0159	0.1168	68
	Average	-1.93e-4	23.06	0.01674	0.146	68

4.4. Robustness Analysis by Applying Matlab Tuner

The same testing process was conducted using the MATLAB Tuner tool. The results showed in Fig. 13 and Fig. 14, the system responded much faster and had a lower steady-state error compared to the manual tuning method. Consequently, it can be concluded that the MATLAB Tuner provides a better tuning solution for achieving enhanced system performance and response in terms of speed control.

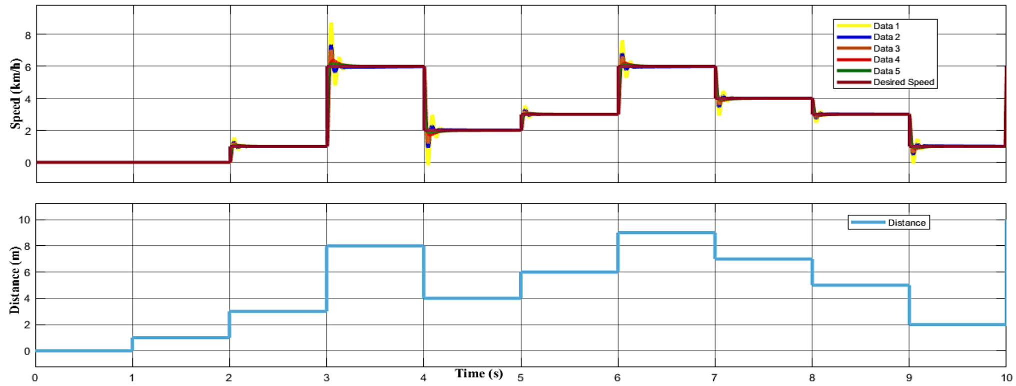

Fig. 13. System response with variation speed and distance

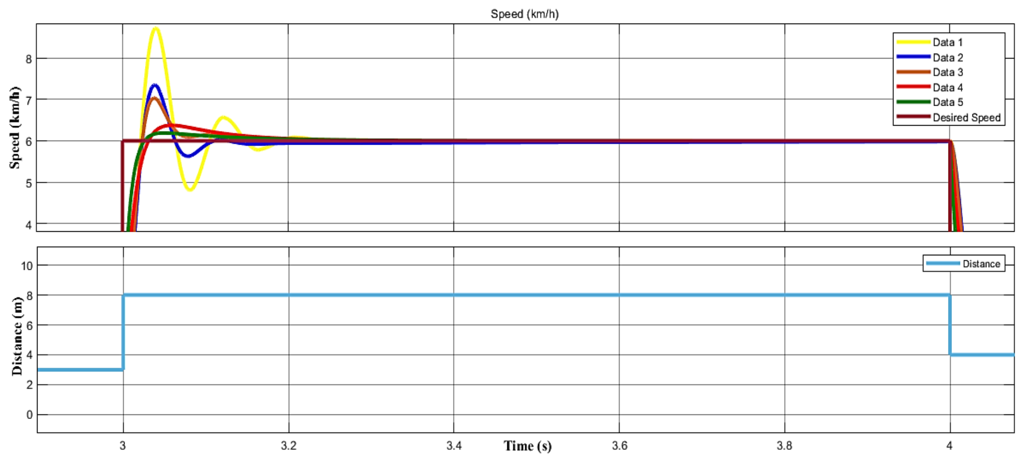

Fig. 14. Zoom view of the system response from 3s – 4s

4.5. Discussion

Based on the system performance results, the Matlab tuner demonstrates advantages over the manual tuning method, including a faster response and reduced steady-state error. However, the manual tuning method has the advantage of producing a response with negligible overshoot. Despite this benefit, manual tuning typically results in a delayed response when achieving the desired speed. Consequently, while the manual tuning method provides a more stable response without oscillation, it may result in a delay in reaching the desired speed. Although manual tuning gives lower overshoot, this can be mitigated by fine-tuning the PID controller parameters within the Matlab PID Tuner, offering a more efficient and time-saving tuning process.

Additionally, it is crucial to recognize that the applicability of this system may be restricted to driving scenarios and conditions within the scope of this study. The system's performance may be compromised if the desired speed exceeds 50 km/h. To address this limitation, rigorous simulation testing is necessary. Integrating advanced control algorithms like fuzzy logic controllers may enhance the system's performance. Therefore, it is important to note that this study primarily focuses on low-speed driving conditions, specifically within the range of 0 km/h to 20 km/h, such as in traffic congestion situations.

Overall, the results indicate that integrating a linear actuator to replace leg activities for controlling vehicle speed, particularly in low-speed driving conditions, yields a fast response and low settling time. This system has the potential to replace leg activities by utilizing a linear actuator to control the pressing and releasing of the brake pedal, effectively managing the vehicle's speed. Consequently, it can be implemented in a real hardware prototype where the linear actuator is attached to the pedal brake. This implementation would reduce the need for frequent pedal pressing and releasing, providing relief to the driver's leg and preventing fatigue.

However, implementing the system into a real hardware prototype can pose significant challenges during real-time testing, including the potential for hardware failures or malfunctions that could lead to equipment damage. To address this, future work will involve validating the system through real-time testing using the hardware-in-the-loop technique. This approach will provide a comprehensive understanding of the system's behavior and performance under realistic conditions, effectively bridging the gap between simulation and physical implementation, and ensuring a more robust and reliable system.

5. Conclusion

This paper introduces a PID controller as a means to effectively control the linear actuator and regulate the speed of the vehicle. By integrating the actuator with the vehicle's dynamics model, the control system ensures that the actuator exerts the right amount of force on the pedal, resulting in precise speed control. The system's performance is compared using two different tuning methods: the Matlab tuner and trial and error. The results demonstrate that the Matlab tuner surpasses the trial and error method by minimizing steady state error and achieving a fast system response, effectively maintaining the actuator force to reach the desired speed. However, it is important to address a slight drawback of the Matlab tuner, which exhibits a slightly higher overshoot during the transient state. This issue can be resolved by fine-tuning the PID parameters within the Matlab tuner, which offers user-friendly control and saves time during the tuning process. Additionally, it is important to note that this system is only applicable in low-speed driving conditions. The performance of the system may be compromised if the intended speed exceeds 50 km/h. As an option for improving system performance, a fuzzy logic controller can be implemented to improve system performance in various driving condition scenarios. This study demonstrates that by integrating a linear actuator as the mechanism to press and release the pedal brake, the vehicle's speed can be effectively controlled. This demonstrates that the system has the potential to be implemented in a real-world hardware prototype due to the potential to replace leg activities for frequently pressing and releasing the brake pedal, thus freeing the driver's leg and preventing fatigue. During real-time testing, implementing the system into an actual hardware prototype can pose significant challenges. Therefore, future work involves validating the system through hardware-in-the-loop testing in real time. This strategy will provide a comprehensive understanding of the system's behavior and performance under realistic conditions, bridging the gap between simulation and physical implementation and ensuring a more robust and effective system. Through the validation process, the proposed system can be implemented in car systems, effectively improving driving comfort by reducing fatigue, particularly on the leg and back. By automating the pedal pressing mechanism and integrating it into the car system, the system minimizes the physical strain associated with repetitive leg movements, leading to enhanced comfort and reduced fatigue for drivers.

Author Contribution: All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.

Funding: This work has been financially supported by the Ministry of Higher Education, Malaysia (MOHE) under the Fundamental Research Grant Scheme (FRGS/1/2021/TK0/UIAM/02/27) with a university reference number (FRGS21-245-0854) and the Fundamental Research Grant Scheme (FRGS/1/2021/ICT02/UMP/02/2) with the university reference number (RDU210110).

Acknowledgement: The author would like to thank the Ministry of Higher Education, Malaysia (MOHE) for financial funding under the Fundamental Research Grant Scheme (FRGS/1/2021/TK0/UIAM/02/27) with a university reference number (FRGS21-245-0854) and the Fundamental Research Grant Scheme (FRGS/1/2021/ICT02/UMP/02/2) with the university reference number (RDU210110).

Conflicts of Interest: The authors declare no conflict of interest.

References

[1] M. Jagannath and V. Balasubramanian, "Assessment of early onset of driver fatigue using multimodal fatigue measures in a static simulator," *Appl. Ergon*, vol. 45, no. 4, pp. 1140–1147, 2014, https://doi.org/10.1016/j.apergo.2014.02.001.

[2] I. Hostens and H. Ramon, "Assessment of muscle fatigue in low level monotonous task performance during car driving," *Journal of Electromyography and Kinesiology*, vol. 15, no. 3, pp. 266–274, 2005, https://doi.org/10.1016/j.jelekin.2004.08.002.

[3] S. A. Oyewole, "Enhancing ergonomic safety effectiveness of repetitive job activities: Prediction of muscle fatigue in dominant and nondominant arms of industrial workers," *Hum Factors Ergon Manuf*, vol. 24, no. 6, pp. 585–600, 2014, https://doi.org/10.1002/hfm.20590.

[4] S. Bahri, *et al.*, "The Association between Risk Factors and Low Back Pain among Commercial Vehicle Drivers in Peninsular Malaysia: A Preliminary Result," *Industrial Health*, vol. 45, no. 2, pp. 268-278, 2007, https://doi.org/10.2486/indhealth.45.268.

[5] I. Yusoff, B. K. Ng, and S. A. Azizan, "Towards sustainable transport policy framework: A rail-based transit system in Klang Valley, Malaysia," *PloS one*, vol. 16, no. 3, p. e0248519, 2021, https://doi.org/10.1371/journal.pone.0248519.

[6] P. H. Ting, J. R. Hwang, J. L. Doong, and M. C. Jeng, "Driver fatigue and highway driving: A simulator study," *Physiol. Behav.*, vol. 94, no. 3, pp. 448–453, 2008, https://doi.org/10.1016/j.physbeh.2008.02.015.

[7] J. F. Carneiro, J. B. Pinto, and F. G. D. Almeida, "Accurate motion control of a pneumatic linear peristaltic actuator," *Actuators*, vol. 9, no. 3, 2020, https://doi.org/10.3390/act9030063.

[8] T. Driver and X. Shen, "Pressure estimation-based robust force control of pneumatic actuators," *International Journal of Fluid Power*, vol. 14, no. 1, pp. 37–45, 2013, https://doi.org/10.1080/14399776.2013.10781067.

[9] P. Lucidarme, N. Delanoue, F. Mercier, Y. Aoustin, C. Chevallereau, and P. Wenger, "Preliminary survey of backdrivable linear actuators for humanoid robots," *ROMANSY 22 – Robot Design, Dynamics and Control*, vol. 584, 2018, pp. 304-313, https://doi.org/10.1007/978-3-319-78963-7_39.

[10] T. Noguchi and A. Kawamura, "Sensorless position control of novel electromagnetic linear actuator for tactile display," *2017 20th International Conference on Electrical Machines and Systems (ICEMS)*, 2017, pp. 1-5, https://doi.org/10.1109/ICEMS.2017.8056160.

[11] B. Rouzbeh and G. M. Bone, "Optimal force allocation and position control of hybrid pneumatic-electric linear actuators," *Actuators*, vol. 9, no. 3, p.86, 2020, https://doi.org/10.3390/act9030086.

[12] I. Dominik, "Advanced controlling of the prototype of SMA linear actuator," *Solid State Phenomena*, vol. 177, pp. 93–101, 2011, https://doi.org/10.4028/www.scientific.net/SSP.177.93.

[13] G. Wrat, M. Bhola, P. Ranjan, S. K. Mishra, and J. Das, "Energy saving and Fuzzy-PID position control of electro-hydraulic system by leakage compensation through proportional flow control valve," *ISA Trans*., vol. 101, pp. 269–280, 2020, https://doi.org/10.1016/j.isatra.2020.01.003.

[14] W. Y. Lai, N. Onn, C. H. H. Tang, and M. Hussein, "Position Control of Hydraulic Actuators Using Fuzzy Pulse Width Modulation (PWM)," *Applied Mechanics and Materials*, vol. 735, pp. 294–298, 2015, https://doi.org/10.4028/www.scientific.net/AMM.735.294.

[15] Z.-Y. Zhao, M. Tomizuka, and S. Isaka, "Fuzzy gain scheduling of PID controllers," *IEEE Trans Syst Man Cybern*, vol. 23, no. 5, pp. 1392–1398, 1993, https://doi.org/10.1109/21.260670.

[16] K. C. Sio and C. K. Lee, "Stability of fuzzy PID controllers," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 28, no. 4, pp. 490-495, 1998, https://doi.org/10.1109/3468.686710.

[17] S. Tzafestas and N. P. Papanikolopoulos, "Incremental fuzzy expert PID control," *IEEE Transactions on Industrial Electronics*, vol. 37, no. 5, pp. 365-371, 1990, https://doi.org/10.1109/41.103431.

[18] S. Das, B. Bandyopadyay, A. K. Paul, and P. Trivedi, "Position control of pneumatic actuator using sliding mode control in conjunction with Robust Exact Differentiator," *2013 International Conference on Control, Automation, Robotics and Embedded Systems (CARE)*, pp. 1-6, 2013, https://doi.org/10.1109/CARE.2013.6733689.

[19] E. Sariyildiz, H. Wang, and H. Yu, "A sliding mode controller design for the robust position control problem of series elastic actuators," *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3055-3061, 2017, https://doi.org/10.1109/ICRA.2017.7989350.

[20] L. Yu, J. Huang, W. Luo, S. Chang, H. Sun, and H. Tian, "Sliding-Mode Control for PMLSM Position Control—A Review," *Actuators*, vol. 12, no. 1, p. 31, 2023, https://doi.org/10.3390/act12010031.

[21] R. Hernandez-Alvarado, O. Rodriguez-Abreo, J. M. Garcia-Guendulain, and T. Hernandez-Diaz, "Self-Tuning Control Using an Online-Trained Neural Network to Position a Linear Actuator," *Micromachines*, vol. 13, no. 5, p. 696, 2022, https://doi.org/10.3390/mi13050696.

[22] J. Zhao, J. Zhong, and J. Fan, "Position control of a pneumatic muscle actuator using RBF neural network tuned PID controller," *Math. Probl. Eng.*, vol. 2015, 2015, https://doi.org/10.1155/2015/810231.

[23] G. S. Choi, H. K. Lee, and G. H. Choi, "A study on tracking position control of pneumatic actuators using neural network," *IECON '98. Proceedings of the 24th Annual Conference of the IEEE Industrial Electronics Society (Cat. No.98CH36200)*, pp. 1749–1753, 1998, https://doi.org/10.1109/IECON.1998.722946.

[24] R. P. Borase, D. K. Maghade, S. Y. Sondkar, and S. N. Pawar, "A review of PID control, tuning methods and applications," *International Journal of Dynamics and Control*, vol. 9, pp. 818–827, 2021, https://doi.org/10.1007/s40435-020-00665-4.

[25] M. I. Karim, A. A. A. Hashim, and N. M. A. Ghani, "Control of Double Link Flexible Robotic Manipulator System," *Journal Europeen des Systemes Automatises*, vol. 55, no. 6, pp. 759–763, 2022, https://doi.org/10.18280/jesa.550607.

[26] M. Khalifa, A. H. Amhedb, and M. Al Sharqawi, "Real time DC motor position control using PID controller in LabVIEW," *Journal of Robotics and Control (JRC)*, vol. 2, no. 5, pp. 342–348, 2021, http://dx.doi.org/10.18196/jrc.25104.

[27] V. Kumar and A. Patra, "Application of Ziegler-Nichols Method for Tuning of PID Controller," *International Journal of Electrical and Electronics Engineers*, vol. 8, no. 2, 2017, pp. 559-570, http://www.arresearchpublication.com/images/shortpdf/1479279897_127ijeee.pdf.

[28] V. V. Patel, "Ziegler-Nichols Tuning Method: Understanding the PID Controller," *Resonance*, vol. 25, pp. 1385–1397, 2020, https://doi.org/10.1007/s12045-020-1058-z.

[29] K. M. Elbayomy, Z. Jiao, and H. Zhang, “PID controller optimization by GA and its performances on the electro-hydraulic servo control system,” *Chinese Journal of Aeronautics*, vol. 21, no. 4, pp. 378–384, 2008, https://doi.org/10.1016/S1000-9361(08)60049-7.

[30] A. A. Aly, “PID Parameters Optimization Using Genetic Algorithm Technique for Electrohydraulic Servo Control System,” Intelligent Control and Automation, vol. 02, no. 02, pp. 69–76, 2011, http://dx.doi.org/10.4236/ica.2011.22008.

[31] A. M. Sadri, J. R. Wright, and R. J. Wynne, “Modelling and optimal placement of piezoelectric actuators in isotropic plates using genetic algorithms,” *Smart Materials and Structures*, vol. 8, no. 4, p. 490, 1999, https://doi.org/10.1088/0964-1726/8/4/306.

[32] M. I. Solihin, L. F. Tack, and M. L. Kean, “Tuning of PID Controller Using Particle Swarm Optimization (PSO),” *Int. J. Adv. Sci. Eng. Inf. Technol.*, vol. 1, no. 4, p. 458-461, 2011, http://dx.doi.org/10.18517/ijaseit.1.4.93.

[33] L. Jia and X. Zhao, “An Improved Particle Swarm Optimization (PSO) Optimized Integral Separation PID and its Application on Central Position Control System,” *IEEE Sensors Journal*, vol. 19, no. 16, pp. 7064-7071, 2019, https://doi.org/10.1109/JSEN.2019.2912849.

[34] N. H. Sunar, M. F. Rahmat, Z. H. Ismail, A. M. Faudzi, S. N. S. Salim, and S. I. Samsudin, “Application of optimization technique for PID controller tuning in position tracking of pneumatic actuator system,” *2013 IEEE 9th International Colloquium on Signal Processing and its Applications*, pp. 33-38, 2013, doi: https://doi.org/10.1109/CSPA.2013.6530009.

[35] Zulfatman and M. F. Rahmat, “Application of Self-Tuning Fuzzy Pid Controller on Industrial Hydraulic Actuator Using System Identification Approach,” *International Journal on Smart Sensing and Intelligent Systems*, vol. 2, no. 2, pp. 246–261, 2009, https://doi.org/10.21307/ijssis-2017-349.

[36] G. -R. Yu, C. -S. You, and R. -J. Hong, “Self-Tuning Fuzzy Control of a Piezoelectric Actuator System,” *2006 IEEE International Conference on Systems, Man and Cybernetics*, pp. 1108-1113, 2006, https://doi.org/10.1109/ICSMC.2006.384548.

[37] D. Saputra, A. Ma’arif, H. Maghfiroh, P. Chotikunnan, and S. N. Rahmadhia, “Design and Application of PLC-based Speed Control for DC Motor Using PID with Identification System and MATLAB Tuner,” *International Journal of Robotics and Control Systems*, vol. 3, no. 2, pp. 233–244, 2023, http://dx.doi.org/10.31763/ijrcs.v3i2.775.

[38] K. J. Åström and T. Hägglund, “The future of PID control,” *Control Eng Pract*, vol. 9, no. 11, pp. 1163–1175, 2001, https://doi.org/10.1016/S0967-0661(01)00062-4.

[39] L. H. Keel, “Analytical Design of PID Controllers [Bookshelf],” *IEEE Control Systems Magazine*, vol. 41, no. 1, pp. 80-81, 2021, https://doi.org/10.1109/MCS.2020.3032803.

[40] G. R. Wicaksono and R. D. Puriyanto, “Programmable Logic Controller (PLC) Based Paint Viscosity Control System,” *Buletin Ilmiah Sarjana Teknik Elektro*, vol. 3, no. 1, pp. 1–9, 2021, https://doi.org/10.12928/biste.v3i1.1109.