APPLICATION OF HYBRID NEURAL – PID CONTROL SYSTEM TO CONTROL WATER LEVEL IN TANK USING PLC S7-400

Vo Hong Ngan[1*], Nguyen Xuan Toai[1], Huynh Thanh Son[1], Vo Thanh Tung[1], Nguyen Quoc Y[2]

[1]*Dong Nai Technology University*
[2]*Binh Duong Economics and Technology University*
*Corresponding author: *Vo Hong Ngan, vohongngan@dntu.edu.vn*

GENERAL INFORMATION

Received date: 10/01/2025

Revised date: 26/04/2025

Accepted date: 03/07/2025

KEYWORD

Neural;

Neural-PID;

PLC;

PID;

SCL.

ABSTRACT

Stability is a critical requirement in industrial control systems. A major challenge lies in designing a high-performance controller capable of meeting the stringent accuracy demands of technological processes. This paper proposes the development of a hybrid Neural–PID controller, which integrates the stability advantages of the conventional PID controller with the adaptive learning capabilities of a neural network, to regulate the water level in a single-tank system. The neural network utilized is a three-layer feedforward architecture, trained using a supervised learning approach. Experimental results indicate that the hybrid Neural-PID controller outperforms the traditional PID controller, particularly under nonlinear operating conditions and scenarios involving significant system variations. During setpoint changes, the Neural-PID controller achieves approximately a 50% reduction in overshoot compared to the conventional PID controller. Moreover, the control error remains low at around 0.5%, and the settling time is significantly faster. Even under substantial system disturbances, the Neural-PID controller maintains effective and robust regulation. Furthermore, this study demonstrates that simple neural networks can be directly implemented on the S7-400 Programmable Logic Controller (PLC) using the Structured Control Language (SCL), thereby paving the way for new applications of artificial intelligence techniques in industrial automation systems.

1. INTRODUCTION

In the field of automatic control systems, PID controllers remain among the most widely adopted control strategies due to their structural simplicity, ease of implementation, and robust performance across a broad range of applications. Nevertheless, when deployed in nonlinear systems or systems subject to significant dynamic variations, conventional

PID controllers often face challenges such as suboptimal parameter tuning under changing operating conditions, system delays, and sensitivity to noise (Araki, 2009; Somefun et al., 2021).

To address these limitations, the integration of emerging technologies such as Artificial Neural Networks (ANNs) with traditional PID frameworks has gained considerable attention as a promising research direction (Webb et al., 2011). Neural networks possess the capability to learn from data and to model complex nonlinear and uncertain system behaviors, thereby enhancing control performance where conventional PID controllers typically struggle (Yu & Rosen, 2013; Grossberg, 2013). The hybrid Neural-PID control approach merges machine learning capabilities, particularly those of artificial neural networks, with classical PID controllers to enhance the adaptability and performance of control systems under complex operating conditions (Rossomando & Soria, 2015).

In this study, the authors focus on the development and implementation of Neural-PID controllers by constructing control functions on the Siemens automation platform. Specifically, the SCL (Structured Control Language) available within the SIMATIC system is utilized to implement and compare two controller types: traditional PID and hybrid Neural-PID controllers (Berger, 2012). While each controller exhibits inherent strengths and limitations, the integration of neural networks with PID aims to leverage the advantages of both methods to optimize control performance. This paper particularly emphasizes the analysis and comparative evaluation of the PID and Neural-PID controllers in regulating the water level of a real single-tank system (Palomino et al., 2022; SJ et al., 2023).

2. METHODOLOGY

2.1. Research method

This research adopts an experimental approach to determine the optimal controller for a single water tank system, as depicted in Figure 1, which represents a plant exhibiting nonlinear characteristics. The system illustrated in Figure 1 comprises a water tank with a variable water level to be regulated, a pump, and sensors for real-time water level measurement. Due to frequent fluctuations in the water level and the presence of noise, the control and simulation processes face considerable challenges. The SIMATIC MANAGER software, specifically designed for the S7-400 controller, is utilized in conjunction with hardware components such as the E4PA Ultrasonic Sensor (Omron), the MM420 inverter, and a Redlion industrial touchscreen panel to control the pump speed supplying water to the tank.

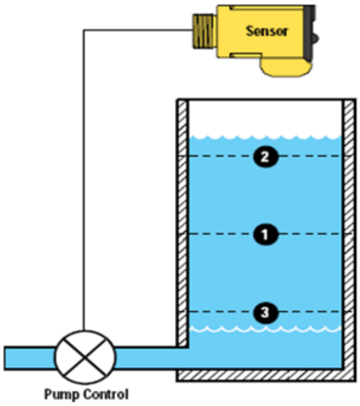


Figure 1. Single water tank control system

In this study, the authors implemented two primary controllers using the Structured Control Language (SCL) within the SIMATIC environment: a conventional PID controller and a hybrid Neural-PID controller that combines the respective advantages of PID control and artificial neural networks. The primary objectives of the controller design are to achieve system stability, operational robustness, optimal efficiency, low implementation cost, and ease of deployment in practical applications. Experimental results confirm that the hybrid Neural-PID controller delivers superior performance in regulating the single water tank system, maintaining very low control error and high stability under dynamic operating conditions.

2.2. Water tank control system design

2.2.1. General diagram of water tank control system

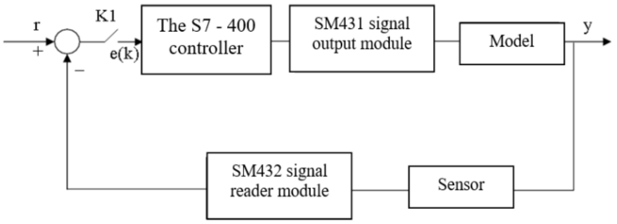


Figure 2. General diagram of water tank control system

The water tank control system is illustrated in Figure 2. The system operates synchronously, with the control key (denoted by symbol K) coordinating the process. The S7-400 controller receives feedback signals from the water level sensor via the SM432 Analog Input Module. The controller then compares the measured feedback signal with the setpoint value provided through the industrial HMI panel. Based on this comparison, two control algorithms PID and Neural-PID are executed within the S7-400 controller for processing. The resulting control signal is subsequently transmitted to the SM431 Analog Output Module, which drives the actuator to regulate the water tank system accordingly.

2.2.2. PID Controller:

The diagram of the PID control system is shown in Figure 3. In the diagram figure 3 includes the following components:

- The transfer function of the PID regulator has the form:

$$G(s) = K_p + \frac{K_i}{s} + K_d s = K_p\left(1 + \frac{1}{T_I s} + T_D s\right) \quad (1)$$

K_p, K_i, K_d are real constants

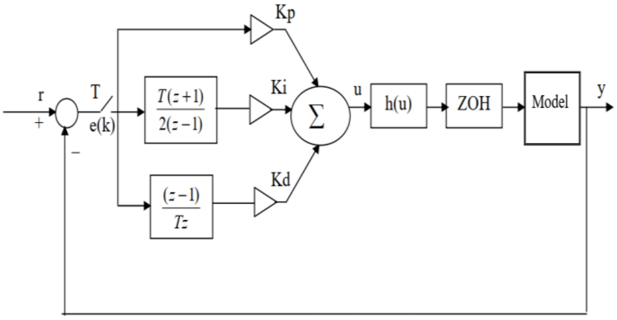


Figure 3. PID controller block diagram

The PID controller parameters in this study are selected based on the Ziegler–Nichols tuning method, taking into account the specific dynamic characteristics of the plant. For the liquid level control application in the water tank system, the PID parameters are determined by analyzing the transient response of the closed-loop system, as illustrated in Figure 4.

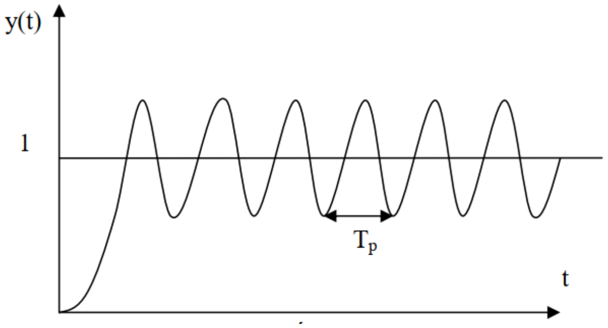


Figure 4. Step response of closed system when $K = K_{max}$

Initially, K_p, K_i, K_d are zero, control with K_p gradually increasing to K_{max} when the response oscillates around the set value with period T_p. The parameters are selected as in Table 1.

Table 1. Second Zeigler-Nichols tuning law

Controller	K_p	T_I	T_D
P	$0.5K_{max}$	∞	0
PI	$0.45K_{max}$	$0.83T_P$	0
PID	$0.6K_{max}$	$0.5T_P$	$0.125T_P$

- ZOH: sampling hold.
- Control law u:

$$u = u_p + u_i + u_d \tag{2}$$

$$u(kT) = K_p e(kT) + u_i[(k-1)T] + \frac{K_i T}{2}\left(e[(k-1)T] + e(kT)\right) + K_d \frac{e(kT) - e[(k-1)T]}{T} \tag{3}$$

2.2.2. Neural - PID Controller:

The Neural-PID controller integrates the stability characteristics of the conventional PID controller with the adaptive learning capability of a neural network (Zhang et al., 2011; Luo et al., 2022; Yang et al., 2024). The neural network employed is a three-layer feedforward structure, as depicted in Figure 5(a), consisting of an input layer, a hidden layer, and an output layer. The input layer comprises two neurons that receive the error signal (e) and its derivative (de/dt). The hidden layer consists of six neurons utilizing a symmetric S-shaped activation function. The output layer contains three neurons employing a sigmoid activation function, with each output neuron corresponding to the adjustment of one of the PID parameters: Kp, Ki, Kd, respectively. In the diagram in Figure 5(a) includes:

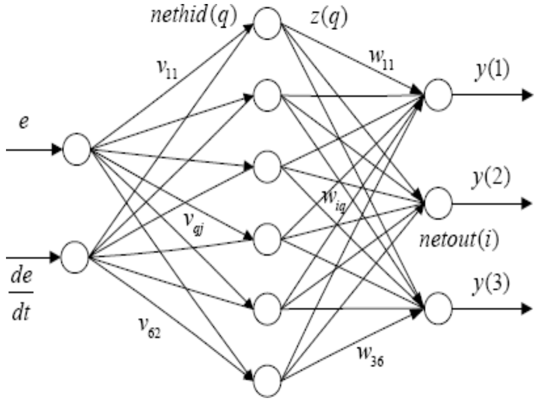
$x_j\ (j = \overline{1,m})$: receive signals from e and de/dt.

$z_q\ (q = \overline{1,l})$: The output of unit q in the hidden layer with l equal to 6.

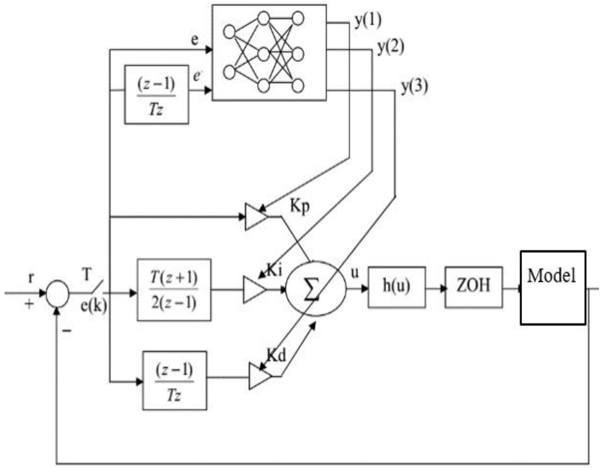
$y_i\ (i = \overline{1,n})$: outputs of the output layer with n equal to 3.

$v_{qj}\ (q = \overline{1,l}; j = \overline{1,m})$: is the connection weight between the input layer and the hidden layer.

$w_{iq}\ (i = \overline{1,n}; q = \overline{1,l})$: is the connection weight between the hidden layer and the output layer.



(a)



(b)

Figure 5. Neural-PID controller: (a) Neural connection diagram, (b) Neural-PID control system

The weighted sum of the inputs to the *q*th neuron in the hidden layer is calculated as follows:

$$net_q = \sum_{j=1}^{m} v_{qj} x_j \qquad (4)$$

The output signal of the *q*th neuron in the hidden layer is:

$$z_q = a_h(net_q) = a_h\left(\sum_{j=1}^{m} v_{qj} x_j\right) \qquad (5)$$

The weighted sum of the input signals to the *i*th neuron in the output layer is:

$$net_i = \sum_{q=1}^{l} w_{iq} z_q = \sum_{q=1}^{l} w_{iq} a_h(net_q)$$

$$= \sum_{q=1}^{l} w_{iq} a_h\left(\sum_{j=1}^{m} v_{qj} x_j\right) \qquad (6)$$

The output signal of the *i*th neuron in the output layer is:

$$y_i = a_0(net_i) = a_0\left(\sum_{q=1}^{l} w_{iq} z_q\right) \qquad (7)$$

Suppose we have a training dataset consisting $k = \overline{1, K}$

of *K* samples (*x*(*k*), *d*(*k*)),

The criterion for training the network is to minimize the error:

$$E = \frac{1}{2}\sum_{i=1}^{n}(d_i - y_i)^2 \qquad (8)$$

The parameters E and k are trained to update the connection weights between layers with the input signal *x*(*k*).

First set *E* =0, *k*=1. Then calculate cumulative error:

$$E = E + \frac{1}{2}\sum_{i=1}^{n}\left(d_i(k) - y_i(k)\right)^2 \qquad (9)$$

If $E \le E_{max}$ then end the learning process.

If $E > E_{max}$ then set *E* =0, *k* =1, and go back to start a new training cycle.

The function *E* forms a convex surface in space, on which there exists a minimum point. Given any set of weights, we can compute the value of *E* on this convex surface. In order to ensure that the system converges quickly to the minimum point, it is necessary to adjust the weights of the network in a suitable direction so that the error function E decreases as quickly as possible along its curved surface.

Figure 5(b) is the diagram of the Neural – PID controller. The control law *u* is similar to the PID controller but the control coefficients K_p, K_i, K_d are replaced by the outputs of the Neural controller *y*(1), *y*(2), *y*(3) as follows:

$$u(kT) = y(1)e(kT) + u_I[(k-1)T]$$

$$+\frac{y(2)T}{2}\left(e[(k-1)T] + e(kT)\right) \qquad (10)$$

$$+y(3)\frac{e(kT) - e[(k-1)T]}{T}$$

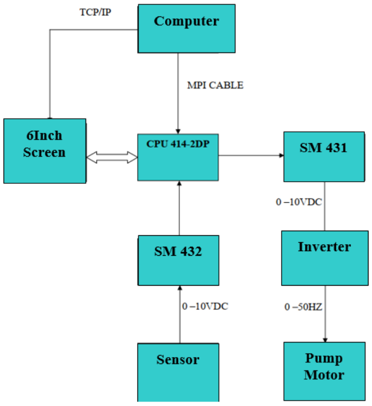
In this system, the weights are updated after each change in input and output. This training method allows the network to learn in real time while the system is still operating, helping the network adapt to any changes and fluctuations in the system. The advantage of this method is that the network has the ability to adapt quickly and maintain good stability when the system changes over time.

3. RESULT AND DISCUSSION

3.1. Realistic water tank control model

The water tank control model is developed, and the system connection diagram is presented in Figures 6(a) and 6(b). As illustrated in Figure 6(b), the computer interfaces with the industrial Human-Machine Interface (HMI) panel to collect and manage data, modify setpoint values, and display real-time control process graphs. The SM432 analog input module acquires signals from the water level sensor and compares them with the setpoint values provided by the HMI. The CPU 414 processes control algorithms, including PID and Neural-PID control strategies, and generates the control signal, which is output through the SM431 analog output module. The SM431 module delivers a 0–10 VDC analog signal to drive the inverter, which in turn adjusts the speed of the motor responsible for pumping water into the tank. The S7-400 controller is responsible for deciding the control output, while the inverter executes the physical actuation. The software tools employed in the system include STEP 7 V5.5 for programming the S7-400 controller and Crimson 3.0 for configuring the HMI panel.

(a)



(b)

Figure 6. Actual control system diagram (a) Actual model (b) System operation diagram

3.2. Water tank control with PID controller

The values of K_p, K_i, K_d are obtained by trial and error as shown in Table 2, first adjust K_p first then adjust K_i, K_d, in the case of the actual model, the noise obtained is the oscillation of the water surface. Initially set the setpoint to 100, after the setpoint is changed to 150, when reaching the setpoint, change the setpoint to 200. The water tank control result of the PID controller gives the control result as shown in the graph in Figure 7. It is seen that when the Setpoint is at 100, the control signal is quite good. When changing the Setpoint from 100 to 150 and up to 200, the system still gives quite good control results, but the overshoot is still quite high, the error is close to 1% but is small enough to be acceptable (the error is caused by model noise). The system needs a period of time to stabilize between setpoint changes and then the system operates very stably.

Table 2. PID controller parameters

Parameter	Value
K_p	4.8
K_i	0.08
K_d	0.9

The PID controller gives good model execution results. The error of the model control process is small enough to be acceptable. The traditional PID controller has the advantage of being easy to control the model, but the ability to respond to system changes is slow.

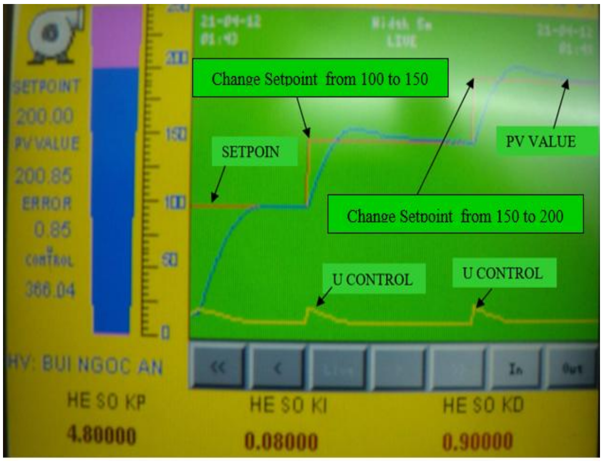


Figure 7. PID control results with Setpoint change

3.3 Water tank control with Neural - PID controller

Initially set point is 125 after setting change set point is 150, when reaching setting change set point is 100. The Neural-PID controller exhibits good model performance, as shown in Figure 8. It is evident that when the setpoint is maintained at 125, the system experiences a noticeable delay. However, when the setpoint is increased from 125 to 150, the system responds more rapidly, attributed to the learning capability of the neural network. When the setpoint is decreased from 150 to 100, a control error appears; however, the error magnitude is relatively small and acceptable, mainly due to the presence of model noise. Subsequently, the system maintains stability and demonstrates excellent flexibility in responding to process variations. The controller parameters have been effectively learned and updated, achieving a control error of approximately 0.5%.

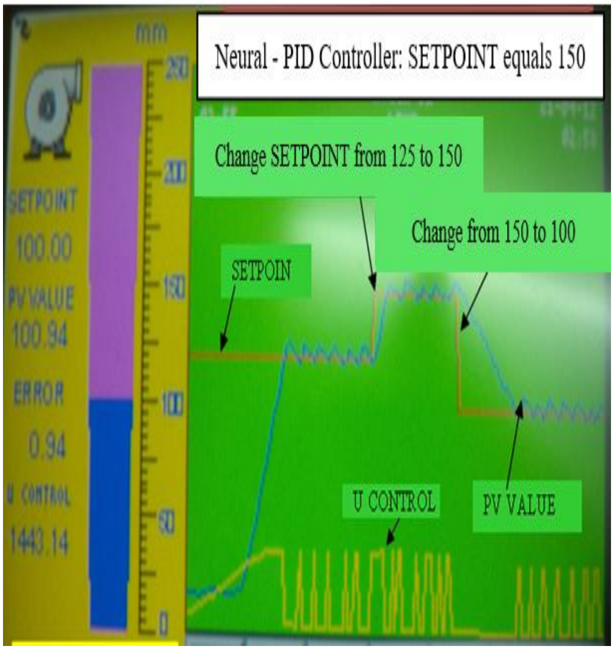


Figure 8. Neural - PID control results with Setpoint change

Figure 9 presents the comparison results between the PID controller and the Neural-PID controller under a setpoint change from 100 to 150. It can be observed that, when the setpoint changes, the Neural-PID controller achieves approximately 50% lower overshoot compared to the conventional PID controller. Although the Neural-PID controller exhibits some initial overshoot, its settling time to reach the desired control value is faster than that of the PID controller. The controller parameters have been effectively learned and updated. Measurements taken for both the PID controller and the Neural-PID controller indicate that both controllers provide stable control signals; however, the hybrid Neural-PID controller delivers a better output response than the traditional PID controller. The Neural-PID controller thus demonstrates good applicability for industrial control systems subject to large system variations.

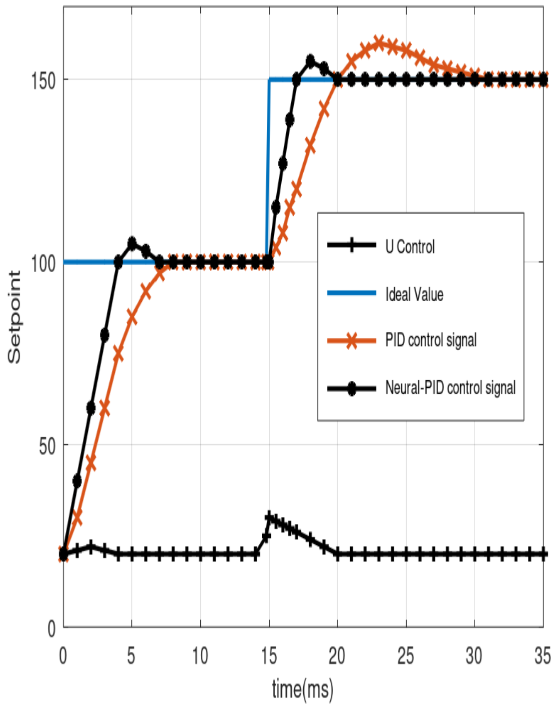


Figure 9. Comparison results of PID controller and Neural - PID controller with Setpoint change from 100 to 150

4. CONCLUSION

Two controllers, namely the PID and the Neural-PID, have been implemented on the S7-400 system using the Structured Control Language (SCL), and have demonstrated high effectiveness in controlling a real single water tank model. The control errors observed during testing were very small, with the *Kp*, *Ki* and *Kd* parameters initially determined based on empirical tuning. Experimental measurements indicate that the Neural-PID controller exhibits significantly reduced overshoot approximately 50% lower compared to the conventional PID controller. The steady-state error of the control process remains low, around 0.5%, and the settling time achieved by the Neural-PID controller is shorter than that of the traditional PID controller. Moreover, the Neural-PID controller maintains effective responsiveness under rapid changes in operating conditions. By integrating the learning capability of neural networks with the inherent stability of PID control, the hybrid Neural-PID controller leverages the advantages of both approaches, resulting in more optimal and robust control performance across experimental evaluations.

REFERENCES

Araki, M. (2009). PID control. *Control Systems, Robotics and Automation: System Analysis and Control: Classical Approaches II*, 58-79.

Berger, H. (2012). *Automating with STEP 7 in STL and SCL: SIMATIC S7-300/400*

Grossberg, S. (2013). Recurrent neural networks. *Scholarpedia*, *8*(2), 1888.

Luo, J., Zhu, L., Wu, N., Chen, M., Liu, D., Zhang, Z., & Liu, J. (2022). Adaptive neural-PID visual servoing tracking control via extreme learning machine. *Machines*, *10*(9), 782.

Mendoza, A. M. E. R., & Yu, W. (2023). Fuzzy adaptive control law for trajectory tracking based on a fuzzy adaptive neural PID controller of a multi-rotor unmanned aerial vehicle. *International Journal of Control, Automation and Systems*, *21*(2), 658-670.

Palomino, S., Gutarra, A., Quispe, A., Alegria, E. J., & Diaz, C. (2022, November). Control of a Water Tank Level Process Using IMC Controller. In *2022 IEEE ANDESCON* (pp. 1-5). IEEE.

Rossomando, F., & Soria, C. (2015). Design and implementation of adaptive neural PID for non linear dynamics in mobile robots. *IEEE Latin America Transactions*, *13*(4), 913-918.

Somefun, O. A., Akingbade, K., & Dahunsi, F. (2021). The dilemma of PID tuning. *Annual Reviews in Control*, *52*, 65-74.

SJ, S. P., Sureshkumar, R., Yogaabitha, A., Vijayakumar, B., Sakthivel, M., & Purshothaman, S. (2023, June). Development of PLC Based Level Control System for Conical Tank System for Water Treatment Plant. In *2023 3rd International Conference on Pervasive Computing and Social Networking (ICPCSN)* (pp. 1563-1568). IEEE.

Tan, K. K., Wang, Q. G., & Hang, C. C. (2012). *Advances in PID control*. Springer Science & Business Media.

Webb, A., Davies, S., & Lester, D. (2011). Spiking neural PID controllers. In *Neural Information Processing: 18th International Conference, ICONIP 2011, Shanghai, China, November 13-17, 2011, Proceedings, Part III 18* (pp. 259-267). Springer Berlin Heidelberg.

Yang, Q., Zhang, F., & Wang, C. (2024). Deterministic Learning-Based Neural PID Control for Nonlinear Robotic Systems. *IEEE/CAA Journal of Automatica Sinica*, *11*(5), 1227-1238.

Zhang, Y., Yu, X., Bi, M., & Xu, S. (2011, August). An adaptive neural PID controller for torque control of hybrid electric vehicle. In *2011 6th International Conference on Computer Science & Education (ICCSE)* (pp. 901-903). IEEE.

Zhang, J., Yang, X., Wang, W., Brilakis, I., Wang, H., & Ding, L. (2024). Transformer-based pavement crack tracking with neural-PID controller on vision-guided robot. In *ISARC. Proceedings of the International Symposium on Automation and Robotics in Construction* (Vol. 41, pp. 699-706). IAARC Publications.

Yu, W., & Rosen, J. (2013). Neural PID control of robot manipulators with application to an upper limb exoskeleton. *IEEE Transactions on cybernetics*, *43*(2), 673-684.

Yu, W., Li, X., & Panuncio, F. (2014). Stable neural PID anti-swing control for an overhead crane. *Intelligent Automation & Soft Computing*, *20*(2), 145-158.

ỨNG DỤNG HỆ THỐNG ĐIỀU KHIỂN NEURAL – PID ĐỂ ĐIỀU KHIỂN MỨC NƯỚC TRONG BỒN CHỨA SỬ DỤNG PLC S7-400

Võ Hồng Ngân[1*], Nguyễn Xuân Toại[1], Huỳnh Thanh Sơn[1], Võ Thanh Tùng[1], Nguyễn Quốc Ý[2]

[1]Trường Đại học Công nghệ Đồng Nai
[2]Trường Đại học Kinh tế - Kỹ thuật Bình Dương
*Corresponding author: *Võ Hồng Ngân*, *vohongngan@dntu.edu.vn*

THÔNG TIN CHUNG

Ngày nhận bài: 10/01/2025

Ngày nhận bài sửa: 26/04/2025

Ngày duyệt đăng: 03/07/2025

TỪ KHOÁ

Neural;
Neural-PID;
PLC;
PID;
SCL

TÓM TẮT

Tính ổn định là một yêu cầu then chốt trong các hệ thống điều khiển công nghiệp. Thách thức lớn nhất nằm ở việc thiết kế một bộ điều khiển tốt có khả năng đáp ứng những yêu cầu nghiêm ngặt về độ chính xác của các quá trình công nghệ. Bài báo này đề xuất phát triển một bộ điều khiển Neural–PID, kết hợp ưu điểm về tính ổn định của bộ điều khiển PID truyền thống với khả năng học thích nghi của mạng Neural, nhằm điều khiển mức nước cho một bồn nước đơn. Mạng Neural được sử dụng có cấu trúc mạng truyền thẳng ba lớp, được huấn luyện theo phương pháp học có giám sát. Kết quả thực nghiệm cho thấy bộ điều khiển Neural–PID vượt trội hơn bộ điều khiển PID truyền thống, đặc biệt trong điều kiện hoạt động phi tuyến và khi hệ thống có biến thiên lớn. Trong quá trình thay đổi các điểm đặt, bộ điều khiển Neural-PID có độ vọt lố thấp cải thiện 50% so với bộ điều khiển PID truyền thống. Sai số của quá trình điều khiển nhỏ khoảng 0,5% và thời gian tiến đến ổn định của bộ điều khiển Neural-PID nhanh hơn bộ điều khiển PID. Khi có sự thay đổi lớn trong hệ thống thì bộ điều khiển Neural-PID vẫn điều khiển thay đổi theo tốt. Ngoài ra, nghiên cứu này chứng minh rằng các mạng Neural đơn giản có thể được triển khai trực tiếp trên PLC S7-400 bằng ngôn ngữ Structured Control Language (SCL), mở ra hướng ứng dụng mới cho các kỹ thuật trí tuệ nhân tạo trong hệ thống tự động hóa công nghiệp.