SEBHA UNIVERSITY JOURNAL OF PURE & APPLIED SCIENCES VOL.24 NO. 3 2025
DOI: 10.51984/JOPAS.V24I3.4046

مجلة جامعة سبها للعلوم البحتة والتطبيقية
Sebha University Journal of Pure & Applied Sciences

Journal homepage: www.sebhau.edu.ly/journal/jopas

Real-Time Implementation and Performance Comparison of PID and Fuzzy Logic Controllers for Industrial Temperature Regulation

Sana Doughman*, Nizar Khemri, Nuredin Tabib, Nuredin Ahmed

Computer Engineering Department, Faculty of Engineering, University of Tripoli, Tripoli, Libya.

Keywords:
Fuzzy Logic Control.
PID.
Temperature Control.
Temperature Process Station Model.*(3504)*.
Real-Time Control.
Membership Functions.
Fuzzy Logic Rules.

A B S T R A C T

This paper aims to achieve precise real-time temperature control in the Temperature Process Station Model (3504) by leveraging the capabilities of fuzzy logic control (FLC). Compared to conventional Proportional–Integral–Derivative (PID) controllers, FLC demonstrates the potential for enhanced performance and accuracy. The implementation of this research was facilitated through the utilisation of the Arduino Support Package, establishing a reliable connection between an Arduino board and MATLAB for effective interfacing and real-time control of the temperature process station. The results of this study show that the FLC outperformed the PID controller in real-time operation. Specifically, the FLC achieved a lower overshoot (1.45% compared to 1.875% for the PID), a faster rise time (396.6 seconds compared to 856 seconds), and a smaller time constant (198 seconds compared to 358 seconds). These outcomes highlight FLC as a more efficient and reliable control approach for temperature regulation in the Temperature Process Station Model (3504).

مقارنة متحكم PID ومتحكم المنطق الضبابي على نظام التحكم في عملية درجة الحرارة الصناعي في الزمن الحقيقي

سناء دغمان*، نزار الخمري، نورالدين الطبيب، ونورالدين أحمد

قسم هندسة الحاسب، كلية الهندسة، جامعة طرابلس، طرابلس، ليبيا.

الكلمات المفتاحية:
التحكم المنطقي الضبابي.
PID.
التحكم في درجة الحرارة.
نموذج محطة عملية درجة الحرارة (3504).
التحكم في الوقت الفعلي.
وظائف العضوية.
قواعد المنطق الضبابي.

الملخص

تهدف هذه الورقة إلى تحقيق تحكم دقيق في درجة الحرارة في الوقت الفعلي في نموذج محطة عملية درجة الحرارة (3504) من خلال الاستفادة من قدرات التحكم المنطقي الضبابي (FLC). مقارنةً بأجهزة التحكم التقليدية النسبية-التكاملية-المشتقة (PID)، يُظهر التحكم المنطقي الضبابي إمكانية تعزيز الأداء والدقة. يتم تسهيل تنفيذ هذا البحث من خلال الاستخدام الناجح لحزمة دعم Arduino، مما ينشئ اتصالاً موثوقًا بين لوحة Arduino وMATLAB للتحكم الفعال في الوقت الفعلي لمحطة عملية درجة الحرارة المذكورة. تظهر نتائج هذه الدراسة أن التحكم المنطقي الضبابي تفوق على وحدة تحكم PID في التحكم في الوقت الفعلي. على وجه التحديد، حققَ التحكم المنطقي الضبابي زيادة أقل (1.45٪ مقارنة بـ 1.875٪ لـ PID)، ووقت صعود أسرع (396.6 ثانية مقارنة بـ 856 ثانية لـ PID)، وثابت زمني أصغر (198 ثانية مقارنة بـ 358 ثانية لـ PID). تسلط هذه النتائج الضوء على التحكم المنطقي الضبابي كنهج تحكم أكثر كفاءة وموثوقية لتنظيم درجة الحرارة في نموذج محطة عملية درجة الحرارة (3504).

1. Introduction

Control systems development remains a cornerstone of modern engineering, driving innovation in industrial automation, robotics, energy management, and process control. The increasing complexity of physical systems requires advanced control strategies capable of dealing with nonlinearities, uncertainties, and dynamic variations. Among conventional techniques, the Proportional–Integral–Derivative (PID) controller remains the most commonly used due to its simplicity and proven reliability. However, its performance deteriorates in nonlinear systems or under varying operating

*Corresponding author.
E-mail addresses: s.marzouk@uot.edu.ly, (N. Khemri) n.khemri@uot.edu.ly, (N. Tabib) N.Tabib@gmail.com, (N. Ahmed) nu.ahmed@uot.edu.ly.
Article History : Received 19 May 25 - Received in revised form 23 October 25 - Accepted 17 November 25

conditions, mainly due to difficulties in precise tuning and the assumption of system linearity.

In contrast, Fuzzy Logic Control (FLC) has emerged as a promising alternative that emulates human reasoning and decision-making to handle imprecision and nonlinearity effectively. FLC relies on linguistic rules and membership functions rather than exact mathematical models, making it suitable for systems where accurate modelling is challenging. Previous research has demonstrated that FLC can offer superior performance to PID in terms of rise time, overshoot, settling time, and steady-state accuracy. However, the majority of existing work is centred on simulation rather than real-time application on industrial hardware.

Fuzzy control has been recognised as a highly active research area, particularly in industrial processes where quantitative relationships between inputs and outputs are difficult to define [1]. FLC provides a framework for translating expert knowledge into automated control strategies through fuzzification, rule evaluation, and defuzzification mechanisms [2]. It has been successfully applied in various systems, including automatic train control, flight systems, and refrigerant flow control, where it demonstrated improved performance compared to commercial controllers [3], [4]. Studies also indicate that neuro-fuzzy controllers can outperform PID by reducing oscillations and improving settling time, particularly in systems characterised by uncertainty [5], [6].

This paper aims to contribute to this body of knowledge by conducting a real-time experimental comparison between FLC and PID controllers using the LAB-VOLT Temperature Process Station Model (3504). The novelty of this work lies in applying a hardware-in-the-loop approach using the Arduino Support Package integrated with MATLAB/Simulink for direct real-time industrial temperature control. The study compares dynamic performance parameters such as rise time, overshoot, and time constant, while also evaluating computational requirements and robustness under real operating conditions.

In recent years, an increasing number of studies have compared PID and FLC across a wide range of industrial applications. Table 1 summarizing literature review findings.

Table 1. Summary of literature review findings.

Application	Conclusion	Year	#
General Control Systems	FLCs offer superior performance with no overshoot, zero steady-state error, and smaller settling time compared to PID.	2007	[7]
Liquid Flow Control	FLCs are preferred for high precision and quick adjustments due to their stability, low overshoot, and fast response times.	2012	[5]
Temperature Control Systems	FLCs enhance industrial automation and control engineering by providing robust control and adaptability.	2012	[8]
Water Level Control	FLCs provide more accurate control of water levels compared to PID.	2016	[9]
Cruise Control Systems	FLC outperforms PID and GA-PID in terms of overshoot, settling time, and steady-state error.	2018	[10]
Magnetic Levitation System	PID performs well across a wide range of operating conditions and minimizes overshoot. The source does not explicitly compare the FLC to the PID.	2018	[11]
DC Motor Speed Control	Fuzzy-PID controller offers more responsive and stable control for speed regulation in DC motors compared to standard PID.	2019	[12]
General Control Systems	FLC has superior performance in terms of speed and precision, with a faster response and smaller overshoot compared to PID.	2020	[13]
Water Level Control	FLC provides better performance in terms of overshoot and settling time, especially in the presence of time delays.	2021	[14]
Industrial Applications	FLC provides a stable and oscillation-free response, while the PID controller exhibits oscillatory behaviour that could potentially damage the system.	2022	[15]
real-time DC motor speed control	FLCs more computationally intensive, offer superior performance in systems that require high adaptability to changing conditions.	2024	[16]
DC motor	FLC excel in nonlinear and uncertain systems, providing robustness and adaptability, though they come with increased computational complexity.	2024	[17]

1.1. LAB-VOLT - Temperature Process Station Model 3504

The LAB-VOLT Temperature Process Station, model 3504, shown in Fig. 1, mainly consists of a (20-200) degree C (70-400-degree F) oven with a built-in capillary-bulb temperature switch (on/off controller), thermostat, air cooling injector, adjustable damper, and overheating protection.

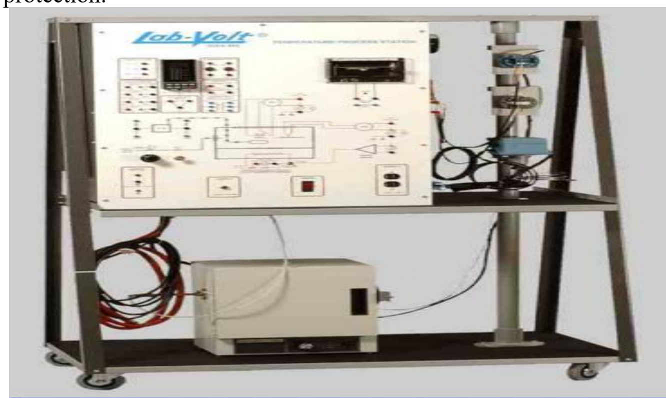

Fig. 1: LAB-VOLT - temperature process station Model 3504.

The process instrumentation includes a capillary-bulb thermometer mounted on the side of the oven, as well as a Resistance Temperature Detector (RTD) and a J-type thermocouple temperature transmitter with electrical connections terminated by banana jacks on the main control panel. Control of the oven temperature can be achieved either manually by adjusting the thermostat and observing the oven temperature on the thermometer (on-off control), or remotely (PID control) by varying the amount of electrical power applied by a TRIAC driver to the heating element of the oven, using a (4-20 mA) signal. The air-cooling injector establishes a flow of air into the oven, thereby creating a cooling load on the process. The air pressure applied to this injector can be varied, using a pressure regulator and a needle valve, in order to change the process load. The oven damper can be used to change the process load and create disturbances. The temperature process workstation consists of a 20-200-degree Celsius (70-400-degree Fahrenheit) oven operated manually as an on-off process using a (24 V) dc relay, or proportionally controlled by a TRIAC driver with 4-20 mA input. The oven is modified with an air-cooling injector and adjustable damper for load and process disturbances, system voltages (120, 220, 240 V - 50/60) Hz. The unit features a pipe-mounted on-off capillary bulb temperature controller with two sets of contacts terminated at banana jacks on the main control panel and a toggle switch that changes control from the Triac driver to an (NC 24 VDC) relay for on-off control. Also featured are a pipe-mounted thermocouple to the current temperature transmitter, complete with type "J" thermocouple, and RTD to current transmitter, complete with 100 Ohms platinum RTD, with all supply and signal connections terminated at banana jacks on the main control panel. Mounted on the control panel are the microprocessor-based controller, strip chart recorder, two alarm lamps, and a pneumatic air regulator [16].

1.2. Fuzzy Logic Controller

Fuzzy logic concept was put forward by Lotfi. A. Zadeh, a professor at the University of California at Berkley. Fuzzy logic is a form of multi-valued logic that allows intermediate values to be defined between conventional evaluations like true/false, yes/no, high/low etc. Fuzzy logic controllers are used in various industrial processes for taking proper actions, like human control actions. Their simplicity makes it them a better choice over other traditional control technique. Fuzzy Logic Controllers techniques. FLC includes several parameters that need to be prior selected and configured, such as the selection of scaling factors, configurations of the canter and width of the membership functions (MF) and so on. The basic block diagram of FLC is shown in Fig. 2. Fuzzy logic control is derived from fuzzy set theory where; the transition lies between membership and non-membership can be graded. Therefore, boundaries of fuzzy sets can be indistinct and ambiguous, making it helpful for approximate systems. FLC is an alternative choice when specific mathematical formulations are not possible. Other advantages of FLC are:

• It needs less data storage in the form of membership functions and rules than conventional look up table for nonlinear controllers.

• It is more robust than other non-linear controllers.

The fuzzy logic controller is mainly composed of three principal elements. These are Fuzzification module (Fuzzifier), Rule base and Inference engine and Defuzzification module (Defuzzifier).

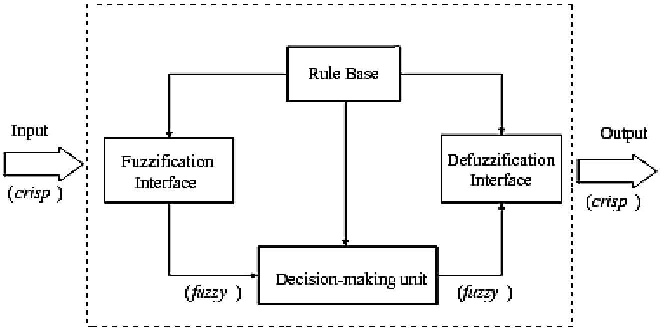

Fig. 2: Basic configuration of FLC system.

2. Methodology

The mathematical model of the Lab-Volt temperature process station involves defining system boundaries, making simplifying assumptions, applying the balance law, representing the system via block diagrams, and deriving transfer functions. The model is primarily based on the thermal energy balance and can be used for further analysis and control system design. An Arduino board is used in this work to establish digital communication between the temperature station (Model 3504) and a computer. Various Arduino models, such as the Uno, Mega, and Due, can be used. A comparison of these models reveals that the Arduino Due stands out as the most powerful and suitable for industrial applications, thanks to its superior processing power, high clock speed, ample memory, extensive I/O capabilities, and versatile connectivity. These features make it particularly well-suited for real-time, high-performance systems like the Temperature Process Station Model explored in this work.

2.1. Hardware and System Layout

The general scheme of connection of all hardware is depicted in Figure 4. The layout of the hardware setup, which includes the connection of the Arduino Due to various input/output system components for the control setup, is shown in this figure. In this setup, commands are sent to the Arduino Due by the (PC), which then interacts with the temperature process station to maintain the desired temperature, enabling real-time monitoring and control. Fig. 3, shows the block diagram for the system layout.

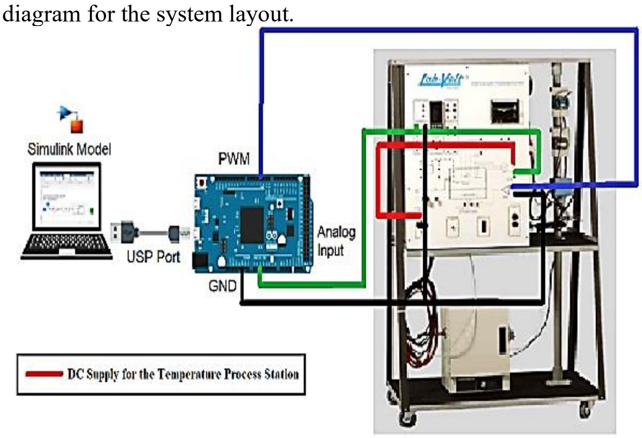

Fig. 3: General hardware setup layout.

2.2. Performance Comparison and Analysis of PID and Fuzzy Logic Controllers for Industrial Temperature Process Station Model (3504)

This part of the paper compares and analyses the performance of two control strategies: PID and FLC, for controlling the educational Temperature Process temperature process station Model (3504).

2.2.1. Step Responses System Identification using Arduino (Open Loop Method)

In this experiment, the step response of the Lab-volt temperature process station will be obtained using an Arduino board by following the procedure outlined below:

1. Set up the equipment needed for the experiment (turn on the AC power of the station and Air compressor and connect Arduino to the PC).
2. Wait for the temperature in the oven to be stable.
3. Set the desired temperature to 42°C in the MATLAB Simulink program with a sample time of one second.
4. Start the program and wait until the desired temperature is reached.
5. When the temperature stabilizes, as illustrated in Fig.4 save the step response data to the computer for analysis.

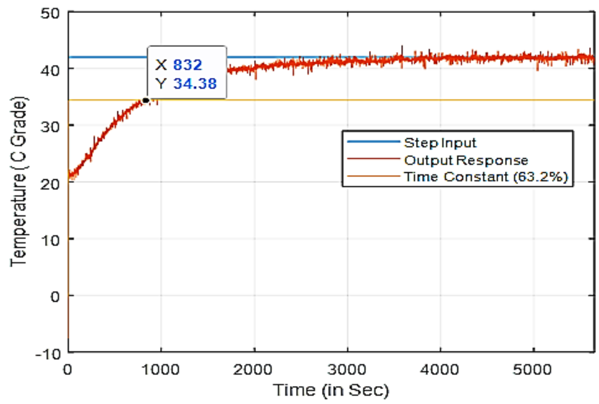

Fig. 4: Step response of lab-volt temperature process station using Arduino.

The figure shows a time constant obtained after 832 seconds. However, this time includes the time constant added to the time delay. To calculate the system's time delay, we zoom in on the initial samples of the figure, as shown in Fig. 5.

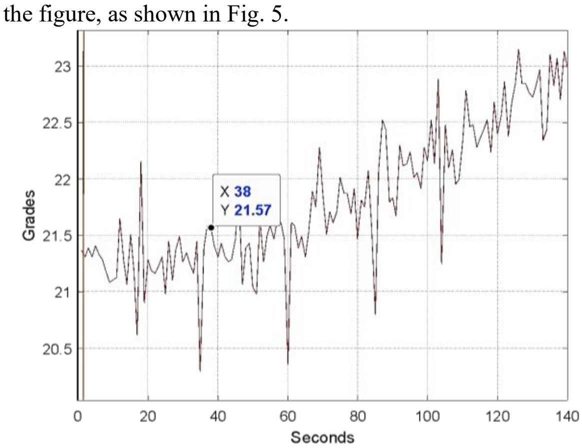

Fig. 5: Zoom of step response to compute time delay.

As shown in Fig. 6, the time delay occurred after 38 seconds. The first sample in the program was discarded due to this delay. From the above analysis, the values of time delay and time constant are:

Time delay = 38-1= 37 seconds.

Time constant = 832-(37+1) = 794 seconds

To calculate the process gain, which is defined as the ratio of output change to input change, the following information is needed:

The initial input and output temperature was 21.3°C. The final input temperature was 42°C.

Due to the inherent nature of digital systems, including quantization errors, the final output temperature was determined to be the average of the last 200 program output samples, which was 41.9656°C. Then, the process gain of the system is:

K=41.9656/42= 0.99818 which approximated to 1.

Then the final transfer function of the system is:

$$P(S) = \left(\frac{e^{-37s}}{794s + 1}\right)$$

2.2.2. PID Controller tuning in SIMULINK

PID Tuner provides a fast and widely applicable single loop PID tuning method for the Simulink PID controller blocks. This method enables achieving a robust design with the desired response time by tuning PID controller parameters.

A typical design workflow with the PID Tuner involves the following tasks:

1. To launch the PID Tuner, double-click the PID controller block to open its block dialog box. In the **Main** tab, click **Tune** as shown in Fig. 6.
2. Upon launch, the software automatically derives a linear plant model from the Simulink model and designs an initial controller.
3. Tune the controller in the PID Tuner by manually adjusting design criteria. The tuner calculates PID parameters that ensure robust system stabilization.
4. Export the parameters of the designed controller back to the PID controller block and verify controller performance in Simulink.

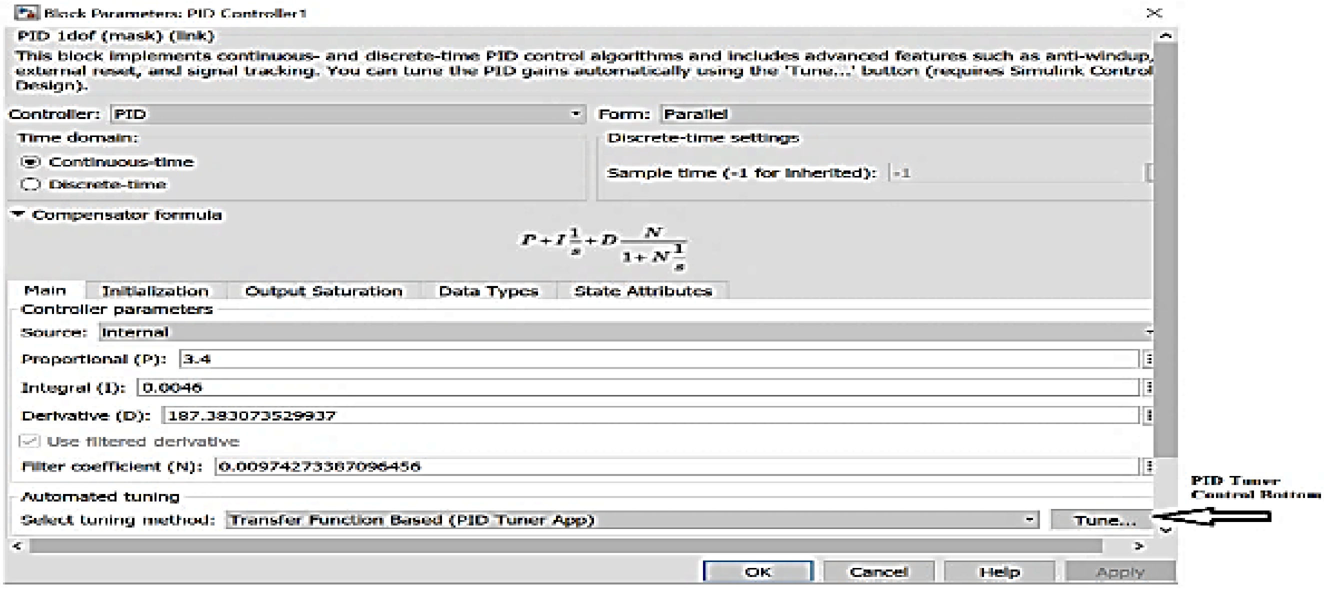

Fig. 6: PID controller dialog box.

The PID coefficients that given by PID tuner are:
P=3.4, I=0.0046, D=187.383 and the filter coefficient (N)=0.0097427.

2.2.3. Design of FLC

The block diagram illustrates the closed-loop nature of the fuzzy logic control system, highlighting its ability to handle imprecision and nonlinearity in the system dynamics. The FLC's rule-based approach allows it to adapt to varying conditions, making it a robust and effective control strategy for temperature regulation.

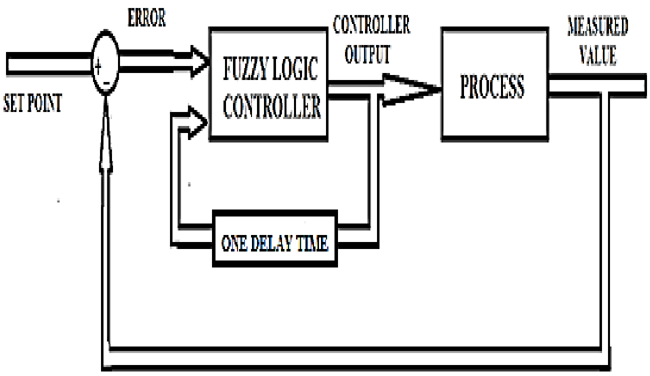

Fig. 7: Block diagram of FLC.

Using the mamandi base fuzzy inference system the FLC developed here is a two-input and single-output. The two inputs are the deviation from set point (**ERROR)** and the delay output of the controller (**CONTROL OUTPUT)**. The idea of using these two inputs is to firstly use the **ERROR** as a dominant parameter to decrease the rising time and quickly reach the transient response. Now, when the error is close to zero the, second input (**CONTROL OUTPUT)** is used to eliminate the overshoot by adjusting itself to the value of the desired temperature.

The FLC was built in the MATLAB environment and designed as shown in Fig. 8.

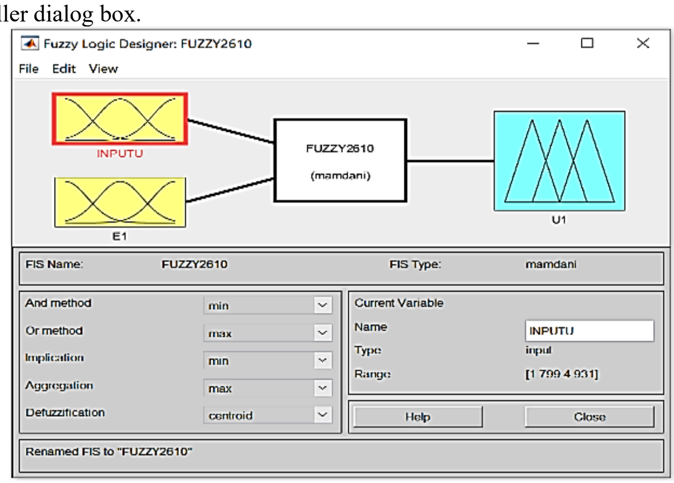

Fig. 8: The FLC that was built in MATLAB.

INPUTU input variable defines the delay of **CONTROL OUTPUT**, **E1** input variable defines the **ERROR,** and **U1** output variable defines **CONTROL OUTPUT**.

Fig. 9 shows the membership function of the first input variable of the FLC. The input variable, which represents the delay of the control output, is divided into nine distinct membership functions, each defined by a triangular shape. The range of the input variable is from 1.799 to 4.931.

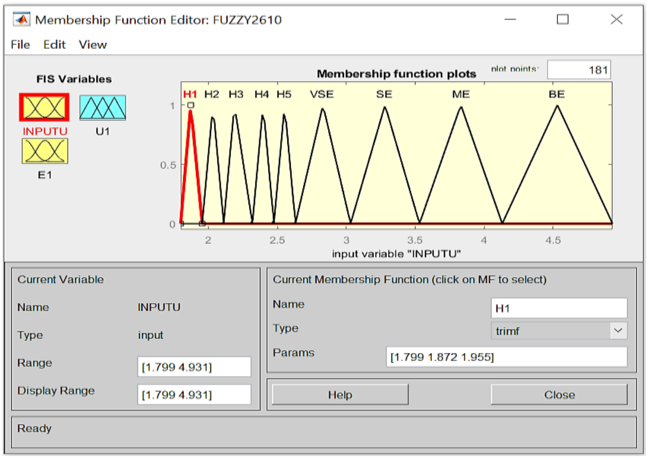

Fig. 9: Membership functions of first input variable of FLC.

Fig. 10 illustrates the membership functions for the second input variable of the FLC. The input variable, which represents the error, is divided into five distinct membership functions, each defined by a triangular shape. The range of the input variable is from -0.208 to 1.66.

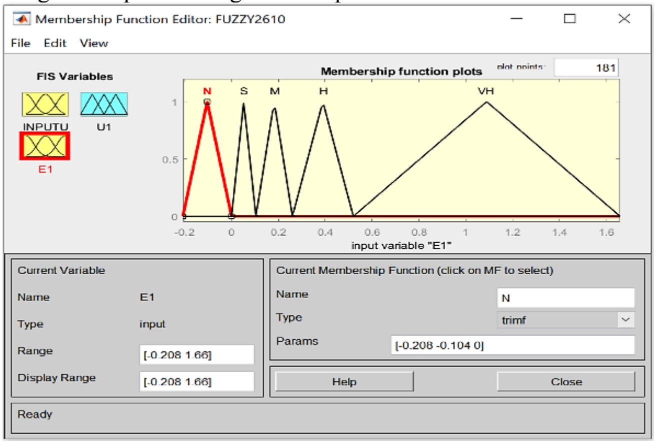

Fig.10: Membership functions of second input variable of FLC.

The membership function of the FLC's output variable is displayed in Fig. 11.

Using a triangular membership function, the output variable is in the range of 1.799 to 4.931.

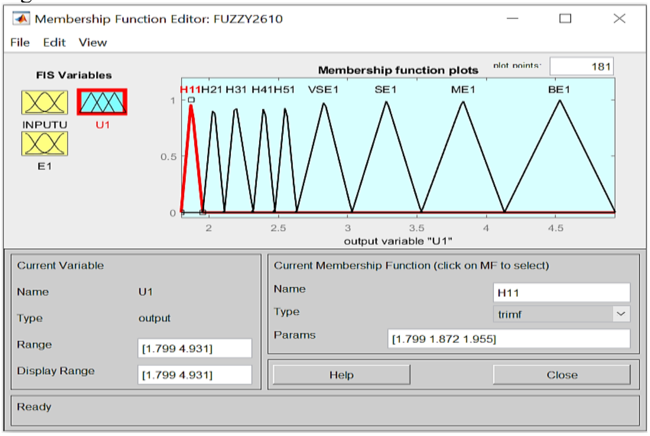

Fig. 11: Membership functions of output variable of FLC.

17 rules were used to control the process as shown in Fig. 12. Fig. 13 shows the rule's viewer action. The mechanism of rules is divided into three actions:

1. **Action 1:**

Table 2. shows Action 1.

IF statement	THEN statement
IF (E1 is VH)	THEN (U1 is BE1)
IF (E1 is H)	THEN (U1 is ME1)
IF (E1 is M)	THEN (U1 is SE1)
IF (E1 is S)	THEN (U1 is VSE1)

2. **Action 2:**

Table 3. shows Action 2.

IF statement	THEN statement
IF (INPUTU is H1) & (E1 is N)	THEN (U1 is H11)
IF (INPUTU is H1) & (E1 is S)	THEN (U1 is H21)
IF (INPUTU is H2) & (E1 is N)	THEN (U1 is H21)
IF (INPUTU is H2) & (E1 is S)	THEN (U1 is H31)
IF (INPUTU is H3) & (E1 is N)	THEN (U1 is H31)
IF (INPUTU is H3) & (E1 is S)	THEN (U1 is H41)
IF (INPUTU is H4) & (E1 is N)	THEN (U1 is H41)
IF (INPUTU is H4) & (E1 is S)	THEN (U1 is H51)
IF (INPUTU is H5)	THEN (U1 is H51)

3. **Action 3:**

1. The function of this action is returning to **Action 2** if any abnormal actions happened like disturbance...etc.

Table. 4. shows Action 3.

IF statement	THEN statement
IF (INPUTU is BE) & (E1 is N)	THEN (U1 is H11)
IF (INPUTU is SE) & (E1 is N)	THEN (U1 is H11)
IF (INPUTU is ME) & (E1 is N)	THEN (U1 is H11)
IF (INPUTU is VSE) & (E1 is N)	THEN (U1 is H11)

Fig. 13shows the rule viewer for the FLC which provides a detailed visualization of the fuzzy rules applied in the control system. The rule viewer shows how the input variables (e.g., error and delay of control output) are processed through the fuzzy inference system to generate the control output. Each rule is represented as an (IF-THEN) statement. The rule viewer allows for a clear understanding of how the FLC makes decisions based on the combination of input values and predefined rules, ensuring adaptive and precise control of the temperature process.

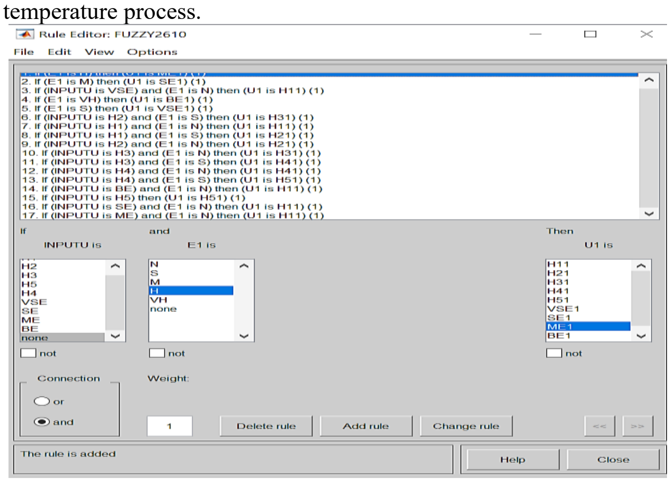

Fig. 12: Fuzzy logic Rule Viewer.

Fig. 14 shows the action of the rule viewer in the FLC which shows how specific input values are mapped to the corresponding output based on the fuzzy rules. The rule viewer action highlights the dynamic process of fuzzy inference, where the input variables (e.g., error and delay of control output) are evaluated against the rule base to determine the appropriate control action. The figure demonstrates the real-time decision-making process of the FLC, where the system adjusts the control output based on the degree of membership of the input variables in the fuzzy sets. This visualization helps to understand how the FLC adapts to varying conditions and ensures stable and accurate temperature.

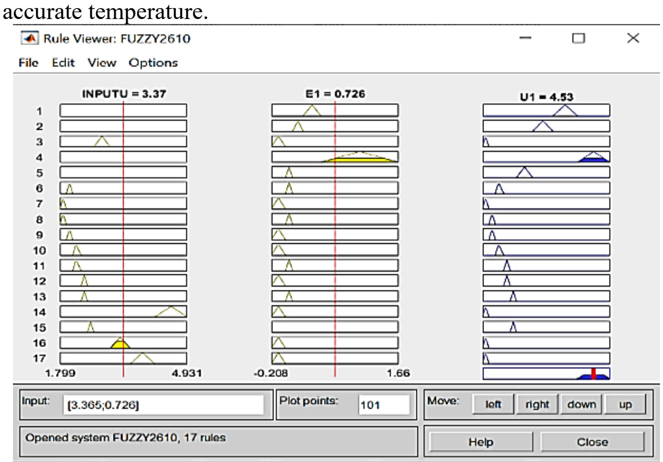

Fig. 13: Rule Viewer Action.

2.2.4. Real-Time experiments for temperature process station Model (3504)

This section presents the real-time experiments conducted using PID and FLC controllers to control the temperature process station Model (3504). The experiments were implemented using the Arduino Due board within the MATLAB Simulink environment. The focus of this part is on the design and implementation of the control strategies, as well as the analysis of their performance in real-time settings.

Experiment Setup:

- The Arduino Due board interfaces with the temperature process station Model (3504) to control the heating process.
- Commands are sent from the PC to the Arduino board, which interacts with the temperature station to maintain the desired temperature.
- Real-time monitoring and control are enabled through the Simulink platform, allowing for precise adjustments and data collection.

Experiment Objectives:

- Evaluate the performance of the PID and FLC controllers in real-time settings.

- Analyze key stability features such as rise time, settling time, and steady-state error.
- Compare the effectiveness of both controllers in achieving precise temperature control under real-world conditions.

The results of these real-time experiments are presented and analyzed in the following sections, providing insights into the practical performance of the PID and FLC controllers in controlling the temperature process station Model (3504).

1. Implementation of PID controller in Real-Time experiment

The SIMULINK platform implementation of PID controller for controlling lab volt temperature station via Arduino board is shown in Fig. 14.

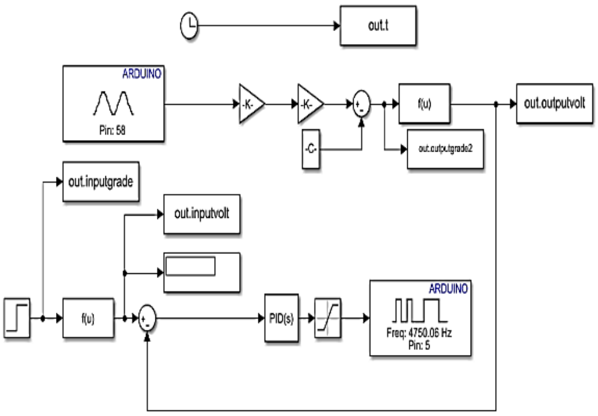

Fig. 14: SIMULINK platform implementation of PID controller.

The system response for this experiment is shown in Fig. 15 where it illustrates the real-time performance of the PID controller for a set point of 40°C. The graph shows the temperature control process over time, highlighting key performance metrics such as rise time, time constant, and overshoot. The system response demonstrates how the PID controller adjusts the temperature to reach and stabilize at the desired set point.

- Rise Time: The system reached the set point approximately after 856 seconds.
- Time Constant: The time constant, calculated as 63.2% of the set point (40°C - 24.16°C = 15.84°C; 15.84°C * 0.632 = 10.01°C; 24.16°C + 10.01°C = 34.17°C), occurred after 358.6 seconds.
- Overshoot: The maximum temperature reached was 40.75°C, resulting in an overshoot of 1.875% (calculated as ((40.75 - 40) / 40) * 100).
- Undershoot: The minimum temperature observed was 39.58°C, resulting in an undershoot of 1.05% (calculated as ((40 - 39.58) / 40) * 100).

Fig. 15 provides a clear visualization of the PID controller's ability to achieve precise temperature control in real-time, with minimal overshoot and undershoot. The extracted values from the graph offer a detailed evaluation of the PID controller's effectiveness in maintaining stable and accurate temperature regulation.

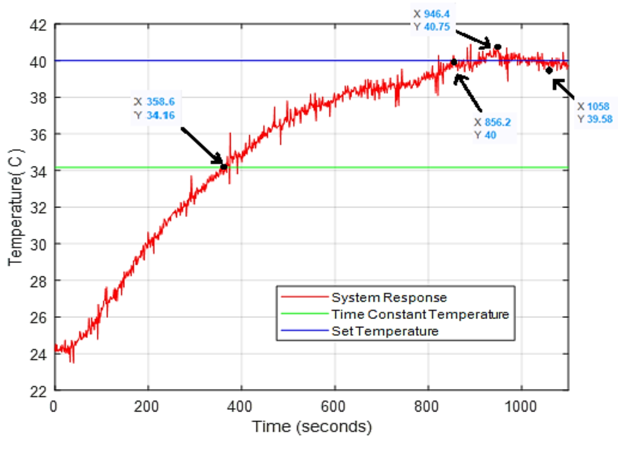

Fig. 15: PID real-time system response for setpoint of 40º C.

Fig. 16 shows again system response of the process with a bolded rectangles that zoomed in Fig. 18. to exactly calculate the time constant.

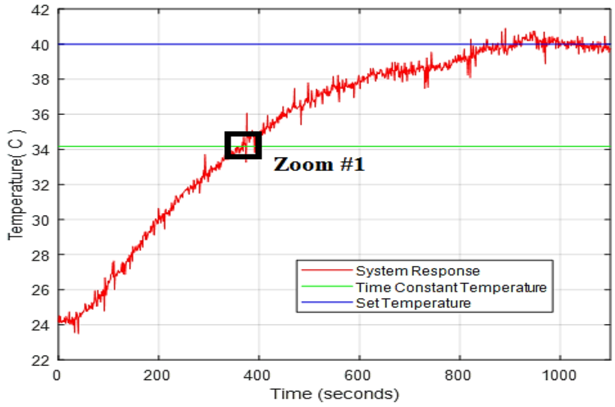

Fig. 16: PID real-time system response for set point of 40 ºC with zoomed rectangle.

Fig. 16 provides a zoomed-in view of the time constant region from Fig. 15, labeled as Zoom #1. It focuses on the specific portion of the temperature response curve where the system reaches 63.2% of the set point (34.17°C), which corresponds to the time constant.

The zoomed view allows for a more precise analysis of the time constant, showing how the (PID) controller achieves this critical point in the system's response.

The time constant is observed to occur at approximately 358.6 seconds, highlighting the (PID) controller's ability to stabilize the system efficiently.

2. Implementation of FLC in Real-Time experiment

Fig. 17 illustrates the Simulink platform implementation of the FLC for controlling the temperature process station Model (3504) in real-time. The model is built within the MATLAB/Simulink environment and integrates the key components of the FLC to achieve precise temperature regulation.

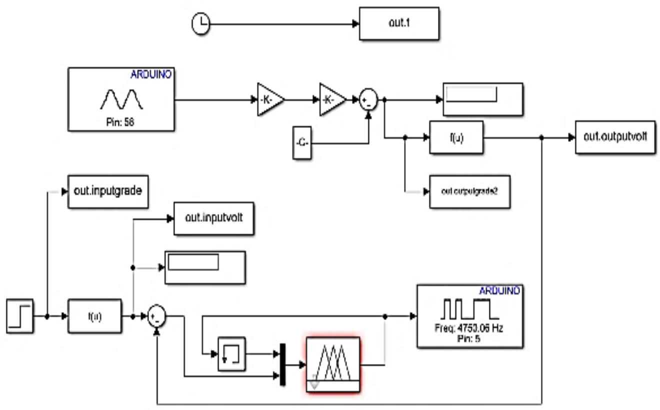

Fig. 17: Simulink platform implementation of FLC.

Fig. 18 illustrates the real-time performance of the FLC for a set point of 40°C. The graph shows the temperature control process over time, highlighting key performance metrics such as rise-time, time constant, and overshoot. The system response demonstrates how the FLC adjusts the temperature to reach and stabilize at the desired set point.

- Rise Time: The system reached the set point approximately after 396.6 seconds.
- Time Constant: The time constant, calculated as 63.2% of the set point (40°C - 24.16°C = 15.84°C; 15.84°C * 0.632 = 10.01°C; 24.16°C + 10.01°C = 34.17°C), occurred after 198.5 seconds.
- Overshoot: The maximum temperature reached was 40.58°C, resulting in an overshoot f 1.45% (calculated as ((40.58 - 40) / 40) * 100).
- Undershoot: The minimum temperature observed was 39.24°C, resulting in an undershoot of 1.9% (calculated as ((40 - 39.24) / 40) * 100).

The figure provides a clear visualization of the FLC's ability to achieve precise temperature control in real-time, with minimal overshoot and undershoot. The extracted values from the graph offer a detailed evaluation of the FLC's effectiveness in maintaining stable and accurate temperature regulation.

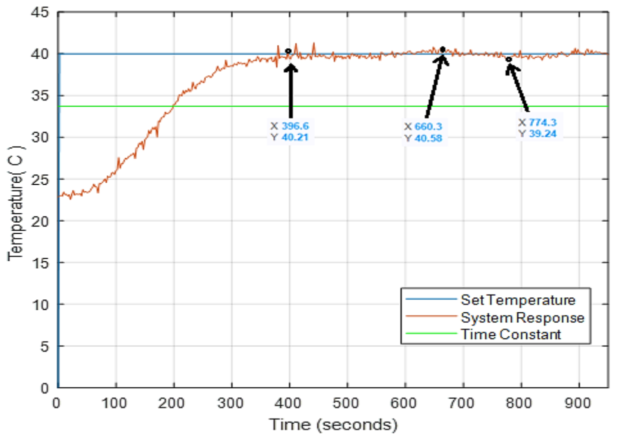

Fig. 18: FLC real time system response for a set point of 40 ºC.

Fig. 19 shows again the system response of the process with bolded rectangles that zoomed in Fig. 20 to exactly circulate the delay time and constant time.

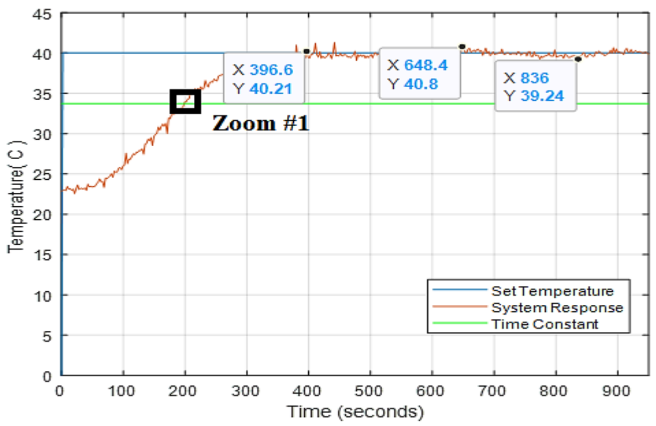

Fig. 19: FLC real-time system response for a set point of 40º c with zoomed rectangle.

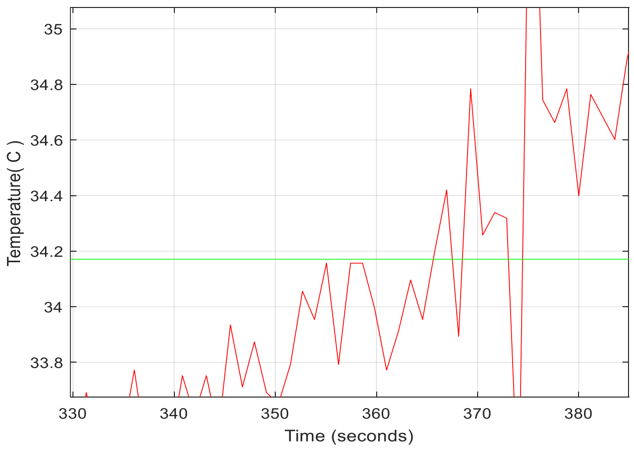

Fig. 20: Zoom #1 for Time Constant Value (PID-Real Time)

Fig20 provides a zoomed-in view of the time constant region from Fig19, labelled as Zoom #1. It focuses on the specific portion of the temperature response curve where the system reaches 63.2% of the set point (34.17°C), which corresponds to the time constant. The zoomed-in view enables a more detailed examination of the time constant, illustrating how the FLC reaches this critical point in the system's response. It was observed that the time constant occurred at approximately 198.5 seconds, highlighting the FLC's ability to efficiently stabilize the system.

2.2.5. Performance Comparison and Analysis of PID and Fuzzy Logic Controllers IN Real-Time

The real-time experimental results highlight the performance differences between the FLC and the PID controllers in terms of transient response characteristics: rise time, overshoot, and time constant. Below is the detailed analysis of the findings:

1.Rise Time:

The PID controller's rise time increased to 856 seconds, while the FLC achieved 396.6 seconds. FLC demonstrated superior performance, with a quicker rise time.

2.Overshoot:

The PID controller's overshoot was 1.875% which is slightly higher than the FLC's (1.45%). FLC performed better in real-time experiment regarding overshoot, demonstrating more controlled and stable behaviour.

3.Time Constant:

The PID controller's time constant increased to 358.6 seconds, while the FLC reduced to 198.5 seconds. FLC consistently outperformed the PID controller in real-time experiment, with a lower time constant, indicating faster system stabilization.

Table 5. Summarized Performance Parameters of PID and FLC control systems in real-time.

Appendix D. Controller Type	Appendix C. Time Constant (sec)	Appendix B. Rise Time (sec)	Appendix A. Overshoot (%)
Appendix H. PID	**Appendix G.** 358.6	**Appendix F.** 856	**Appendix E.** 1.875%
Appendix L. FLC	**Appendix K.** 198.5	**Appendix J.** 396.6	**Appendix I.** 1.45%

3. Discussion

Both controllers were tested under identical conditions.
PID: Overshoot = 1.875%, Rise Time = 856 s,
Time Constant = 358 s
FLC: Overshoot = 1.45%, Rise Time = 396.6 s,
Time Constant = 198 s

The FLC achieved faster response and smoother convergence. Its adaptive rule-based nature allowed dynamic adjustment of control effort, reducing overshoot and improving settling. In contrast, the PID controller's fixed gains resulted in slower response and minor oscillations. Theoretical analysis confirms that FLC performs better in nonlinear systems due to its ability to adapt to changing process behaviour without precise parameter knowledge.

4. Conclusion

This study presented a real-time comparison between PID and Fuzzy Logic Controllers for the LAB-VOLT Temperature Process Station (Model 3504). The FLC achieved significantly better performance in terms of rise time, overshoot, and time constant. Its adaptive nature allows precise temperature regulation without requiring detailed process modelling. Although the FLC requires higher computational effort and involves more design parameters, the results show that modern embedded platforms can support real-time fuzzy control efficiently. The main limitations of this study include evaluation at a single setpoint, the absence of intentional disturbance testing, and the use of a fixed rule base. Future research will focus on disturbance rejection tests, multi-setpoint tracking, and hybrid adaptive Fuzzy–PID or Neuro-Fuzzy designs to enhance robustness and learning capability. This work confirms that FLC provides a practical, intelligent, and efficient alternative to PID in real-time industrial control systems.

5. References

[1] R. Langari, "Past, present and future of fuzzy control: a case for application of fuzzy logic in hierarchical control," 18th International Conference of the North American Fuzzy Information Processing Society - NAFIPS (Cat. No.99TH8397), New York, NY, USA, 1999, pp. 760-765, doi: 10.1109/NAFIPS.1999.781796.

[2] C. C. Lee, "Fuzzy logic in control systems: fuzzy logic controller. I," in IEEE Transactions on Systems, Man, and Cybernetics, vol. 20, no. 2, pp. 404-418, March-April 1990, doi: 10.1109/21.52551.

[3] S. D. Prof. Pradip Kumar Saha Shubhajit Pal, Prof. Goutam Kumar Panda, "Comparative Study between Conventional PID and Fuzzy Logic Controller for a Current Controlled D.C. Drive Using MATLAB/Simulink," International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 3, no. 4, pp. 9245-9252, 4/2014. Ess & Ess Research Publications.

[4] J. Y. M. Cheung and A. S. Kamal, "Fuzzy logic control of

refrigerant flow," UKACC International Conference on Control '96 (Conf. Publ. No. 427)، Exeter, UK, 1996, pp. 2-5 vol.1, doi: 10.1049/cp:19960538.

[5] A. K. Gaurav, "Comparison between Conventional PID and Fuzzy Logic Controller for Liquid Flow Control: Performance Evaluation of Fuzzy Logic and PID Controller by Using MATLAB/Simulink " International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 1, no. 1, June 2012.

[6] E. Pathmanathan and R. Ibrahim, "Development and implementation of Fuzzy Logic Controller for Flow Control Application," 2010 International Conference on Intelligent and Advanced Systems, Kuala Lumpur, Malaysia, 2010, pp. 1-6, doi: 10.1109/ICIAS.2010.5716249.

[7] S.R.Vaishnav and Z.J.Khan. "Design and Performance of PID and Fuzzy Logic Controller with Smaller Rule Set for Higher Order System.", Proceedings of the World Congress on Engineering and Computer Science San Francisco, USA, October 24-26, 2007

[8] O. S. Isizoh A. , Anazia A., " Temperature Control System Using Fuzzy Logic Technique," (IJARAI) International Journal of Advanced Research in Artificial Intelligence, vol. 1, p. 3, 2012.

[9] A. B. Bikas Mondal, Bhaskar, and Rahit Saha, " Performance Comparison of Conventional PID and Fuzzy Logic Controller in the Field of over headed Water Level Control System," International Journal of Computer Sciences and Engineering, Joun 2016.

[10] A. K. M. Younis S. Dawood, and Muhammed A. Ibrahim, " Comparison of PID, GA and Fuzzy Logic Controllers for Cruise Control System ISSN " International Journal of Computing and Digital Systems, p. 5, Sep-2018.

[11] T. M. W. V. Singhe Wijesinghe, and H.D.N.S. Priyankara, " Fuzzy Logic Controller vs PID Controller for Real Time Magnetic Levitation System " IEEE, 2018.

[12] M. B. Devendra Somwanshia , Gaurav Kumarc , Gajal Parashar " Comparison of Fuzzy-PID and PID Controller for Speed Control of DC Motor using labVIEW " International Conference on Pervasive Computing Advances and Applications, 2019.

[13] M. T. Ahmed, " Analysis And Design Of A Fuzzy Controller And Performance Comparison Between The PID Controller And Fuzzy Controller " INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH, vol. 9, no. 10, October2020.

[14] F. A. OMAR, "performance comparison of PID and Fuzzy Logic Controller for water level control with appling time delay " Konya Journal of Engineering Sciences, vol. 9, p. 4, 2021.

[15] S. a. Vijayaraghavan, " Fuzzy Logic Controller Design And Simulation For Industrial Application "International Journal of Creative Research Thoughts (IJCRT), 2022.

[16] H. A. R. Muhammad Akmal Ridzuan1, "Real Time Comparison between PID and Fuzzy Logic Controller for DC Motor Speed Control," EVOLUTION IN ELECTRICAL AND ELECTRONIC ENGINEERING, vol. 5No. 1, April 2024.

[17] Agus Mulya Putra a, *, H. M. a. , 2, and R. A. R. , "Comparison of Proportional Integral Derivative and Fuzzy Logic Controllers: A Literature Review on the Best Method for Controlling Direct Current Motor Speed," International Journal of Robotics and Control Systems, vol. Vol. 5, No. 1, December , 2024.