

IoT-Enabled Supervised Learning-Based Prediction Model for Smart Instrumentation Controllers in Signal Conditioning Systems

S. Prakash^{1,*}, B Kalaiselvi², M. Batumalay³

¹Department of EEE, Bharath Institute of Higher Education and Research, Chennai, India

²Department of ECE, Bharath Institute of Higher Education and Research, Chennai, India

³Faculty of Data Science and IT, INTI International University, 71800 Nilai, Negeri Sembilan, Malaysia

³Centre for Data Science and Sustainable Technologies, INTI International University, 71800 Nilai, N. Sembilan, Malaysia

(Received: April 05, 2025; Revised: June 30, 2025; Accepted: October 08, 2025; Available online: October 22, 2025)

Abstract

This study proposes an intelligent Machine Learning (ML)-based smart controller for industrial flow process systems to enhance accuracy, adaptability, and robustness compared to conventional Proportional–Integral–Derivative (PID) controllers. The main idea is to replace reactive PID tuning with a proactive data-driven control strategy capable of predicting deviations and adjusting process parameters in real time. The objective is to develop and evaluate supervised learning models that can replicate and improve PID performance using real-time operational data collected from a flow process station. The proposed system integrates Internet of Things (IoT) sensors and edge computing to continuously acquire and process flow rate, pressure, and valve position data for model training and testing within the WEKA platform. Four classifiers—Linear Regression, Multilayer Perceptron (MLP), Sequential Minimal Optimization Regression (SMOreg), and M5P model tree—were compared using Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Relative Absolute Error (RAE), and model-building time as key evaluation metrics. Experimental results demonstrated that the M5P pruned tree model achieved the best overall performance with an MSE of 0.0024, RMSE of 0.0577, and model-building time of only 0.03 seconds, outperforming Linear Regression (RMSE = 0.0028), MLP (RMSE = 0.026), and SMOreg (RMSE = 0.0279). The findings show that the M5P-based controller closely replicates PID behavior while offering superior predictive accuracy, faster computation, and self-adaptive learning capabilities. The novelty of this research lies in demonstrating that an IoT-enabled, data-driven smart controller can achieve real-time predictive control without requiring explicit mathematical models, thereby simplifying tuning complexities and paving the way for autonomous, scalable, and intelligent control systems in Industry 4.0 environments.

Keywords: IoT, Machine Learning, PID Controller, Flow Controller, RMSE, Linear Regression, Multilayer Perceptron, M5P, SMOreg, Process Innovation

1. Introduction

Industrial process control systems are essential for achieving precision, reliability, and operational efficiency across manufacturing, energy, and chemical industries. Traditionally, these systems rely on Proportional–Integral–Derivative (PID) controllers, which remain among the most widely used control strategies because of their simplicity and stability under linear and time-invariant conditions [1]. However, PID controllers are reactive in nature, responding to deviations only after they occur. Their effectiveness depends heavily on precise mathematical modeling and manual parameter tuning, which can be complex, time-consuming, and error-prone when dealing with nonlinear, time-varying, or multivariable systems [2], [3].

With the emergence of Industry 4.0, industrial automation has undergone a paradigm shift toward intelligent, adaptive, and data-driven control frameworks. The integration of the Internet of Things (IoT) into control architectures enables the continuous acquisition of real-time data from distributed sensors and actuators, allowing systems to monitor process variables dynamically and make informed control decisions [4]. When combined with Machine Learning (ML), IoT facilitates predictive and autonomous control mechanisms that can identify temporal patterns, forecast process

*Corresponding author: S. Prakash (prakash.eee@bharathuniv.ac.in)

DOI: <https://doi.org/10.47738/jads.v6i4.803>

This is an open access article under the CC-BY license (<https://creativecommons.org/licenses/by/4.0/>).

© Authors retain all copyrights

deviations, and optimize control actions before errors occur [5], [6]. This evolution from reactive to predictive control signifies a major advancement in process optimization and operational efficiency in modern industrial systems. Despite these developments, challenges remain in implementing intelligent control systems that are both computationally efficient and adaptable to diverse process conditions. Many previous studies have focused on model-based control or reinforcement learning approaches, which, although effective, often require extensive domain-specific modeling and significant computational resources, making them less practical for real-time applications [7], [8]. Furthermore, most existing ML-based controllers have limited integration with IoT frameworks, restricting their ability to leverage distributed sensor networks for adaptive decision-making and continuous learning [9].

In response to these limitations, the present study proposes an IoT-enabled supervised learning-based smart controller designed to replace the conventional PID controller in industrial flow process systems. The proposed system employs machine learning techniques to predict control actions proactively, thereby enhancing adaptability and accuracy without the need for explicit mathematical modeling. The development and implementation of the intelligent controller are carried out using the Weka platform, an open-source software environment that offers a wide range of data mining and machine learning algorithms. Weka, developed at the University of Waikato, New Zealand, has been successfully utilized in multiple domains such as cybersecurity for anomaly detection [10], predictive maintenance for early fault identification [11], and agriculture for crop yield prediction based on environmental and soil data [12]. Its capacity for data preprocessing, classification, regression, and model validation makes it a suitable tool for building intelligent control systems.

In this research, supervised learning algorithms including Linear Regression, Multilayer Perceptron, SMOreg (Sequential Minimal Optimization for Regression), and M5P model tree are trained and evaluated using real-time data from a conventional PID-controlled flow process. The comparative performance of these models is assessed based on Mean Squared Error (MSE), Root Mean Squared Error (RMSE), and model-building time to identify the most effective predictive model for implementation [13]. The integration of IoT-enabled data acquisition and supervised learning not only enhances control precision but also facilitates remote monitoring, predictive maintenance, and system scalability.

This work aims to demonstrate that an IoT-enabled, machine learning-based controller can replicate and surpass the performance of traditional PID systems by offering predictive control capabilities, reduced tuning complexity, and higher adaptability in dynamic environments. The study contributes to the advancement of intelligent instrumentation and control systems, aligning with the broader objectives of Industry 4.0 to achieve autonomous, data-driven industrial operations.

2. Literature Review

Research on intelligent control and machine learning in industrial systems has evolved toward more autonomous, adaptive, and data-driven approaches. Numerous studies have investigated the integration of the IoT, artificial intelligence, and advanced control algorithms to enhance system performance, predictive accuracy, and adaptability in dynamic environments.

Several investigations have demonstrated the effectiveness of data-driven control frameworks in improving automation and decision-making. Deep learning-based models have been employed to manage smart networks and optimize encrypted traffic flow in distributed environments [4]. Reinforcement learning approaches have been used for nonlinear system control through Q-Learning and Markov Decision Process algorithms, enabling controllers to learn from environmental feedback and improve long-term performance [5], [6]. Other studies have implemented supervised and ensemble learning methods such as Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Extreme Learning Machines (ELM) for predictive control and anomaly detection [7], [8], [9]. These approaches consistently indicate that machine learning, particularly supervised and hybrid models, provides high prediction accuracy and stability in complex, time-varying systems.

While reinforcement learning methods [5], [6] focus on self-adaptation and long-term optimization, they require large datasets and high computational resources, limiting real-time deployment in industrial settings. In contrast, supervised and ensemble learning models [7], [8], [9] are faster to train and implement but often lack online adaptability once deployed. Fuzzy-based approaches have been applied to energy management systems, combining fuzzy inference with

heuristic optimization such as the BAT algorithm to minimize energy consumption and cost [10]. However, these models rely heavily on rule-based structures and expert-defined parameters, which may limit scalability.

In wireless sensor networks, regression-based models such as M5P and SMOreg have demonstrated strong localization accuracy and computational efficiency [11]. Yet, they perform best under static conditions and may degrade in performance under non-stationary or noisy environments. Neural network-based models have been applied to IoT security systems using predictive temperature analysis to detect abnormal patterns [12], achieving high detection accuracy but facing challenges in interpretability and model transparency. Fractional-order control models have also been explored to enhance the dynamic response of industrial processes, showing improvements in rise time and settling time but introducing increased tuning complexity [1].

Despite substantial progress, several limitations persist in existing research. Reinforcement learning and deep neural network approaches often exhibit high model complexity and long training times, making them less suitable for low-latency, resource-constrained IoT environments [5], [8], [7]. Supervised and regression-based methods [7], [8], [9], though computationally efficient, are typically static and unable to adapt autonomously to new system states without retraining. Fuzzy systems and heuristic controllers [10] lack generalization when exposed to large-scale, heterogeneous data streams. Fractional-order and hybrid control models [1] provide enhanced responsiveness but require sophisticated tuning algorithms and are not easily implemented in embedded architectures.

Broader reviews on machine learning in manufacturing and industrial automation have also identified significant barriers to the practical deployment of intelligent controllers, including data heterogeneity, communication latency, and the absence of unified IoT standards [13], [14]. These reviews highlight that while predictive maintenance, anomaly detection, and process optimization are well-studied, few frameworks have successfully integrated IoT-based real-time sensing with adaptive machine learning control under industrial constraints.

Across the literature, there is a clear progression toward integrating IoT, machine learning, and intelligent control to achieve predictive and adaptive process management. Reinforcement and deep learning techniques provide adaptability but demand substantial computational power [5], [6], [7], while supervised and ensemble models offer precision and stability suitable for real-time industrial applications [7], [8], [9]. Fuzzy and heuristic optimization frameworks contribute interpretability and robustness in energy management [10], whereas regression-based and neural models enhance reliability in wireless networks and IoT security [11], [12]. Fractional-order controllers expand the control domain by improving transient response and robustness [1].

Synthesizing these insights reveals that an effective intelligent control system must balance accuracy, adaptability, and computational efficiency. The convergence of IoT-enabled data acquisition with supervised learning algorithms offers a viable pathway toward real-time, predictive, and autonomous control. Building upon this synthesis, the present study proposes an IoT-enabled supervised learning-based prediction model for smart instrumentation controllers in signal conditioning systems. This model leverages the predictive power of machine learning, the connectivity of IoT, and the adaptability of intelligent control to optimize process performance and reliability in dynamic industrial environments.

3. Methodology

3.1. System Configuration

The proposed smart controller was developed to replace the conventional Proportional–Integral–Derivative (PID) controller within a closed-loop flow control system [15], [16]. The structure of the control loop consists of a Setpoint Reference (SP), a summing junction, the proposed smart controller, a Final Control Element (FCE), a flow process station, and a measuring device. The instantaneous control error is defined as

$$e(t) = SP - y(t) \quad (1)$$

$e(t)$ represents the deviation between the setpoint (SP) and the measured process output ($y(t)$). The proposed controller generates a control signal $u(t)$, which is sent to the FCE that manipulates the process input. The overall system dynamics can be expressed as

$$y(t) = G_p(s) \cdot u(t) + d(t) \quad (2)$$

$G_p(s)$ denotes the process transfer function and $d(t)$ represents external disturbances affecting system stability.

The feedback mechanism ensures that any deviation from the desired setpoint is minimized in real time [17], [18]. The proposed configuration introduces an intelligent data-driven control model that learns optimal control actions from process behavior rather than relying on explicit analytical models. Figure 1 shows the closed-loop structure of the smart flow control system, consisting of the proposed intelligent controller, the final control element, the flow process station, and the feedback sensor loop.

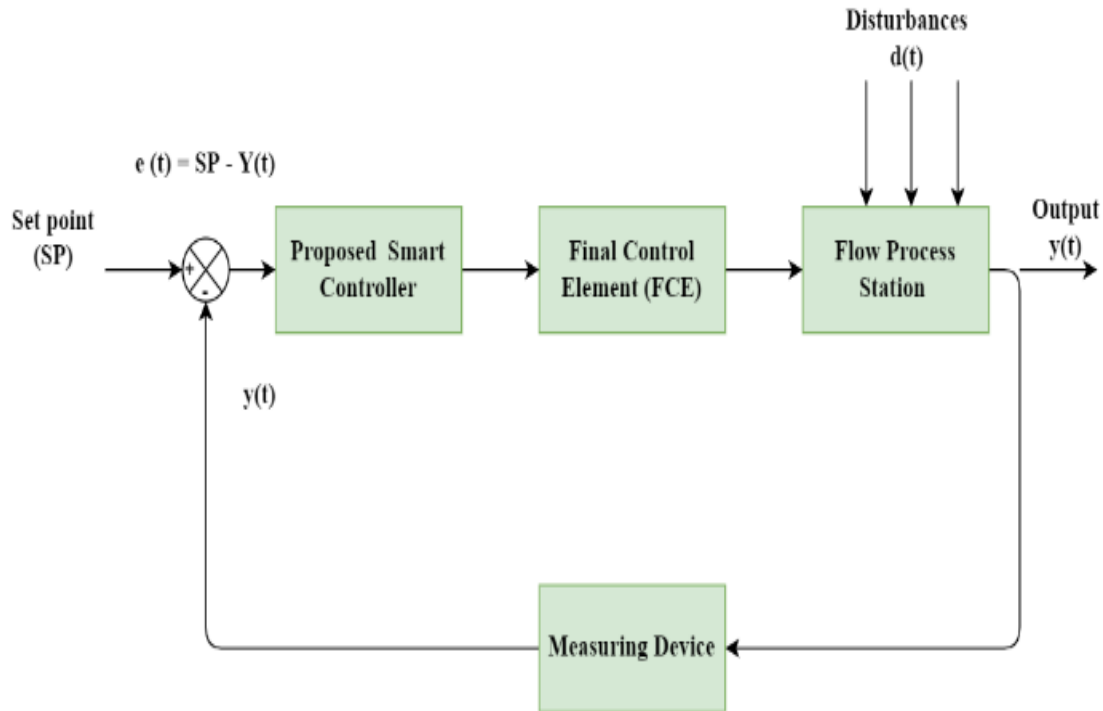


Figure 1. Proposed Block Diagram with Flow Controller

3.2. Data Acquisition and Preprocessing

The experiment was conducted on a real-time flow process station operating under a conventional PID controller. The setpoint was maintained at 250 LPH, corresponding to a transmitter output of 2.5 V and approximately 30% valve opening. At steady state, 1,332 instances of flow, voltage, and control valve positions were recorded as training data for the machine learning algorithms. The parameters measured include time (t), process variable in liters per hour (PV_{LPH}), transmitter voltage (PV_V), control percentage (CP1), and error (E). These parameters form the input–output dataset represented as [19].

$$\begin{aligned} X &= [t, PV_{LPH}, PV_V, CP1, E] \\ Y &= [u(t)] \end{aligned} \quad (3)$$

X represents the input features and Y denotes the target output variable. Table 1 presents a subset of the real-time data collected from the PID-controlled system.

Table 1. Real-time data of the flow process station

Time (s)	PV (LPH)	PV (Volts)	CP1 (%)	Error (%)
49.27	103.91	1.04	88.01	29.22
49.41	115.26	1.15	86.03	26.95
155.20	256.65	2.57	31.67	-1.33
191.03	254.46	2.54	30.81	-0.89
213.22	253.36	2.53	30.43	-0.67

213.63	252.99	2.53	30.47	-0.60
--------	--------	------	-------	-------

This dataset forms the foundation for the supervised learning framework that drives the proposed smart controller.

3.3. Mathematical Modeling of the Conventional PID Controller

For comparison purposes, the conventional PID controller is described by the following expression:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \quad (4)$$

K_p , K_i , and K_d are the proportional, integral, and derivative gains, respectively. These parameters determine the response characteristics of the controller, including rise time, settling time, and steady-state error. Although PID controllers are reliable, they rely heavily on manual tuning and perform poorly in nonlinear systems [20]. This limitation motivates the adoption of a machine learning-based model that can approximate nonlinearities adaptively without requiring manual gain adjustment.

3.4. Supervised Learning Framework

The proposed controller employs a supervised learning approach in which the model learns from pre-recorded input–output data pairs. The objective of the learning algorithm is to find a function $f_\theta(X)$ that maps inputs X to outputs Y by MSE between predicted and actual outputs. The optimization objective is expressed as

$$J(\theta) = \frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2 \quad (5)$$

n is the total number of data samples, Y_i is the actual output, $\hat{Y}_i$ is the model's prediction, and θ represents the set of trainable parameters. The learning process involves dataset partitioning into training and testing subsets [21]. The model undergoes multiple epochs of training, where its parameters are iteratively updated to minimize the cost function. The trained model is then validated on unseen data to assess generalization capability. Model evaluation is performed using three statistical metrics:

$$\begin{aligned} RMSE &= \sqrt{\frac{1}{n} \sum_{i=1}^n (Y_i - \hat{Y}_i)^2} \\ MAE &= \frac{1}{n} \sum_{i=1}^n |Y_i - \hat{Y}_i| \\ R^2 &= 1 - \frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{\sum_{i=1}^n (Y_i - \bar{Y})^2} \end{aligned} \quad (6)$$

RMSE represents the root mean squared error, MAE the mean absolute error, and R^2 the coefficient of determination. Figure 2 represents the conceptual workflow of the supervised learning process, illustrating data training, evaluation, model tuning, and deployment stages.

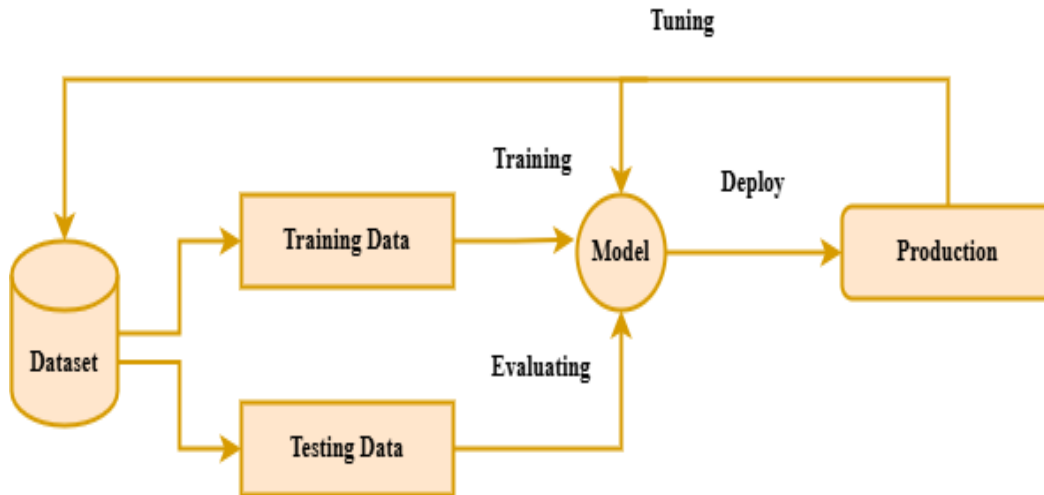


Figure 2. Supervised Learning Flow Block Diagram

3.5. Learning Algorithms

Four supervised learning models were tested and compared. The first model, linear regression, assumes a linear relationship between inputs and outputs as represented by

$$\hat{Y} = \beta_0 + \sum_{j=1}^m \beta_j X_j \quad (7)$$

β_0 is the intercept and β_j represents regression coefficients. The second model, the MLP, uses a nonlinear mapping function given by

$$\hat{Y} = f(W_2 \cdot \sigma(W_1 X + b_1) + b_2) \quad (8)$$

W_1 and W_2 are weight matrices, b_1 and b_2 are bias terms, and $\sigma(\cdot)$ is an activation function. The third model, (Sequential Minimal Optimization for Regression (SMOreg)), solves the regression version of the support vector machine using quadratic optimization:

$$\begin{aligned} \min_{\alpha} \quad & \frac{1}{2} \alpha^T Q \alpha - e^T \alpha \\ \text{subject to} \quad & 0 \leq \alpha_i \leq C \end{aligned} \quad (9)$$

Q is the kernel matrix and C is the regularization constant. The fourth model, M5P model tree, constructs a decision tree where each leaf node fits a linear regression function:

$$\hat{Y}_i = \beta_{0i} + \sum_{j=1}^m \beta_{ji} X_j \quad (10)$$

This approach combines the interpretability of decision trees with the precision of regression models, providing a balance between accuracy and computational efficiency.

3.6. Implementation Characteristics

Real-time implementation of the smart controller requires continuous sensor feedback and stable communication between IoT-enabled field devices and the central control algorithm. Sensor reliability is essential, as missing or corrupted data can disrupt prediction accuracy [22], [23]. When a fault is detected, the system temporarily restricts autonomous adjustments and alerts the operator for manual oversight. The adaptive control action generated by the machine learning model can be expressed as

$$u(t) = f_{\theta}(x_t) = \arg \min_{\theta} \mathbb{E}[(Y_t - \hat{Y}_t)^2] \quad (11)$$

This formulation ensures that the model continuously refines its predictions using real-time data, allowing the controller to self-optimize under changing process conditions. Compared to conventional PID control, which depends on static parameters K_p , K_i , and K_d , the proposed approach dynamically updates its internal parameters to maintain optimal control accuracy. Consequently, the controller exhibits predictive, adaptive, and self-tuning behavior suitable for nonlinear and time-varying industrial environments.

The methodology integrates experimental data collection, supervised learning model development, and mathematical control theory into a unified framework. By combining real-time IoT data acquisition with adaptive learning algorithms, the proposed smart controller achieves predictive and robust control performance. The mathematical formulations presented here establish a quantitative foundation for evaluating the system's behavior relative to traditional PID control approaches.

4. Results and Discussion

The dataset used in this study was collected from a conventional flow process station after the system reached a steady-state condition to ensure that the acquired data accurately represented stable operating behavior. The recorded data consisted of five primary attributes—time, process variable (flow rate in liters per hour), transmitter output voltage, control valve opening percentage, and percentage error. This dataset, comprising 1,332 instances, served as the foundation for training and testing the machine learning models used in developing the proposed intelligent controller.

The entire data analysis and model development process were conducted using the Waikato Environment for Knowledge Analysis (WEKA) software tool. WEKA is an open-source platform developed at the University of Waikato, New Zealand, designed to perform machine learning and data mining tasks, including data preprocessing, regression, classification, clustering, and visualization. In this study, WEKA was used primarily for supervised learning model training, validation, and optimization to identify the most accurate and computationally efficient model suitable for real-time control implementation.

4.1. Workflow of Model Development

The development workflow for obtaining the optimum model involved several stages, beginning with data collection, preprocessing, model training, and performance validation. The flowchart presented in [figure 3](#) illustrates the systematic process used to build and evaluate models within the WEKA environment.

The process starts with data collection from the flow process station, which is followed by preprocessing to remove noise and normalize the dataset. The cleaned data is then used to train multiple classifiers, including Linear Regression, Multilayer Perceptron, SMOreg, and M5P. Each classifier undergoes parameter tuning by varying learning rate, batch size, and kernel size until an optimal configuration is achieved. Model performance is continuously evaluated using RMSE and RAE as validation criteria. Once a model achieves the lowest RMSE and RAE values, it is identified as the optimum model and selected for deployment as the core of the smart controller.

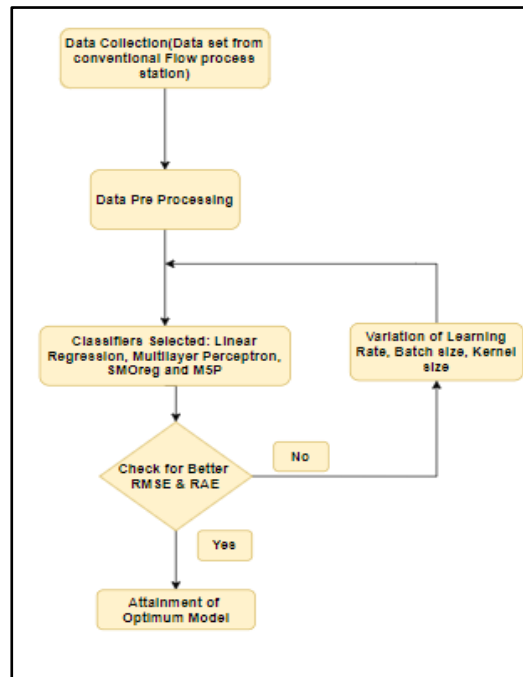


Figure 3. Experimental flowchart for model optimization using WEKA

4.2. Model Training and Validation

The dataset, in comma-separated format, was loaded into WEKA to perform numerical value training using the four selected classifiers. Each algorithm was trained and validated using a 10-fold cross-validation method to ensure that the models generalized well to unseen data. This method divides the dataset into ten equal partitions; nine are used for training, and one is used for testing, repeating the process ten times so that each portion is used for testing once.

During model configuration, specific hyperparameters were adjusted to improve performance. In the Multilayer Perceptron model, the learning rate was increased from its default value of 0.3 to 0.5 to accelerate convergence. For the SMOREg model, 347,978 kernel evaluations were executed with approximately 75% caching to balance speed and accuracy. The M5P algorithm, which constructs decision trees with linear regression functions at each terminal node, was implemented in its pruned form to prevent overfitting. The performance of each classifier was assessed based on three key indicators: RMSE, RAE, and model-building time. The results are summarized in [table 2](#).

Table 2. Experimental Results of the Developed Model

Category	Classifier	RMSE	Relative Absolute Error (%)	Time to Build Model (s)
Functions	Linear Regression	0.0028	0.0584	0.01
Functions	Multilayer Perceptron	0.026	0.6186	0.28
Functions	SMOREg	0.0279	0.6642	0.13
Trees	M5P	0.0024	0.0577	0.03

The Linear Regression and M5P classifiers achieved the lowest RMSE and RAE values, indicating higher prediction accuracy. The M5P model showed slightly better precision than Linear Regression while maintaining a short training time. In contrast, the Multilayer Perceptron and SMOREg classifiers required more time to build their models and produced higher error rates, making them less suitable for real-time control applications.

4.3. Model Building Time Analysis

The relationship between the four classifiers and the time required to build their respective models is presented in [figure 4](#). The graph clearly shows significant variation in computational efficiency among the selected algorithms. Each classifier exhibits distinct model-building behavior based on its internal learning structure, computational complexity, and optimization mechanism.

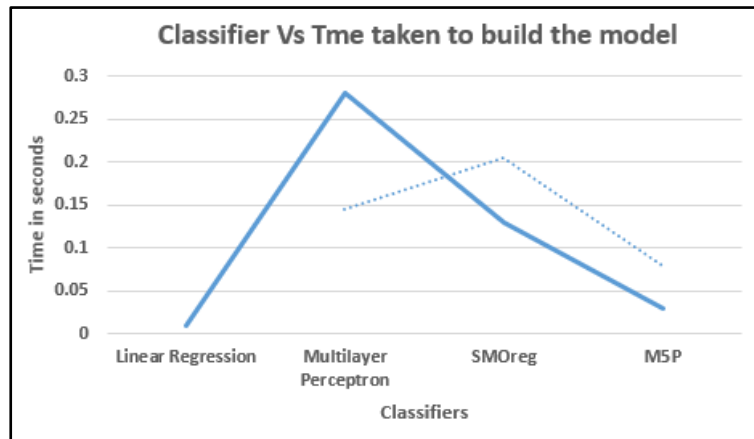


Figure 4. Classifier versus time taken to build the model

The Linear Regression model demonstrates the shortest training duration, with an approximate model-building time of only 0.01 seconds. This efficiency results from its relatively simple mathematical formulation, which relies on a closed-form analytical solution to minimize the error function. Since it involves no iterative optimization or backpropagation, Linear Regression provides an almost instantaneous computation time, making it highly suitable for real-time industrial control applications where rapid model deployment is essential.

In contrast, the MLP exhibits the longest model-building time, approximately 0.28 seconds. This substantial increase is primarily attributed to its multi-layered neural network architecture, which requires iterative weight adjustment through backpropagation and gradient descent across multiple epochs. The higher number of parameters and nonlinear activation functions substantially increase the computational load, especially during convergence toward the optimal error minimum. While this structure enables the MLP to capture complex nonlinear relationships, it also imposes a higher computational cost, making it less efficient for applications demanding low-latency control responses.

The SMOREg algorithm demonstrates intermediate performance, with a model-building time of around 0.13 seconds. This value reflects the algorithm's dependence on kernel-based optimization, which balances between accuracy and computational demand. Although SMOREg performs well in capturing nonlinear dependencies, the iterative process of solving multiple quadratic subproblems and evaluating kernel matrices contributes to its moderate training time. Despite being faster than MLP, its time complexity still limits its use in scenarios requiring continuous retraining or adaptive online control.

The M5P model tree combines the interpretability of regression analysis with the hierarchical structure of decision trees, resulting in a compact and efficient training process. With a total model-building time of approximately 0.03 seconds, it ranks as one of the most computationally efficient algorithms tested. Its efficiency arises from its two-phase approach: partitioning the data space using decision tree logic, followed by fitting local linear regression models within each leaf node. This hybrid mechanism allows the M5P to achieve both computational speed and predictive precision, making it particularly advantageous for adaptive real-time flow control systems.

Overall, the graph reveals a clear descending order of computational efficiency: Multilayer Perceptron (slowest), SMOREg (moderate), M5P (fast), and Linear Regression (fastest). The steep slope observed between the Multilayer Perceptron and M5P indicates a substantial computational gap between deep neural networks and tree-based regression techniques. This distinction highlights the practical importance of selecting models that balance accuracy with computational feasibility, particularly in industrial automation and IoT-based process control environments where real-time responsiveness is critical.

4.4. Error Distribution and Prediction Accuracy

The evaluation of prediction accuracy among the selected classifiers was conducted using RMSE as a key performance metric. RMSE quantifies the magnitude of the difference between the predicted output and the actual process value, making it a reliable indicator of how closely a model replicates the true system behavior. A lower RMSE value corresponds to higher predictive accuracy and a more stable learning model.

The comparison of RMSE values across the four classifiers is illustrated in [figure 5](#), which presents a clear visual differentiation in error distribution among the algorithms. The figure displays an ascending and descending trend forming a parabolic shape, where error values initially rise sharply from Linear Regression to SMOreg before declining again at M5P. This pattern reveals that the level of model complexity does not necessarily correlate with higher predictive performance, especially in small to moderately sized industrial datasets.

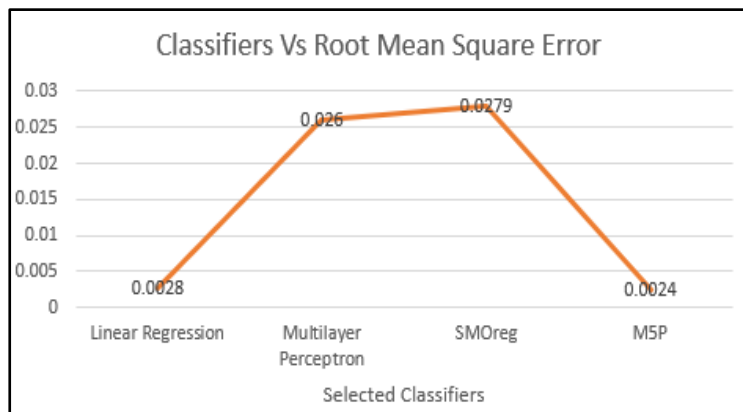


Figure 5. Classifiers versus Root Mean Square Error (RMSE)

The M5P model tree achieved the lowest RMSE value of 0.0024, confirming its superior predictive precision. This result highlights the model's hybrid strength—combining decision tree segmentation with localized linear regression at each leaf node. This structural approach allows the M5P model to effectively capture both linear and piecewise-linear patterns within the flow process data, yielding smooth and accurate predictions across all operating regions.

The Linear Regression model follows closely with an RMSE of 0.0028, demonstrating that for systems exhibiting predominantly linear or near-linear dynamics, traditional regression methods remain highly effective. Its minimal deviation from the M5P model indicates that the flow process under investigation behaves in a mostly linear manner, where simple regression models can approximate control responses with remarkable accuracy. Both models maintained stability and consistency during testing, with no significant outliers or fluctuations in prediction error.

In contrast, the MLP and SMOreg models produced noticeably higher RMSE values of 0.026 and 0.0279, respectively. The elevated error levels in these models stem from their nonlinear learning architectures, which, while theoretically capable of modeling complex system behaviors, require substantially larger datasets and longer training times to achieve equivalent accuracy. In the current experimental setup—characterized by a moderate number of training samples and quasi-linear process dynamics—these models exhibited slight overfitting tendencies and slower convergence, leading to higher residual errors.

The graphical pattern in [figure 5](#) reinforces the conclusion that simpler models, such as Linear Regression and M5P, outperform more complex nonlinear approaches under conditions where the underlying process dynamics are primarily linear or weakly nonlinear. Their reduced computational overhead and robust generalization make them highly practical for real-time flow control environments, where low-latency response and consistent accuracy are paramount.

Moreover, the sharp contrast between the peak error values of the Multilayer Perceptron and SMOreg models and the minimal errors of the M5P and Linear Regression models emphasizes the importance of model-to-process compatibility in data-driven control system design. The ability of the M5P model to combine interpretability, speed, and accuracy positions it as the optimal choice for embedding within an intelligent control framework that must adapt continuously to variations in process conditions.

4.5. Comprehensive Model Performance Analysis

A holistic evaluation of the classifiers' performance was conducted by comparing three critical indicators: RMSE, RAE, and the total time required to build each model. The combined results are illustrated in [figure 6](#), which provides a clear comparative view of each algorithm's predictive accuracy, computational efficiency, and structural complexity.

The figure shows that the Linear Regression and M5P pruned tree models achieved the most favorable balance between accuracy and computational speed. Both models recorded extremely low RMSE and RAE values, accompanied by minimal model-building time. The Linear Regression model demonstrates a near-instantaneous computation time, emphasizing its suitability for real-time industrial applications where rapid retraining or adaptive response is essential. The M5P pruned tree, although slightly slower than Linear Regression, exhibits marginally lower prediction error, proving its ability to model subtle nonlinearities that may exist in the process data.

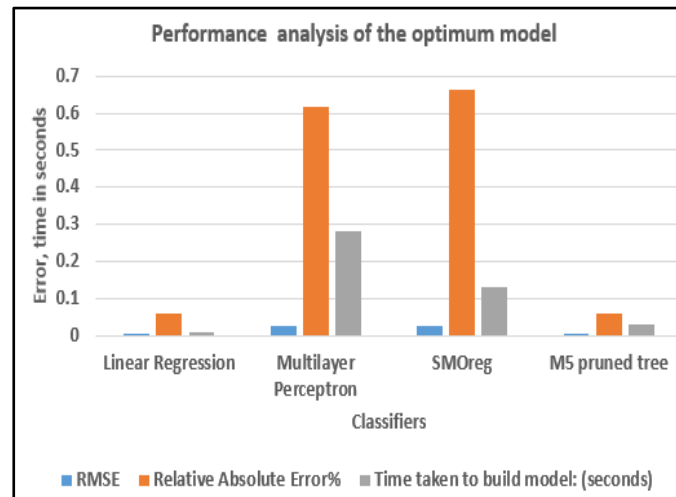


Figure 6. Performance analysis of the optimum model

The MLP model, on the other hand, shows a distinctly different behavior. As seen in the figure, the MLP exhibits the highest RAE percentage—approximately 0.62—and a significantly longer training time compared to the other classifiers. This increase in both error magnitude and computational time is attributed to the model’s multilayer architecture, which relies on iterative weight updates through backpropagation. While this deep structure enables the MLP to approximate complex nonlinear functions, it also introduces high computational cost and sensitivity to hyperparameter selection, such as learning rate and number of hidden neurons. In environments where control decisions must be made within milliseconds, such overhead can be impractical.

The SMOreg classifier follows a similar trend to the Multilayer Perceptron, with a slightly higher RAE value of approximately 0.66 and a moderate model-building time. The SMOreg algorithm, based on support vector regression, utilizes kernel functions to map data into higher-dimensional spaces. Although this enhances its ability to capture nonlinear relationships, it also significantly increases computational complexity, especially during kernel evaluation and optimization stages. The model’s relatively high RMSE and prolonged computation time make it less favorable for implementation in time-sensitive control systems, despite its robustness in handling noisy data.

In contrast, the M5P pruned tree model demonstrates an optimal compromise between simplicity, accuracy, and computational efficiency. Its structure, which combines decision tree partitioning with linear regression at the terminal nodes, allows it to deliver precise predictions while avoiding the overfitting problems often associated with purely nonlinear models. The M5P model’s low RMSE and RAE values indicate that it consistently generates accurate predictions with minimal residual error, while its shorter training time highlights its computational practicality. This makes it particularly well-suited for real-time intelligent control systems that require both fast response and high reliability.

The comparative behavior seen in [figure 6](#) therefore establishes two distinct performance clusters. The first cluster, consisting of Linear Regression and M5P, demonstrates low error values and short computation times, representing models that are ideal for online or embedded implementations. The second cluster, comprising Multilayer Perceptron and SMOreg, displays higher error rates and longer training durations, making them more suitable for offline learning scenarios or systems with ample computational resources.

From an implementation standpoint, the findings highlight the importance of aligning model complexity with system requirements. For industrial IoT-based flow control systems, where controllers must adapt dynamically with minimal

delay, the Linear Regression and M5P models are the most suitable. Their low RMSE and RAE ensure accurate predictive control, while their fast model-building time facilitates rapid retraining when new data is introduced. Conversely, while the Multilayer Perceptron and SMOreg algorithms offer flexibility in modeling nonlinear processes, their slower computation speeds and higher error margins make them less effective for real-time deployment.

In conclusion, the comparative performance analysis underscores the M5P pruned tree as the optimum classifier for the proposed smart controller. It achieves the best compromise between precision, adaptability, and computational efficiency. Its hybrid architecture effectively captures both linear and segmented process dynamics, ensuring that the smart controller can operate reliably under varying flow conditions with minimal deviation and high responsiveness.

4.6. Discussion

The results obtained from the comparative analysis of the four classifiers reveal a distinct clustering pattern in terms of both prediction accuracy and computational performance [20]. The MLP and SMOreg models form one group characterized by relatively higher error magnitudes and longer training durations. Both algorithms exhibit similar prediction behavior, with closely aligned RMSE and RAE values, suggesting that their nonlinear architectures respond in comparable ways to the given dataset. This similarity is attributed to their shared reliance on iterative optimization and nonlinear kernel transformations, which—while powerful in modeling complex relationships—require extensive data and computation to reach their full predictive potential. Under the conditions of this study, where the process data is predominantly quasi-linear, such complexity does not necessarily translate into superior accuracy [21].

In contrast, the Linear Regression and M5P models consistently outperformed the other classifiers, forming a second group that combines precision with computational efficiency. Both models achieved low RMSE and RAE values while maintaining extremely short model-building times, indicating their suitability for real-time implementation [22]. The M5P model, in particular, emerged as the most effective classifier. Its hybrid design, which integrates decision-tree segmentation with local linear regression at each terminal node, enables it to partition the input space adaptively and model localized variations within the process data. This capability allows the M5P model to capture complex process dynamics that would otherwise be difficult to represent using purely linear approaches, while still retaining the interpretability and simplicity characteristic of regression models.

A key strength of the M5P algorithm lies in its ability to generalize effectively without overfitting. By pruning the decision tree during training, the model avoids unnecessary branching and maintains smooth transitions between data regions. This structural regularization ensures that the model remains stable and computationally lightweight, which is essential for integration into embedded industrial controllers [23]. The model's consistent predictive accuracy across multiple cross-validation trials further confirms its robustness and reliability for dynamic process environments.

The Linear Regression model, though simpler in structure, also demonstrated commendable predictive accuracy. Its performance closely matched that of the M5P model, indicating that the flow process exhibits primarily linear relationships between input and output variables under steady-state conditions. This result underscores an important insight: for many practical control systems, particularly those with linear or weakly nonlinear behavior, simpler models can deliver equivalent or even superior results compared to more complex architectures [24].

From a control systems perspective, the performance of the M5P-based intelligent controller mirrors that of a conventional PID controller. Both approaches effectively minimize steady-state error and maintain smooth dynamic responses. However, the intelligent controller offers a distinct advantage—it does not require manual tuning of proportional, integral, and derivative gains. Instead, it continuously adapts its predictive mapping based on data patterns learned from the process itself. This self-learning capability enables faster adaptation to process disturbances, nonlinearity, or parameter drift, which are common challenges in real-world flow control systems [20].

Furthermore, the comparative evaluation validates that machine learning-based predictive control can achieve an equivalent level of precision as traditional control methods, while significantly improving flexibility and scalability. The M5P and Linear Regression models' ability to provide near-instantaneous prediction updates makes them ideal for modern IoT-enabled control architectures, where real-time decision-making and continuous optimization are required. Their compact computational footprint also allows for seamless deployment on embedded hardware such as microcontrollers and edge devices [21].

Overall, the findings confirm that the M5P pruned tree model is the optimal solution for the proposed smart flow controller. It effectively balances prediction accuracy, model interpretability, and computational efficiency. By replicating the dynamic performance of a conventional PID controller while offering superior adaptability to changing operating conditions, the M5P-based intelligent controller represents a significant advancement in data-driven process automation. This approach demonstrates how supervised learning techniques can be successfully integrated into control system design, bridging the gap between classical control theory and modern artificial intelligence [22], [23], [24].

5. Conclusion

The experimental results confirm that the performance of the intelligent flow controller developed in this study closely aligns with that of a conventional PID controller in terms of accuracy, stability, and dynamic response. Among the evaluated models, the Multilayer Perceptron and SMOreg classifiers exhibited nearly identical error distributions, demonstrating consistent predictive trends despite their higher computational complexity. In contrast, the Linear Regression and M5P pruned tree models consistently achieved superior performance, with both delivering minimal prediction error and rapid computation. This reinforces the validity of implementing supervised learning-based approaches for intelligent flow control, where data-driven models can replicate and, in some cases, outperform traditional PID control strategies.

The M5P pruned tree model, in particular, proved to be the most effective among all tested algorithms. Its hybrid structure—combining decision-tree segmentation with local linear regression—enables it to capture subtle nonlinearities in process behavior while maintaining computational efficiency and interpretability. The findings demonstrate that replacing conventional controllers with machine learning-based models can significantly reduce tuning complexities, eliminate the need for manual parameter adjustment, and enhance overall system adaptability.

Beyond performance, the study highlights the broader implications of machine learning (ML)-based flow controllers in modern industrial automation. These controllers rely on data-driven modeling, AI frameworks, and in certain architectures, cloud or edge computing integration. While more computationally intensive than PID controllers, they offer far greater autonomy, predictive intelligence, and adaptability—qualities that are crucial for IoT-enabled smart instrumentation systems. By leveraging real-time data for continuous learning and predictive decision-making, ML-based controllers can enhance signal conditioning efficiency, minimize process disturbances, and improve overall system reliability.

Future research should focus on expanding this work toward cascade process control systems, where slave controllers can be replaced or augmented by smart controllers to further reduce tuning and coordination complexity. Integrating federated learning frameworks presents another promising direction, allowing decentralized model training across multiple IoT nodes while preserving data privacy and improving scalability. In parallel, exploring adaptive self-tuning models and energy-efficient learning algorithms will be essential for enabling deployment in resource-constrained embedded environments.

Collectively, these advancements will pave the way toward more autonomous, robust, and secure smart control architectures that can dynamically adapt to environmental variations and process uncertainties. Such innovations will not only enhance industrial automation and signal conditioning systems but also extend their applications into emerging fields such as biomedical instrumentation, environmental monitoring, and intelligent manufacturing.

6. Declarations

6.1. Author Contributions

Conceptualization: S.P., B.K., and M.B.; Methodology: S.P.; Software: B.K.; Validation: S.P., B.K., and M.B.; Formal Analysis: S.P., B.K., and M.B.; Investigation: S.P.; Resources: B.K.; Data Curation: B.K.; Writing Original Draft Preparation: S.P., B.K., and M.B.; Writing Review and Editing: S.P., B.K., and M.B.; Visualization: S.P.; All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

- [1] M. A. Fawwaz, K. Bingi, R. Ibrahim, P. A. M. Devan, and B. R. Prusty, "Design of PIDDA Controller for Robust Performance of Process Plants," *Algorithms*, vol. 16, no. 9, pp. 437-449, 2023, doi: 10.3390/a16090437.
- [2] K. Bakthavatchalam, B. Karthik, V. Thiruvengadam, S. Muthal, D. Jose, K. Kotecha, and V. Varadarajan, "IoT Framework for Measurement and Precision Agriculture: Predicting the Crop Using Machine Learning Algorithms," *Technologies*, vol. 10, no. 1, pp. 1-13, 2022, doi: 10.3390/technologies10010013.
- [3] "WEKA: Waikato Environment for Knowledge Analysis," University of Waikato, New Zealand. [Online]. Available: <http://www.cs.waikato.ac.nz>
- [4] P. Wang, F. Ye, X. Chen, and Y. Qian, "Datanet: Deep Learning-Based Encrypted Network Traffic Classification in SDN Home Gateway," *IEEE Access*, vol. 6, no. 1, pp. 55380-55391, 2018, doi: 10.1109/ACCESS.2018.2872430.
- [5] P. Ramanathan, K. K. Mangla, and S. Satpathy, "Smart Controller for Conical Tank System Using Reinforcement Learning Algorithm," *Measurement*, vol. 116, no. 1, pp. 422-428, Feb. 2018, doi: 10.1016/j.measurement.2017.11.007.
- [6] L. Liu, P. Wang, J. Lin, and L. Liu, "Intrusion Detection of Imbalanced Network Traffic Based on Machine Learning and Deep Learning," *IEEE Access*, vol. 9, no. 1, pp. 7550-7563, 2021, doi: 10.1109/ACCESS.2020.3048198.
- [7] M. Mengcan, C. Xiaofang, and X. Yongfang, "Constrained Voting Extreme Learning Machine and Its Application," *Journal of Systems Engineering and Electronics*, vol. 32, no. 1, pp. 209-219, Feb. 2021, doi: 10.23919/JSEE.2021.000018.
- [8] J. Tang, C. Deng, and G.-B. Huang, "Extreme Learning Machine for Multilayer Perceptron," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 4, pp. 809-821, Apr. 2016, doi: 10.1109/TNNLS.2015.2424995.
- [9] R. Khalid, N. Javaid, M. H. Rahim, S. Aslam, and A. Sher, "Fuzzy Energy Management Controller and Scheduler for Smart Homes," *Sustainable Computing: Informatics and Systems*, vol. 21, no. 1, pp. 103-118, Mar. 2019, doi: 10.1016/j.suscom.2018.11.010.
- [10] P. Singh and S. Agrawal, "Node Localization in Wireless Sensor Networks Using the M5P Tree and SMOreg Algorithms," in *Proc. IEEE Conf. Computational Intelligence and Communication Networks (CICN)*, vol. 2013, no. 1, pp. 104-104, 2013, doi: 10.1109/CICN.2013.32.
- [11] X. Liu, "Application of Temperature Prediction Based on Neural Network in Intrusion Detection of IoT," *Security and Communication Networks*, vol. 2018, no. 1, pp. 1-10, Dec. 2018, doi: 10.1155/2018/1635081.
- [12] S. O. Manta-Costa, R. S. Araújo, R. S. Peres, and J. Barata, "Machine Learning Applications in Manufacturing—Challenges, Trends, and Future Directions," *IEEE Open Journal of the Industrial Electronics Society*, vol. 5, no. 1, pp. 1085-1103, 2024, doi: 10.1109/OJIES.2024.3431240.
- [13] H. El-Sofany, S. A. El-Seoud, O. H. Karam, and A. H. Kassem, "Using Machine Learning Algorithms to Enhance IoT System Security," *Scientific Reports*, vol. 14, no. 12077, pp. 1-12, 2024, doi: 10.1038/s41598-024-62861-y.
- [14] S. Pandey, M. Chaudhary, and Z. Tóth, "An Investigation on Real-Time Insights: Enhancing Process Control with IoT-Enabled Sensor Networks," *Discover Internet of Things*, vol. 5, no. 29, pp. 1-12, 2025, doi: 10.1007/s43926-025-00124-6.
- [15] M. S. Hasibuan, R. Z. A. Aziz, D. A. Dewi, T. B. Kurniawan, and N. A. Syafira, "Recommendation Model for Learning Material Using the Felder Silverman Learning Style Approach," *HighTech and Innovation Journal*, vol. 4, no. 4, pp. 811-820, Dec. 2023, doi: 10.28991/HIJ-2023-04-04-010.

-
- [16] S. Malik, "Data-Driven Decision-Making: Leveraging the IoT for Real-Time Sustainability in Organizational Behavior," *Sustainability*, vol. 16, no. 15, pp. 6302-6318, 2024, doi: 10.3390/su16156302.
- [17] B. Kalaiselvi, "Design of Smart Positioner for a Control Valve to Optimise Backlash Problem," *Indian Journal of Science and Technology*, vol. 8, no. 32, pp. 1-12, Nov. 2015, doi: 10.17485/ijst/2015/v8i32/87454.
- [18] B. Kalaiselvi and R. Karthekeyan, "Impact of Intelligent Controller in a Multiprocess System Using Artificial Neural Network—BPN," in *Proc. Int. Conf. Green Computing Communication and Electrical Engineering (ICGCCEE)*, Coimbatore, India, vol. 2014, no. 1, pp. 1-6, doi: 10.1109/ICGCCEE.2014.6922414.
- [19] G. Bujgoi and D. Sendrescu, "Tuning of PID Controllers Using Reinforcement Learning for Nonlinear System Control," *Processes*, vol. 13, no. 3, pp. 735-749, 2025, doi: 10.3390/pr13030735.
- [20] M. Blessing and S. Suriseti, "Real-Time Control and Optimization in Machine Learning for Dynamic Systems," *Journal of Intelligent Control and Systems*, vol. 12, no. 2, pp. 210-220, 2024.
- [21] K. Cheon, J. Kim, M. Hamadache, and D. Lee, "On Replacing PID Controller with Deep Learning Controller for DC Motor System," *Journal of Automation and Control Engineering*, vol. 3, no. 6, pp. 452-456, 2015, doi: 10.12720/joace.3.6.452-456.
- [22] G. K. H. Pang, "Intelligent Process-Control System Design," *Engineering Applications of Artificial Intelligence*, vol. 5, no. 6, pp. 493-503, 1992, doi: 10.1016/0952-1976(92)90026-G.
- [23] B. S. Kang, D. H. Choe, and S. C. Park, "Intelligent Process Control in Manufacturing Industry with Sequential Processes," *International Journal of Production Economics*, vol. 60-61, no. 1, pp. 583-590, 1999, doi: 10.1016/S0925-5273(98)00178-9.
- [24] Z. Chang, A.-J. Li, and Y. Li, "Intelligent Process Control with Uninformative Data," in *Proc. 28th Int. Conf. Automation and Computing (ICAC)*, Birmingham, United Kingdom, vol. 2023, no. 1, pp. 1-6, doi: 10.1109/ICAC57885.2023.10275196.