

Article

A Comparative Study on Fuzzy Logic-Based Liquid Level Control Systems with Integrated Industrial Communication Technology

Hasan Mhd Nazha ^{1,*}, Ali Mahmoud Youssef ², Mohamad Ayham Darwich ^{2,3}, Their Ahmad Ibrahim ² and Hala Essa Homsieh ²

¹ Faculty of Mechanical Engineering, Otto Von Guericke University Magdeburg, Universitätsplatz 2, 39106 Magdeburg, Germany

² Faculty of Technical Engineering, University of Tartous, Tartous P.O. Box 2147, Syria; alimyousef@tartous-univ.edu.sy (A.M.Y.); a.darwich@au.edu.sy (M.A.D.); theiribrahim@tartous-univ.edu.sy (T.A.I.); halahoumsia@tartous-univ.edu.sy (H.E.H.)

³ Faculty of Biomedical Engineering, Al-Andalus University for Medical Sciences, Tartous P.O. Box 101, Syria

* Correspondence: hasan.nazha@ovgu.de

Abstract: This study presents an advanced control system for liquid level regulation, comparing a traditional proportional-integral-derivative (PID) controller with a fuzzy logic controller. The system integrates a real-time monitoring and control interface, allowing flexible adjustments for research and training applications. Unlike the PID controller, which relies on predefined tuning parameters, the fuzzy logic controller dynamically adjusts control actions based on system behavior, making it more suitable for processes with non-linear dynamics. The experimental results highlight the superior performance of the fuzzy logic controller over the PID controller. Specifically, the fuzzy logic controller achieved a 21% reduction in maximum overshoot, a 62% decrease in peak time, and an 83% reduction in settling time. These improvements demonstrate its ability to handle process fluctuations more efficiently and respond rapidly to changes in liquid levels. By offering enhanced stability and adaptability, the fuzzy logic controller presents a viable alternative for liquid level control applications. Furthermore, this research contributes to the development of flexible and high-performance control solutions that can be implemented in both industrial and educational settings. The proposed system serves as a cost-effective platform for hands-on learning in control system design, reinforcing contemporary engineering education and advancing intelligent control strategies for industrial automation.

Keywords: liquid level control; fuzzy logic controller; PID controller performance



Received: 2 January 2025

Revised: 4 February 2025

Accepted: 8 February 2025

Published: 2 March 2025

Citation: Nazha, H.M.; Youssef, A.M.; Darwich, M.A.; Ibrahim, T.A.; Homsieh, H.E. A Comparative Study on Fuzzy Logic-Based Liquid Level Control Systems with Integrated Industrial Communication Technology. *Computation* **2025**, *13*, 60. <https://doi.org/10.3390/computation13030060>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Level control systems are crucial in various industries, including petroleum, chemical processing, water treatment, and steel manufacturing, where precise regulation of liquid levels directly impacts process stability, safety, and efficiency. Effective level control also influences interconnected variables such as flow and pressure, necessitating the use of reliable control strategies, particularly in complex and non-linear environments [1]. While proportional-integral-derivative (PID) controllers remain the dominant choice in industrial applications due to their simplicity and cost-effectiveness, their performance can be suboptimal in handling non-linear, time-varying, and multi-variable processes [2]. To address these challenges, researchers have explored advanced control techniques that incorporate artificial intelligence (AI) and machine learning (ML) to enhance system performance [3].

Recent advancements in industrial automation have led to the development of various intelligent control methodologies that extend beyond conventional PID control. Model

Predictive Control (MPC), for example, has gained traction for its ability to handle multi-variable processes with constraints, improving stability in complex environments [4]. Additionally, adaptive control techniques such as gain scheduling and self-tuning regulators have been implemented to enhance PID performance in non-linear systems [5]. Reinforcement Learning (RL)-based controllers have also emerged as promising solutions, allowing real-time adaptation to dynamic process changes without requiring an explicit system model [6]. Fuzzy logic controllers (FLCs) have demonstrated significant potential in addressing the limitations of traditional PID control. Unlike PID controllers that rely on precise mathematical models, FLCs use linguistic rules and heuristic reasoning to provide robust control in non-linear and uncertain environments [7]. Studies have shown that integrating fuzzy logic with other intelligent control strategies, such as neural networks and genetic algorithms, further enhances adaptability and control accuracy [8]. For instance, a recent study by Ayyagari et al. (2019) demonstrated a hybrid fuzzy–PID approach that reduced overshoot and settling time in complex liquid level control applications [9].

Despite these advancements, there remains a gap in the practical implementation of adaptive control systems tailored for highly non-linear single-input/single-output (SISO) and multi-tank processes. This study aims to bridge that gap by developing an AI-based control system—specifically, an FLC—designed to improve the response characteristics of liquid level control. Unlike traditional PID controllers, the FLC approach dynamically adjusts control parameters, offering superior performance in mitigating overshoot, reducing peak time, and enhancing stability. The study experimentally evaluates the PID performance on a Level Process Station 3503, a system designed for training and industrial simulation, allowing for precise replication of real-world operational conditions.

Additionally, this research extends beyond simulation by constructing a laboratory model as an alternative to the Foxboro 762 cna local controller (Foxboro 762 cna, Schneider Electric Company, Andover, MA, USA). By providing an educational and experimental platform, this study equips students and researchers with hands-on experience in modern control methodologies. Ultimately, this research not only validates the effectiveness of FLC over traditional PID approaches but also contributes to the ongoing advancements in intelligent control strategies for industrial applications. Future studies can expand upon these findings by exploring hybrid AI-based controllers to further enhance adaptability and efficiency in non-linear process control environments.

2. Materials and Methods

2.1. Control System Implementation

We began by studying the components of the liquid level processing platform “3503”, followed by simulating the system using mathematical equations in MATLAB’s Simulink library (MATLAB R2023b, Version 9.14.0, The MathWorks Inc., Natick, MA, USA). We analyzed the system’s response under both a traditional PID controller and an artificial intelligence controller (fuzzy logic controller) designed with MATLAB’s Fuzzy Logic Toolkit. Next, we proceeded with the practical implementation of the proposed control system model, utilizing WinCC Flexible (2008 SP5), Step7-Microwin (V4 SP9), PC-Access OPC (V1.0 SP2), and E-Plan (V 2.4) software. Hardware components included Siemens’s products such as the PLC-S7-200 with an EM-235 analog expansion and a PPI/USB communication cable, recognized for their reliability in control and automation.

The Level Process Station 3503 is designed for control systems in processes for practical training in the measurement and control of various industrial operations [10]. It consists of the following main components, as shown in Figure 1.

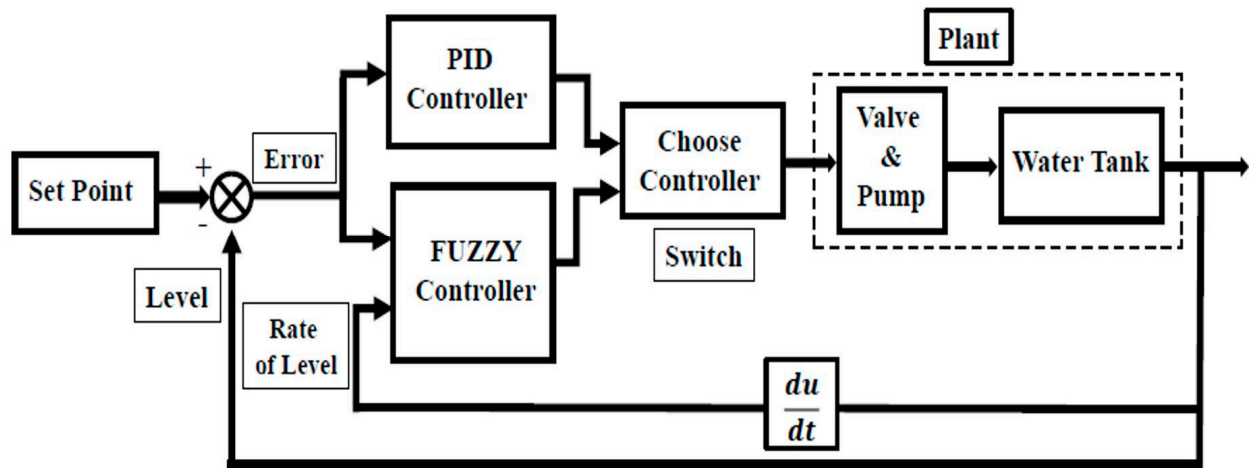


Figure 1. Block diagram of the system.

The Foxboro 762 CNA controller was selected for its dual-loop proportional-integral-derivative (PID) control functionality, which enabled precise regulation of interdependent process variables such as liquid level and flow rate. Its microprocessor-based architecture supports flexible configurations, including operation as two independent control loops or as a single-station cascade/auto-selector system, a feature critical for accommodating dynamic experimental conditions. The controller's enhanced capabilities, such as its dual auto/manual transfer station and dual 3-variable indicator, provide operational flexibility by allowing seamless transitions between automated and manual control modes during calibration and testing. The front-panel interface incorporates a fluorescent bar graph display for real-time visualization of control variables (e.g., pressure, temperature) and a numeric readout paired with an integrated keypad, enabling direct parameter adjustments, adaptive PID tuning, and alarm threshold configuration. The controller processes analog input signals (4–20 mA, 0–10 V) and generates corresponding analog outputs (4–20 mA) for actuator control, ensuring alignment with the precision requirements of the experimental setup [11].

Liquid level measurement was achieved using a Foxboro IDP10 I/A differential pressure transmitter, a two-wire device that measures hydrostatic pressure across a calibrated span of 0–500 kPa (0–92 cm level range) with an accuracy of $\pm 0.1\%$ of span. The transmitter integrates a silicon piezoresistive sensor with a stainless-steel isolation diaphragm, ensuring stable performance across a temperature range of $-40\text{ }^{\circ}\text{C}$ to $85\text{ }^{\circ}\text{C}$. Its HART protocol compatibility facilitates remote diagnostics and configuration, eliminating the need for external signal conditioning while providing a standardized 4–20 mA output proportional to the measured pressure [12].

To actuate the flow control valve (CV-1), an I/P (current-to-pressure) converter translated the controller's 4–20 mA output signal into a proportional pneumatic pressure signal (3–15 psi/20–102 kPa), modulating the valve position to regulate liquid flow and maintain the process variable within operational limits [13].

A centrifugal pump (24 VDC, constant speed) transferred water from the lower to the upper tank, with operation initiated via a front-panel switch [14]. The flow control valve (CV-1), serving as the final control element, adjusted the flow rate dynamically based on the controller's output signal to ensure precise level regulation [15].

2.2. Fuzzy Logic Controller Design

We used fuzzy logic for liquid level control in the tank because it can be described linguistically, and fuzzy controllers are effective in handling non-linear systems, excelling

in such applications [16]. Thus, a fuzzy logic controller (FLC) was chosen as an alternative to the PID controller for liquid level control. The Mamdani inference methodology was used in designing the inference rules for our controller, as it is the most commonly used fuzzy methodology, providing a specific output decision that improves system stability, increases effectiveness, and offers flexibility to varying system input values [17]. Two inputs were defined for the fuzzy controller: the liquid level error (referred to as Level) and the rate of change in the liquid level (referred to as Rate), with one output being the control valve (Valve). The fuzzy logic controller was designed using the Fuzzy Logic Toolkit in the Matlab environment.

After conducting iterative simulations and analyzing system responses, five membership functions for the liquid level error (first input variable) were defined within the normalized range of $[-1, 1]$: Small, Positive, High-Positive, Negative, and High-Negative. This configuration emerged from an empirical refinement process. Initial tests with three membership functions (Negative, Small, Positive) resulted in oscillations and sluggish convergence due to insufficient granularity in distinguishing error magnitudes. Expanding to five functions enabled finer differentiation between small, moderate, and large deviations, improving the controller’s ability to apply context-sensitive corrections (e.g., aggressive adjustments for High-Negative or High-Positive errors and subtle refinements near the setpoint). The normalized range and partitioning were informed by observed system dynamics, operational constraints, and iterative validation against step and disturbance responses, as shown in Table 1.

Table 1. Fuzzy ranges of the first input variable of the fuzzy controller.

Linguistic Expression	Liquid Level Error Range
(HN) High-Negative	$[-1 \ -1 \ -0.8 \ -0.4]$
(N) Negative	$[-0.8 \ -0.4 \ 0]$
(S) Small	$[-0.2 \ 0 \ 0.2]$
(P) Positive	$[0 \ 0.4 \ 0.8]$
(HP) High-Positive	$[0.4 \ 0.8 \ 1 \ 1]$

Similarly, after conducting tests and analyzing the response, three membership functions for the second input variable (rate of level change) were defined within the range of $[-0.1, 0.1]$: Small, Positive, and Negative. These ranges were selected based on experience and understanding of the operational mechanism, as shown in Table 2.

Table 2. Fuzzy ranges of the second input variable of the fuzzy controller.

Linguistic Expression	Error Rate Range
(N) Negative	$[-0.1 \ -0.1 \ -0.08 \ 0]$
(S) Small	$[-0.08 \ 0 \ 0.08]$
(P) Positive	$[0 \ 0.08 \ 0.1 \ 0.1]$

In the same manner, after testing the controller’s response, five membership functions for the output variable (control valve CV-1) were defined within the range of $[-1, 1]$: Close-Fast, Close-Slow, No-Change, Open-Low, and Open-Fast, as shown in Table 3.

Table 3. Fuzzy ranges of the output variable of the fuzzy controller.

Linguistic Expression	Output Variable Range
(CF) Close-Fast	[−1 −1 −0.84 −0.36]
(CL) Close-Low	[−0.84 −0.36 0]
(NC) No-Change	[−0.3653 −0.005291 0.3547]
(OL) Open-Low	[0 0.36 0.84]
(OF) Open-Fast	[0.12 0.84 1 1]

The differing ranges for the liquid level error ($[-1, 1]$) and rate of level change ($[-0.1, 0.1]$) stem from their distinct roles in the control logic and their empirical normalization. The level error range represents the normalized percentage deviation from the setpoint (e.g., $-1 =$ maximum allowable negative deviation, $+1 =$ maximum positive deviation). This normalization standardizes the input across operational scenarios. Conversely, the narrower rate range ($[-0.1, 0.1]$) was derived directly from observed system behavior. During testing, the rate of level change rarely exceeded ± 0.1 per unit time (normalized) under both transient and steady-state conditions. This tighter range ensures higher sensitivity to small rate fluctuations, enabling the controller to detect gradual drifts early. Crucially, scaling parameters map raw sensor data (e.g., level in meters, rate in m/s) to these normalized ranges, ensuring proportional significance. For instance, a small rate value (e.g., 0.05) within the $[-0.1, 0.1]$ range is amplified by its scaling gain to have a meaningful influence on the control signal, balancing resolution, and stability. The magnitudes of the ranges themselves do not dictate the output; rather, the relative scaling ensures both inputs contribute appropriately to the rule-based decisions.

The design of the fuzzy logic controller was based on the inference rules specified in the MATLAB/Fuzzy Rule Editor, using the Mamdani inference method, as previously mentioned. The continuous level variables and their rates were first mapped to five distinct membership functions, representing linguistic terms such as Very Low, Low, Medium, High, and Very High. This mapping process involved defining appropriate triangular membership functions to capture the dynamic range of the input variables accurately.

For any possible values of the fuzzy controller’s inputs, the outputs depend on these inference rules, which interpret the fuzzy sets derived from the continuous inputs. In the proposed design, the maximum number of inference rules is 15, as the total number of rules equals the product of the number of membership functions for the input variables. Figure 2 illustrates the designed inference rules in the MATLAB Rule Editor, reflecting how different combinations of membership functions influence the controller’s output.

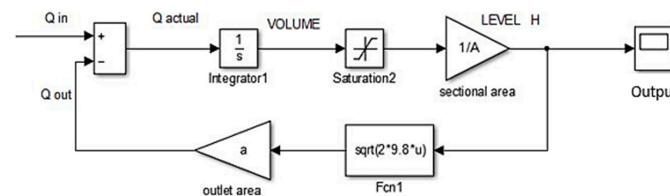


Figure 2. Illustration of the designed inference rules in the Matlab Rule Editor. The asterisk (*) denotes the algebraic product between the entities. The Saturation block limits the input signal within specified upper and lower bounds, ensuring that the output remains within the defined range.

The Matlab/Simulink toolbox was used to design a comprehensive model of the Level Processing Platform with a PID controller and an artificial intelligence controller (fuzzy logic controller) designed using the FIS Editor tool. Equation (1) describes the relationship between the level and the incoming and outgoing flow rates. For the level to remain

constant, the incoming flow rate must equal the outgoing flow rate. The level increases when the incoming flow rate is greater than the outgoing flow rate and decreases when the outgoing flow rate is greater than the incoming flow rate.

$$A \frac{dH}{dt} = Q_{in} - Q_{out} \quad (1)$$

where

A: Base area of the tank m^2 .

H: liquid level in the tank m.

Q_{in} : Incoming flow m^3/s .

Q_{out} : Outgoing flow m^3/s .

The resulting equation yields the following equation:

$$H(s) = \frac{[Q_{in}(s) - a\sqrt{2 \times g \times H(s)}]}{A \times S} \quad (2)$$

Equation (2) represents the Laplace-domain relationship between the liquid level and flow rate, adopted from the foundational model by Setiawan [16]. Figure 2 illustrates the corresponding Simulink implementation of this equation, depicting the functional block diagram of the tank system with incoming and outgoing flow rates. For brevity, detailed derivations of the governing equations are omitted here, as they align with standard liquid level control dynamics thoroughly characterized [16].

The pump in the studied system has a constant capacity and cannot be controlled in terms of speed; thus, the incoming flow rate provided by the pump is fixed when operational. According to the pump's technical specifications, the flow rate corresponding to the tank height is $0.00245 [m^3/s]$. However, the incoming flow rate can be controlled from a minimum to a maximum value by adjusting the control valve opening, which is connected in series with the pump. When the valve is fully closed, the incoming flow rate is zero, and when it is fully open, it allows the maximum flow rate to pass to the tank. Therefore, the pump will be represented by a constant, which is the maximum flow rate, and the control valve will be represented by an integrator with minimum and maximum value limits (0–1) and an initial valve opening value.

The "Fuzzy Controller with Rule viewer" block was used to represent the fuzzy logic controller in the Simulink environment. This block has two inputs: the first is the liquid level error (error), and the second is the rate of change of the liquid level (rate of level). It has a single output, which is the control valve opening, and serves as the input for the control valve. To integrate the system designed using the "FIS Editor" tool with the "Fuzzy Controller with Rule viewer" block, the designed system is saved and exported to the workspace, then opened and invoked in the properties window of the block. Figure 3 illustrates the final construction of the designed model after setting the PID controller parameters obtained through the Ziegler–Nichols tuning method.

We implemented the system practically, replacing the platform controller with the Siemens PLC-S7-200 and the analog expansion EM-235. The ladder logic language was used to design the programming code for the system in the "Step7-Microwin" environment, ensuring control for both the open-loop and closed-loop configurations of the Level Processing Platform (3503). The PID controller was designed using the "PID Wizard" and integrated into the closed-loop mode within a subroutine, which was then called within the main program using interrupts.

The proposed system requires a monitoring and data collection program to enhance the control system and achieve its intended purpose. To accomplish this, we used WinCC Flexible from Siemens due to its extensive capabilities and widespread use with many

programmable control systems and industrial applications. Interactive graphical user interfaces (HMI) were designed to allow selection of the control type, display and modification of PID controller parameters, and visualization of key parameters such as liquid level, valve opening, setpoint value, and actual value. Data are stored in an Excel file, and all response curves are displayed, as shown in Figures 4 and 5.

OPC technology is a Microsoft standard that enables industrial applications to exchange data on a single or multiple computers using a “Client-Server” architecture, overcoming compatibility issues between control system components [15]. This technology creates an intermediary interface independent of the connected device type, allowing users to integrate various hardware and software through an appropriate OPC program [16]. Figure 6 depicts the designed and implemented laboratory model.

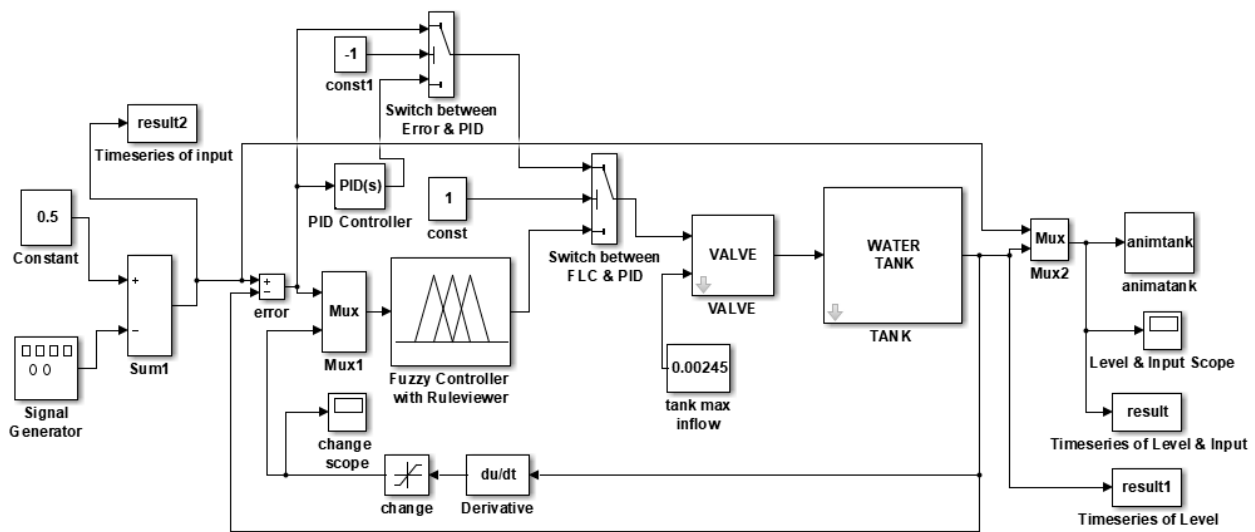


Figure 3. Final construction of the designed model MATLAB. The grey arrows indicate physical connections in Simulink’s Simscape environment, representing the flow of fluid between the valve and tank block. The Fuzzy Controller uses fuzzy logic to make decisions based on input variables. The Rule Viewer provides a visual representation of the applied fuzzy rules, showing how different input conditions influence the output by mapping inputs to their corresponding fuzzy sets and applying the defined rules.

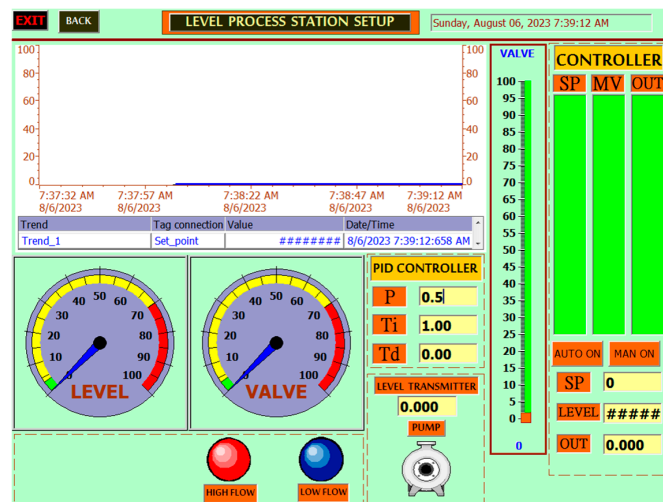


Figure 4. Interactive graphical user interface (HMI) of the monitoring and data collection system, developed using WinCC Flexible. The interface displays the control type selection, PID controller parameters, and real-time visualization of key parameters such as liquid level, valve opening, setpoint, and actual values.

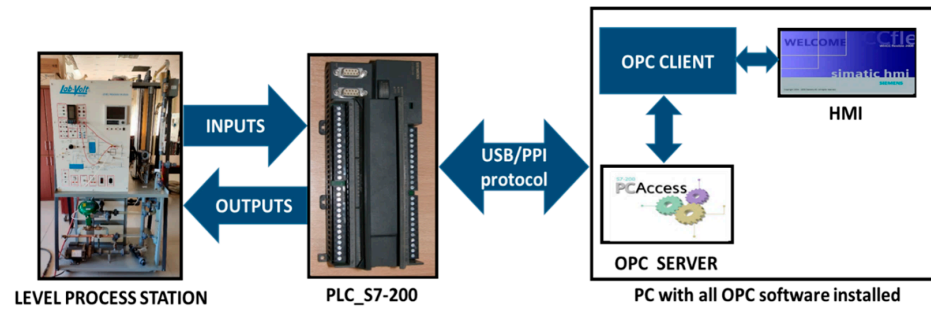


Figure 5. Schematic diagram of the proposed control system, illustrating key components such as sensors, control loops, and data acquisition interface. The colors in the schematic diagram represent different components of the control system: sensors are shown in blue, control loops in red, and the data acquisition interface in black.

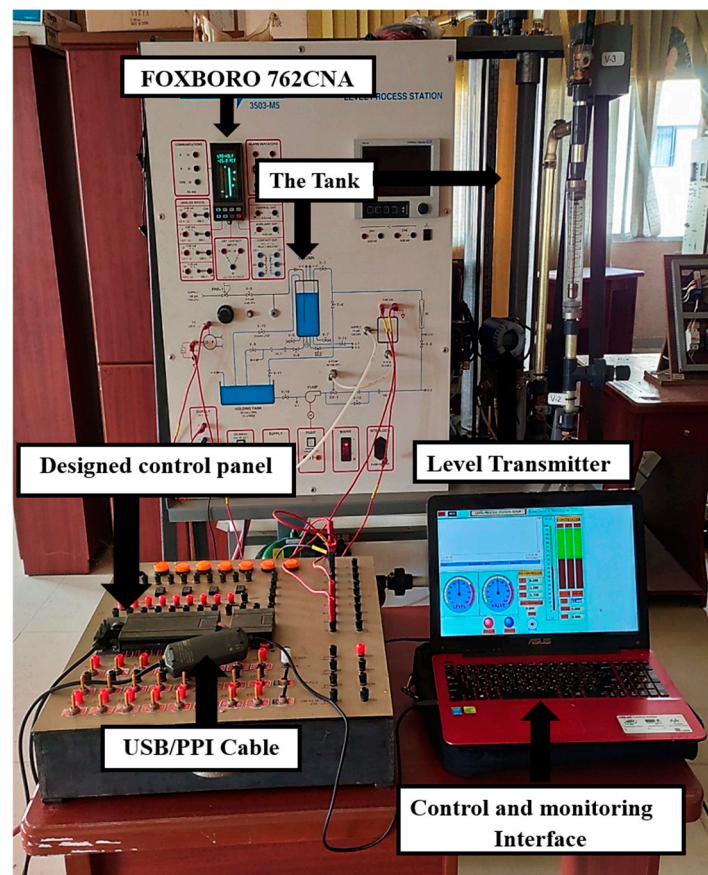


Figure 6. Designed and implemented models.

3. Results

The model shown in Figure 7 was extensively tested using both the fuzzy logic controller (FLC) and the PID controller to evaluate performance under identical conditions. The PID controller parameters were determined using the Ziegler–Nichols tuning method, with values set at $K_p = 0.48$, $K_i = 0.021$, and $K_d = 4.408$. Two reference levels were employed during testing, the first at 0.7 and the second at 0.3, alternating at a constant frequency of 0.003 Hz, in line with the technical specifications of the studied system. These specifications are detailed in Table 4.

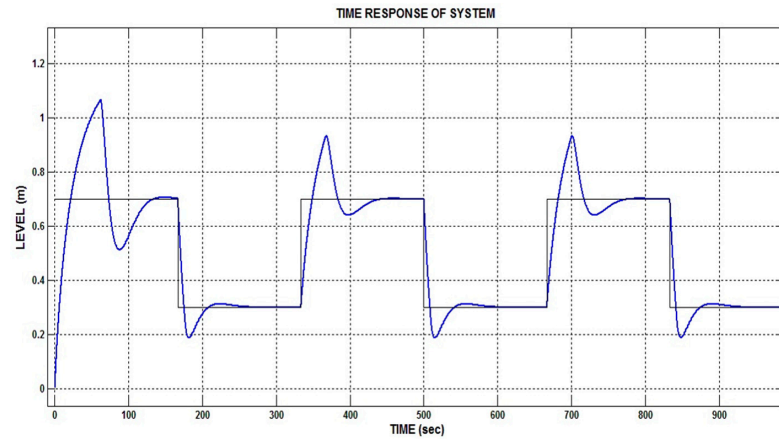


Figure 7. System response curve for both the fuzzy logic controller (FLC) and PID controller, tested under identical conditions. The PID controller parameters were determined using the Ziegler–Nichols tuning method with $K_p = 0.48$, $K_i = 0.021$, and $K_d = 4.408$. Two reference levels, 0.7 and 0.3, were alternated during testing to evaluate system performance.

Table 4. Technical specifications of the studied system.

Tank Height (H)	Maximum Pump Flow Rate	Tank Base Area (A)	Drain Outlet Area (a)
0.92 m	0.00245 m ³ /s	0.0324 m ²	0.0005 m ²

Figure 8 illustrates the system’s response curve under the control of the PID controller, highlighting its performance during transitions between the reference levels. The frequency of 0.003 Hz was chosen to allow sufficient time for the system to respond to changes and for the control action to stabilize during each test cycle. The results revealed that the PID controller provided adequate control but exhibited notable limitations, particularly during rapid fluctuations or when dealing with non-linear dynamics. These performance characteristics are influenced by the Ziegler–Nichols tuning method, which, while widely used, may not always yield optimal parameters for highly dynamic systems. The observed overshoot and undershoot reflect inherent challenges in PID control for such applications.

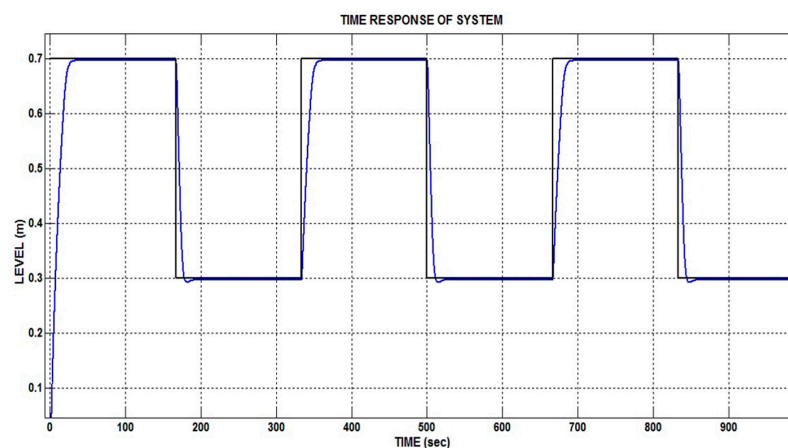


Figure 8. System response curve showing the performance of the fuzzy logic controller (FLC). The curve illustrates how the FLC manages the system’s behavior in response to varying inputs, highlighting its ability to adjust control parameters and maintain stability across different operating conditions.

Figure 8 shows the system response curve when using the fuzzy logic controller. Table 5 presents the quality indicators for the system response curves when using the PID controller and the fuzzy logic controller.

Table 5. Quality indicators for the system response curves when using the PID controller and the fuzzy logic controller.

Steady-State Error%	Time to Reach Peak Magnitude (s)	Settling Time (s)	Maximum Overshoot%	The Used Controller
None	None	35	None	Fuzzy logic controller
None	62	192	23.4	PID Ziegler–Nichols

In contrast, the fuzzy logic controller was tested using the MATLAB Fuzzy Logic Toolkit, leveraging the Rule Viewer tool to analyze system behavior in real-time. Figure 9 depicts the FLC’s response during testing, showcasing the error value, error rate of change, and valve opening under two critical conditions: the initial state and the steady state when the reference value was achieved. The FLC demonstrated superior adaptability to changes in the system, dynamically adjusting control actions based on performance metrics rather than fixed set points.

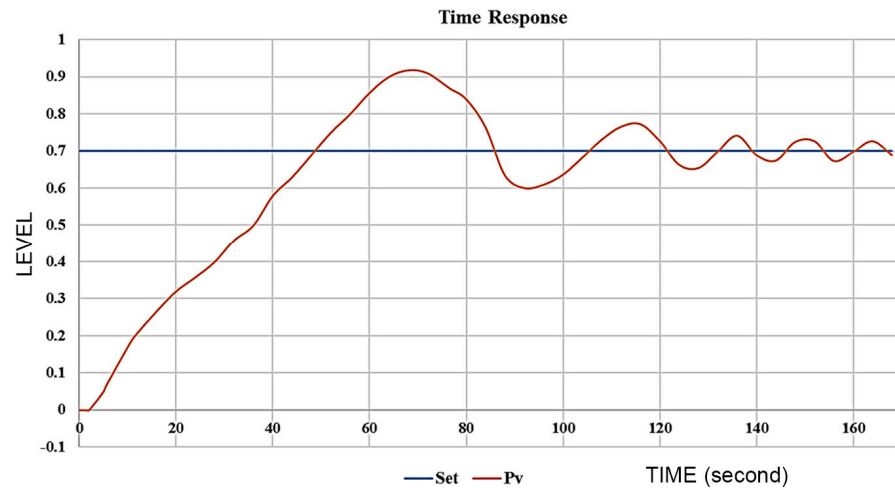


Figure 9. System response using the local platform controller.

We ran the system with a reference level of 0.7 m, which matches the value used in software testing. We monitored the system’s response using both the local controller and the designed experimental model with the same constants used in the software tests. We also varied the discharge valve opening to observe how the implemented model responded to level changes while monitoring the transmitter’s level currents and output module currents. Figure 9 shows the system response using the local platform controller. Table 6 compares the output currents of the local platform controller and the designed model controller.

Table 6. Output currents of the local platform controller and the designed model controller.

Condition	Designed Model		Foxboro	
	% Age Opening of Valve	Tank Level%	% Age Opening of Valve	Tank Level%
When achieving the desired value (0.7 m)	76.25%	83.75%	76.25%	83.75%
When achieving a new desired value (0.5 m)	57.6%	71.25%	57.6%	71.25%
When the drain valve is opened	20.62%	100%	20.62%	100%
When achieving the desired value after opening the drain valve	56.87%	84.37%	56.87%	84.37%

4. Discussion

The results obtained from the comparative testing of the fuzzy logic controller (FLC) and the PID controller highlight significant advancements in control system performance, emphasizing the superiority of the FLC for the studied liquid level process. The use of the Ziegler–Nichols method for tuning the PID controller provided a standardized basis for comparison, ensuring the validity of the observed differences in system response. This methodology, widely recognized in control engineering, was employed to determine the PID parameters ($K_p = 0.48$, $K_i = 0.021$, and $K_d = 4.408$) used in this study, ensuring a fair and structured evaluation of both control strategies.

Despite its widespread use and established reliability in industrial applications, the PID controller exhibited limitations when responding to rapid changes in reference levels (e.g., from 0.7 m to 0.3 m at 0.003 Hz). The response curves (Figure 7) indicate considerable overshoot and longer settling times, particularly under non-linear process conditions. These performance characteristics are influenced by the inherent linear nature of PID controllers and the constraints of the Ziegler–Nichols tuning method, which, while effective for many applications, may not optimize performance in highly dynamic or non-linear systems.

In contrast, the fuzzy logic controller demonstrated significantly improved performance, with reduced overshoot, peak time, and settling time (Figure 8). The FLC dynamically adjusted control actions based on real-time system conditions, resulting in smoother responses and faster stabilization. As shown in Table 2, the FLC achieved a 21% reduction in overshoot, a 62% decrease in peak time, and an 83% reduction in settling time compared to the PID controller. These improvements underscore the FLC's capability to manage complex system dynamics with greater flexibility, adaptability, and precision.

As shown in Table 2, the quality indicators reveal a marked improvement in system behavior with the FLC. The reduction in overshoot and settling time underscores the controller's ability to adapt to fluctuating inputs and minimize deviations. This aligns with previous studies that highlight the capability of fuzzy logic to handle uncertainty and non-linearity in control systems, often outperforming traditional methods in scenarios with dynamic and unpredictable disturbances. The differences in performance also suggest that while Ziegler–Nichols tuning provides a structured approach, alternative or adaptive tuning methods might improve PID controller behavior in complex environments.

The practical testing phase further corroborated the findings from software modeling. The designed experimental model exhibited superior performance compared to the Foxboro 762 cna controller, as illustrated in Figure 9. This reinforces the importance of transitioning from legacy systems to more adaptable solutions like the FLC. The local controller struggled to accommodate rapid changes in discharge valve openings, leading to delayed responses and suboptimal stability. Conversely, the designed model, utilizing the FLC, swiftly adjusted to changes in discharge conditions, maintaining level stability and achieving a smoother response.

These findings are in harmony with prior research advocating for the integration of fuzzy logic in control systems, which has demonstrated similar improvements in overshoot and settling time when applied to non-linear control applications [16,17]. However, discrepancies arise when comparing this study with other research employing advanced tuning algorithms for PID controllers, such as genetic algorithms or adaptive tuning methods. In some cases, these enhanced PID techniques have been reported to approach the performance of fuzzy logic controllers, particularly in systems with less pronounced non-linearity. The observed conflict may stem from differences in system complexity, the nature of disturbances, or the technical limitations of the Ziegler–Nichols tuning method used in this study.

The integration of fuzzy logic, as validated by the Rule Viewer tool, emphasizes the controller's capacity for real-time decision-making. By continuously evaluating error values and their rate of change, the FLC adjusts valve openings dynamically, avoiding the rigid dependency on predefined parameters characteristic of PID controllers. This adaptability is crucial for modern control systems that require precision and robustness in variable environments.

5. Conclusions

This study highlights the significant advantages of leveraging artificial intelligence, particularly fuzzy logic controllers (FLCs), in enhancing liquid level control systems compared to traditional PID controllers. The FLC demonstrated superior performance by significantly reducing maximum overshoot, peak time, and settling time, thereby showcasing its effectiveness in managing dynamic and non-linear processes where traditional PID controllers often struggle. Despite these promising results, the practical implementation of the fuzzy controller was not achieved due to the unavailability of the Fuzzy Control++ programming environment, which limited the study to theoretical analysis and simulation-based validation. Future research should address this gap through experimental validation to provide a direct performance comparison with PID controllers under real-world conditions.

The integration of OPC (OLE for Process Control) technology further enhanced the system's functionality by enabling seamless communication and real-time data exchange between different control components. This interoperability is critical for modern industrial automation, where flexible and efficient data flow is essential for optimal performance. Additionally, the developed experimental model proved to be a cost-effective and practical alternative to proprietary controllers, reinforcing local expertise in industrial process control. Its design emphasizes not only operational efficiency but also sustainability by reducing dependency on expensive, vendor-specific solutions.

Beyond its industrial relevance, the model serves as a valuable educational tool, providing students and researchers with hands-on experience in programming, system integration, and control system design. This aligns with contemporary educational goals focused on experiential learning and equipping future engineers with practical skills in automation and control technologies.

Future research directions should explore the integration of advanced AI-based techniques, such as neural networks, to optimize controller performance further. These techniques could enhance adaptability, predictive capabilities, and overall system robustness. Moreover, expanding the model's application to various industrial control platforms would improve its scalability and practical relevance. This would involve tailoring control rules and parameters to meet the specific dynamics and requirements of different processes.

Addressing current limitations related to implementation and experimental validation will strengthen the case for AI-driven controllers in industrial automation. Collectively, these innovations support a forward-thinking approach to advancing both industrial automation and technical education, promoting sustainable development driven by local expertise and cutting-edge technologies.

Author Contributions: Conceptualization, methodology, data curation, software: M.A.D. and A.M.Y.; investigation, visualization, formal analysis, validation, writing—original draft: T.A.I. and H.E.H.; supervision, project administration, resources, writing—review and editing: H.M.N. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The raw data supporting the conclusions of this article will be made available by the authors on request.

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

References

1. Owa, K.O.; Sharma, S.K.; Sutton, R. Optimised Multivariable Nonlinear Predictive Control for Coupled Tank Applications. In Proceedings of the IET Conference on Control and Automation 2013: Uniting Problems and Solutions, Birmingham, UK, 4–5 June 2013; Institution of Engineering and Technology: Stevenage, UK, 2013. [\[CrossRef\]](#)
2. Roland, B. PID Parameter Tables after ITAE to Control Overshooting Systems Found with AI Algorithm. *Int. J. Electr. Comput. Eng. Res.* **2023**. [\[CrossRef\]](#)
3. Kumar, V.; Rana, K.P.S.; Kumar, J.; Mishra, P. Self-Tuned Robust Fractional Order Fuzzy PD Controller for Uncertain and Nonlinear Active Suspension System. *Neural Comput. Appl.* **2016**. [\[CrossRef\]](#)
4. Singh, H.; Verma, R. Comparative Analysis of PID and Fuzzy Logic Controllers in Industrial Applications. *Int. J. Control Autom. Syst.* **2022**, *20*, 547–559. [\[CrossRef\]](#)
5. Smith, T.; Garcia, L. Advances in Intelligent Control Systems: A Review of Fuzzy Logic and AI-Based Controllers. *Sensors* **2023**, *23*, 1789. [\[CrossRef\]](#)
6. Kumar, N.; Das, S. Implementation of Fuzzy Logic-Based Liquid Level Control in Chemical Processing Plants. *Chem. Eng. Res. Des.* **2024**, *186*, 312–328. [\[CrossRef\]](#)
7. Wong, P.; Tan, K. Design and Experimental Validation of AI-Based Controllers for Process Automation. *J. Process Eng.* **2023**, *150*, 99–115. [\[CrossRef\]](#)
8. Reddy, A.S.; Reddy, G.N. An Adaptive Fractional Order Controller Design: A Realization for Liquid Level Regulation in Liquid Level Plant. *Meas. Sens.* **2024**, *31*, 100977. [\[CrossRef\]](#)
9. Ayyagari, R.; Raol, J.R. *Control Systems: Classical, Modern, and AI-Based Approaches*, 1st ed.; Wiley: Hoboken, NJ, USA, 2019. [\[CrossRef\]](#)
10. Instrumentation Society of America. *Control Valve Technical Specifications*; ISA: Research Triangle Park, NC, USA, 2022.
11. Mohindru, P. Review on PID, Fuzzy, and Hybrid Fuzzy PID Controllers for Controlling Non-Linear Dynamic Behaviour of Chemical Plants. *Artif. Intell. Rev.* **2024**, *57*, 4. [\[CrossRef\]](#)
12. Mamdani, E.H.; Assilian, S. An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller. *Int. J. Hum. Comput. Stud.* **1999**, *51*, 135–147. [\[CrossRef\]](#)
13. Aung, T.; Maung, M.; Tun, H. Design and Development of Fuzzy Logic Controller for Liquid Flow Control. *Int. J. Electron. Comput. Sci. Eng.* **2013**, *3*, 208–218.
14. Zheng, Z.; Tu, X. Real-Time Communication between PC and S7-200 PLC Based on OPC. *Procedia Eng.* **2008**, *32*, 354–357.
15. Mahnke, W.; Leitner, S.-H.; Damm, M. *OPC Unified Architecture*; Springer: Berlin/Heidelberg, Germany, 2009; ISBN 9783540688983.
16. Jadmiko, S.W.; Suyitno, L.A.; Lukman, F.M.; Gibran, B.T.G. Aplikasi Kendali PID Pada Simulator Water Level Control Berbasis TK-Series Controller. *J. Teknol. Rekayasa* **2021**, *6*, 155. [\[CrossRef\]](#)
17. Aung, T.T.E.; Soe, Z.C. Myanmar Liquid Flow Control by Using Fuzzy Logic Controller. *Int. J. Trend Sci. Res. Dev.* **2018**, *2*, 2190–2193. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.