Digital Object Identifier: 10.24054/rcta.v2i46.4141

Industrial control through reinforcement learning

Control industrial por aprendizaje reforzado

Ing. Yessica Cindy Vannesa Mora Cubides[1], **Ing. Daniel Steven Arias Otálora**[1], **PhD. José Antonio Tumialan Borja**[2], **PhD. Hugo Fernando Velasco Peña**[1]

[1] ***Institution 1,*** *Universidad de la Salle Mechatronics Engineering Program, Industrial Process Automation Group*
[2] ***Institution 2,*** *Universidade São Paulo-USP, Escola de Engenharia de São Carlos, Laboratório de Escoamentos Multifásicos Industriais.*

Correspondence: *jtumialan@usp.br*

Received: *april 24, 2025.* ***Accepted:*** *july 03, 2025.* ***Published:*** *august 09, 2025.*

How to cite: *Y. C. Mora Cubides, D. S. Arias Otálora, J. A. Tumialan Borja, and H. F. Velasco Peña, "Industrial control through reinforcement learning", RCTA, vol. 2, no. 46, pp. 201–208, Aug. 2025.*
Recovered from https://ojs.unipamplona.edu.co/index.php/rcta/article/view/4141

Abstract: In this article, the implementation of the flow control technique with artificial intelligence (DDPG) is presented in a fully instrumented functional prototype with industrial sensors and actuators, simulating flow recirculation through three tanks. The methodology used for process identification (first-order plus dead time model (FOPDT)) through ClientServer OPC communication with Matlab® is presented. The design of the reinforcement learning algorithm and its adaptation in the learning environment with experimental data are also presented. The simulation results were satisfactory compared to traditional control techniques, demonstrating robustness against forced disturbances. Finally, the implementation of reinforced learning control integrating TIA Portal and Matlab (through a PLC-S7-1500 controller) was evaluated with a reference of 600 l/h, achieving 0% overshoot with a settling time of 22s. Compared to other control systems, a better response in settling time and overshoot-free control was observed. Finally, perturbations were applied to the system, observing their effect in relation to the flow.

Keywords: reinforcement learning, efficiency, artificial intelligence, industrial processes.

Resumen: En este artículo se presenta la implementación de la técnica de control de caudal con inteligencia artificial (DDPG) en un prototipo funcional completamente instrumentado con sensores y actuadores industriales, simulando la recirculación de caudal a través de tres tanques. Se presenta la metodología utilizada para la identificación del proceso (modelo de primer orden más tiempo muerto (FOPDT)) mediante comunicación OPC Cliente Servidor con Matlab®. También se presenta el diseño del algoritmo de aprendizaje por refuerzo y su adaptación en el entorno de aprendizaje con datos experimentales. Los resultados de la simulación fueron satisfactorios en comparación con las técnicas de control tradicionales, demostrando robustez frente a perturbaciones forzadas. Finalmente, se evaluó la implementación del control de aprendizaje reforzado integrando TIA Portal y Matlab (a través de un controlador PLC-S7-1500) con una referencia de 600 l/h, logrando un sobre impulso del 0% con un tiempo de asentamiento de 22s. Comparado con otros sistemas de control, se observó una mejor respuesta en el tiempo de asentamiento y un control libre de sobre impulso. Finalmente, se aplicaron perturbaciones al sistema, observando su efecto con relación al flujo.

Palabras clave: aprendizaje reforzado, eficiencia, inteligencia artificial, proceso industrial.

1. INTRODUCTION

Reinforcement learning is a technique in which an agent learns to perform a task through repeated trial-and-error interactions within a dynamic environment. The core of this technique involves incorporating behaviors by interacting with the environment, without explicitly programming the solution of the problem [1]. Based on this concept, multiple uses of these learning environments have been generated, such as game playing or controls development for robotics [2], and even energy optimization in buildings [3].

In general, there are several techniques of artificial intelligence that are responsible for solving complex tasks based on unexplored and high-dimensional sensory data, making it a powerful tool for the development of complex tasks. Unlike other branches of AI, reinforcement learning receives delayed feedback, where the agent receives feedback after generating a decision and prediction [3]. Reinforcement learning has emerged as a very powerful approach for automated decision making in multiple control systems fields [4], representing a far-reaching and promising methodology.

In the field of process control, [5] presents a comparison between two Proportional-Integral (PI) controller tuning techniques: the traditional Zero-Pole cancellation method and an innovative strategy based on reinforcement learning for adaptively adjusting a PI controller in a refrigeration system (HVAC). The results demonstrated that the innovative method enables energy consumption optimization and reduces operating costs.

To address this issue more efficiently, new techniques are being explored, including those that leverage Artificial Intelligence (AI) and optimization. These strategies adopt dynamic and continuous approaches using tools such as neural networks [6]-[10], genetic algorithms [11]-[13], fuzzy logic [14][15], and other optimization methods [16][17]. The main goal of applying AI-based tuning techniques is to automate and refine the characteristic parameters of a PID controller, thereby achieving better performance than traditional methods.

While PID control remains widely used due to its simplicity, approaches such as reinforcement learning are not restricted to a single linear model of the system. These advanced techniques can be applied to more complex scenarios, as demonstrated by [18]. Furthermore, such strategies allow for the incorporation of user preferences into decision-making [19], energy optimization [20], and fault detection [21].

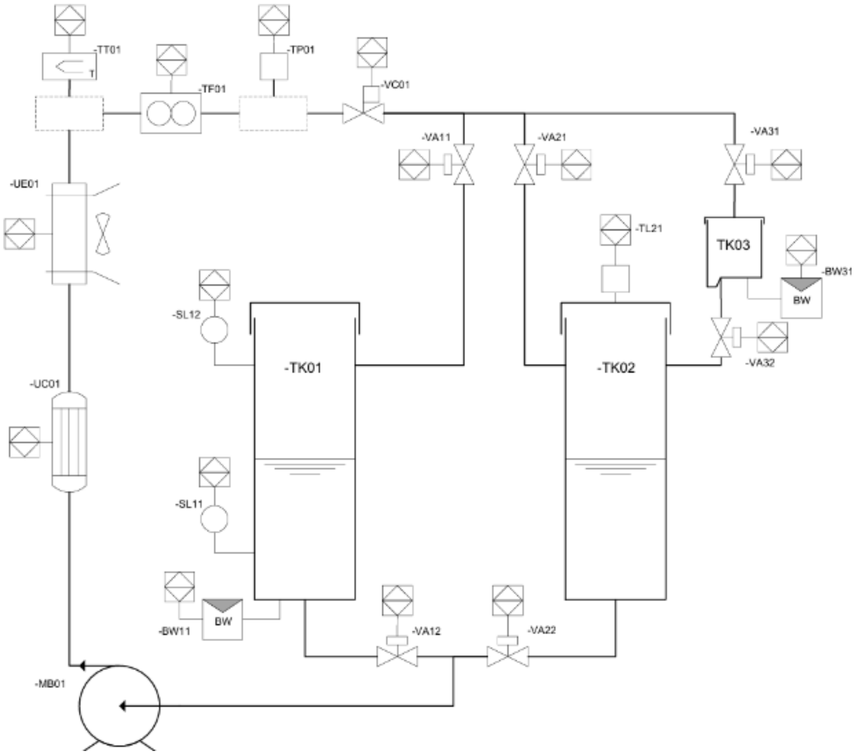

Fig. 1. *Diagram P&ID Training unit, [22].*

This article focuses on the development of a reinforcement learning-based controller for an industrial process training plant, using a learning environment with a mathematical model of the system. The objective is to maintain a constant flow in a system of three interconnected tanks, by adjusting the frequencies of a hydraulically controlled pump through a frequency converter and a Siemens S7-1500 PLC [22].

The type of reinforcement learning applied is DDPG (Deep Deterministic Policy Gradient), which is a critic-actor algorithm with a policy that maximizes long-term reward [23]. The following steps are performed during the agent's training stage:

In equation (1), the parameters ϕ are initialized, from the critic the observation S is taken, and perform the action A .

$$Q(S, A; \phi) \qquad (1)$$

Initially, the actor network (2) takes the observation S and returns the action that maximizes the long-term reward

$$\pi\ (S; \phi) \qquad (2)$$

For the current observation S (3), the action A is selected, where N represents the modeled noise

$$A = \pi(S; \phi) + N \qquad (3)$$

The action A is executed, and the reward R and the next observation S′ is calculated

The information (S, A, R, S') is stored in the experience buffer.

A mini-batch of experiences M is randomly generated from the experience buffer (S_i, A_i, R_i, S')

If S_i is a terminal state, the target value y_i for the value function is set to R_i; otherwise, it is set according to equation (4)

$$y_i = R_i + \gamma Q_t + (S_i{'}, \pi_t(S_i{'}; \theta_t); \phi_t \qquad (4)$$

The goal of the value function is to sum the immediate reward R_i with the discounted future reward. To calculate the cumulative reward, the agent first computes the next action and the next observation $S'_i$from the experience samples using the target actor. The agent estimates the cumulative reward using the next action for the target critic.
The critic's parameters are updated by minimizing the loss L over all experience samples, equation (5).

$$L\ = \frac{1}{2\mathrm{M}} \sum_{\mathrm{i=1}}^{\mathrm{M}} (y_i - Q(S_i, A_i;)\phi)^2 \qquad (5)$$

The actor's parameters are updated using the policy gradient, aiming to maximize the expected discounted reward, equation (6).

$$\begin{gathered} \nabla_{\theta} J \approx \frac{1}{M} \sum_{i=1}^{M} G_{ai} + G_{ai} \qquad (6) \\ G_{ai} = \nabla_A Q'(S_i, A; \phi) \text{ where } A = \pi(S_i; \theta) \\ G_{\pi i} = \nabla_{\theta} \pi(S_i; \theta) \end{gathered}$$

To update the parameter values of the critic and actor, smoothing factors τ are used (7).

$$\begin{gathered} \phi_t = \tau\phi + (1 - \tau)\phi_t\ (target\ critic\ parameters) \qquad (7) \\ \phi_t = \tau\phi + (1 - \tau)\theta_t\ (target\ actor\ parameters) \end{gathered}$$

2. WORK METODOLOGY

2.1 OPC communication

Through the OPC architecture and the KepServerEX server, client-server communication was established between the Siemens S7-1500 PLC and the MATLAB® platform, which allowed the acquisition, monitoring and control of process variables in real time, as shown in Figure 2. This integration facilitated the recording of the dynamic behavior of the flow rate with variations in the pump operating frequency. Based on the experimental data obtained, the dynamic model of the plant was identified, which was represented by a first-order model with dead time (FOPDT), commonly used in industrial control systems due to its simplicity and ability to approximate real processes.

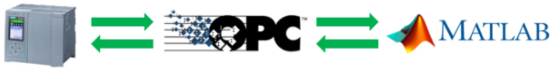

***Fig. 2**. Connection diagram with KepServer*

2.2 Mathematical Model

Using the response shown in Figure 3, data were obtained from the step response (Order to the drive: 46.875 Hz) and the response curve, Figure 4, to identify the mathematical model proposed by Alfaro [24].

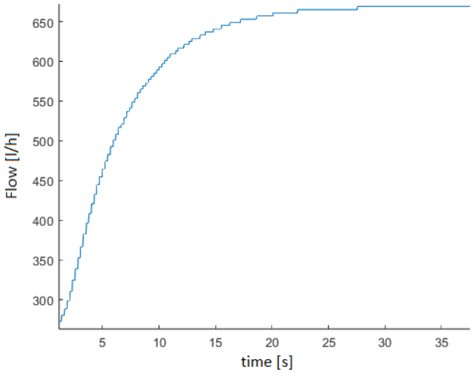

***Fig. 3**. Process curves diagram*

The obtained transfer function is presented in equation (8).

$$G_{p1}(S) = \frac{0.02575e^{-1.6s}}{5.17s+1} \qquad (8)$$

Figure 4 compares the plant’s time response with the transfer function given in equation (8). The average error of the models obtained was 0.30%.

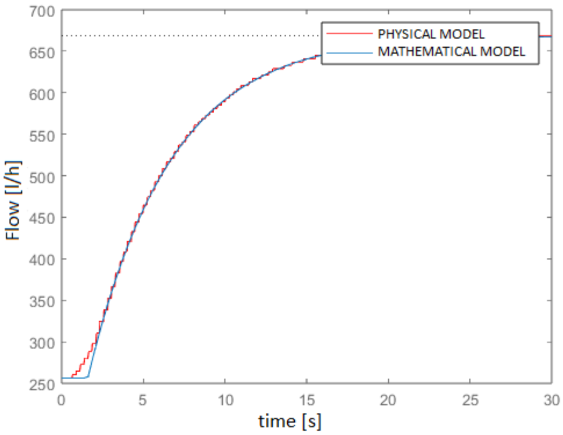

***Fig. 4**. Physical model vs. mathematical mode*

2.3 Environment Reward Table

The ranges and rewards provided by the environment are presented in Table I. The variation range of the flow through the tanks, 200-700 l/h, was used as the termination condition for the plant.

2.4 General Diagram of the Learning Environment

Simulink® was used to represent the model through block diagrams, as shown in Figure 5. The integrator H represents the initial condition of the hydraulic pump frequency.

2.5 General Diagram of the Learning Environment

The learning environment was constructed using an RL agent, implementing the reward table from Table 1 (calculated reward), the employed environment (mathematical model) and the termination conditions such as the physical limits of the plant and the target flow reference value (See Figure 6).

***Table 1:** environment reward table*

Reward	State
+10	For maintaining the value within a variation less than 1 l/h
-1	For maintaining the flow value within the operational limits
-650	For allowing the flow to exceed the limits

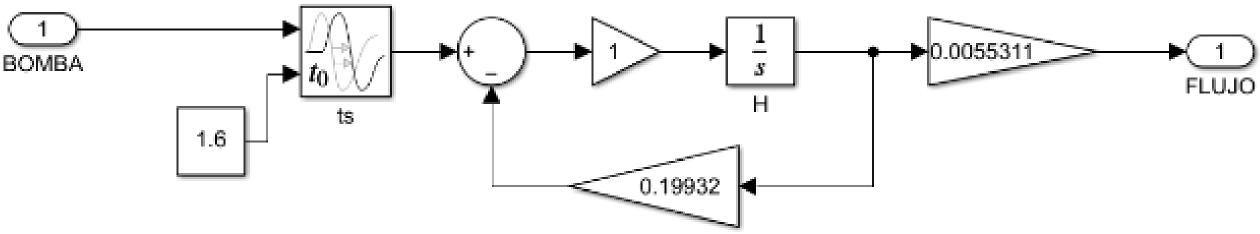

***Fig. 5**. Block diagram of the mathematical model*

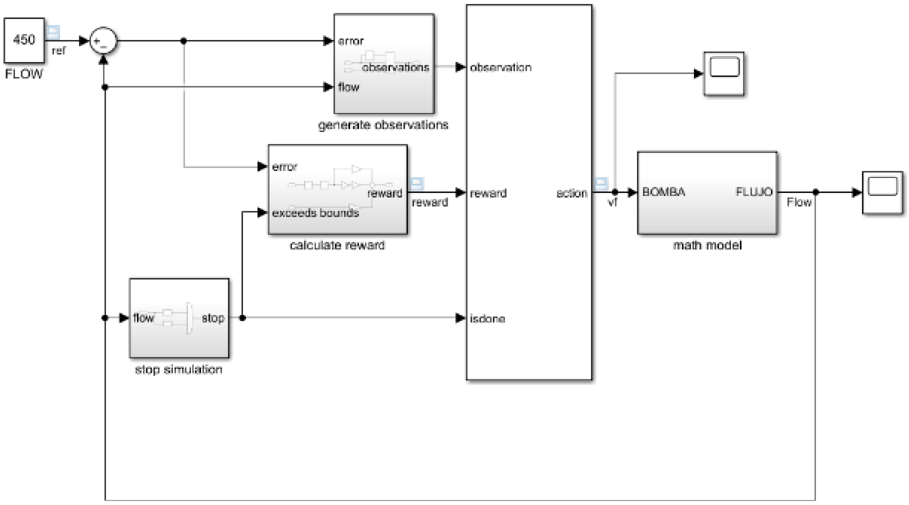

Fig. 6. *Block diagram of the learning environment*

2.6 Learning Environment Programming

The Algorithm 1 and Table 2 summarizes the key parameters used to configure the learning environment. Based on these, the learning model was developed.

The training results are presented graphically in Figure 7, with a total of 880 training epochs, a reward exceeding 600, and a training time of 2 hours and 12 minutes.

Algorithm 1 Learning Environment

```
$RLAgent \leftarrow GetAgentCharacteristics$
$RLAgent \leftarrow InitializeLearningEnvironment$
$InitializeParameters \leftarrow (Flow = 0, Pump = 0)$
while $Flow < 200 || Flow > 700$ do
    $Rand \leftarrow GenerateRandomNumbersBetween(A, B)$
    $Flow \leftarrow Rand(200, 700)$
end while
while $Pump < 7000 || Pump > 25000$ do
    $Rand \leftarrow GenerateRandomNumbersBetween(A, B)$
    $Pump \leftarrow Rand(7000, 25000)$
end while
$RLAgent \leftarrow TrainAgent$ =0
```

***Table 2:** learning environment characteristics*

Criterion	Value
Approximators used by the policy	Critic, Actor
Sampling time	1s
Simulation time	200s
Number of neurons in critic	25
Number of neurons in actor	25
Type of agent used for learning	Deep Deterministic Policy Gradient (DDPG)
Objective factor	1.00E-03
Critic learning factor	1.00E-03
Actor learning factor	1.00E-04
Gradient threshold (Critic, Actor)	1.0
Agent experience buffer	1.00E+06
Agent batch size	6.40

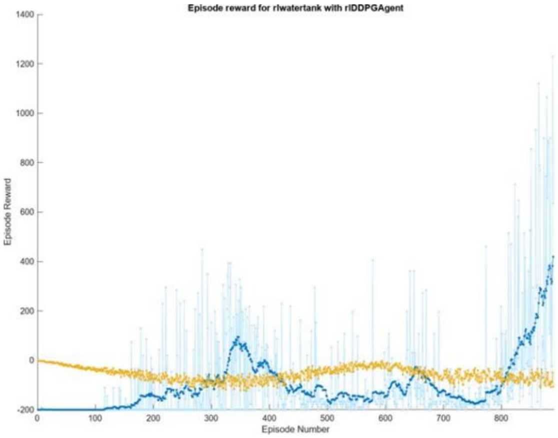

Fig. 7. *Reward diagram by episode (Episode reward in blue ans Value of the Target Critc in yellow)*

3. IMPLEMENTATION OF THE PROTOTYPE

The communication between the trained agent and the physical plant (Siemens S7-1500 PLC) was established through KepServer (Figure 8).

3.1 Reinforcement Learning Controller Analysis

The controller response was analyzed for two reference points (350 and 600 l/h), and its behavior was evaluated under artificially forced disturbances (opening and closing of the plant's solenoid valves), generating abrupt variations in the system flow rate. This was done to assess the controller's robustness to unscheduled changes in operating conditions. Satisfactory responses were obtained, as shown in Figure 9.

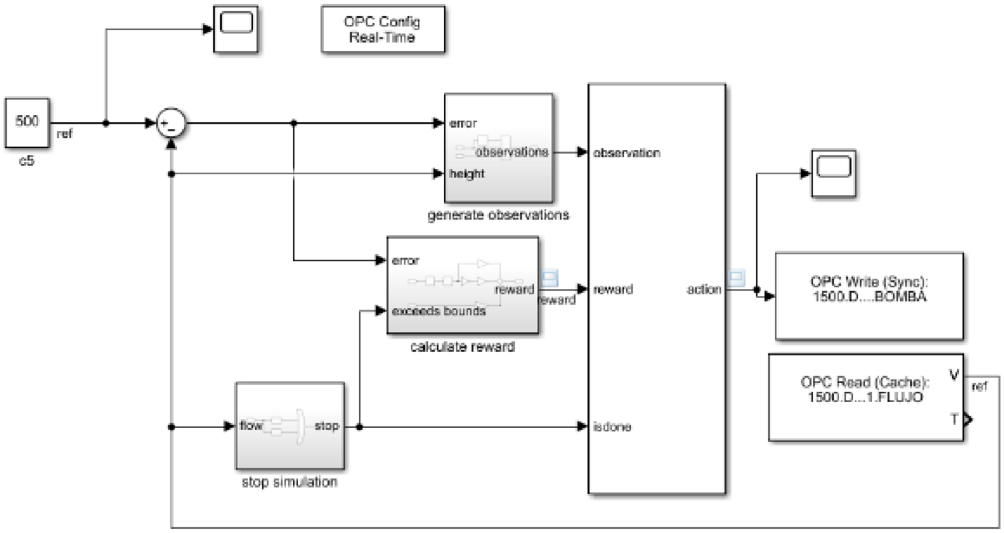

***Fig. 8**. OPC connection block diagram*

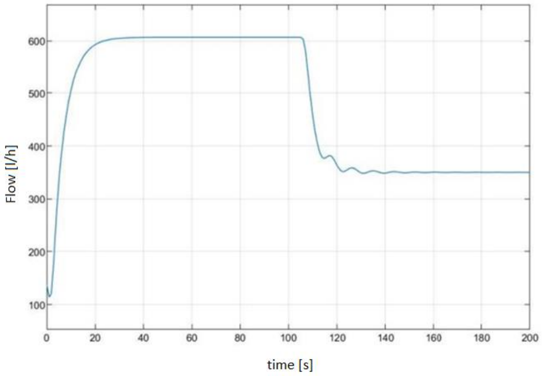

***Fig. 9**. Plant response to the proposed control system with two references values*

From Figure 9, the settling time and overshoot were calculated for a reference of 600 l/h, resulting in 22 seconds and 0% respectively. For a reference of 350 l/h, a settling time of 19 seconds and 0% overshoot were obtained.

3.2 Comparison with other control techniques

Reinforcement learning control was compared with other control techniques under the same operating conditions (PI control, cascade control, Smith predictor) in Table 3, using the same reference value of 600 l/h.

***Table 3:** comparison table of different controllers in the physical plant*

Control Structure	Settling Time	Overshoot
PI Control	35s	15%
Cascade Control	32s	0%
Smith Predictor	25s	5%
Reinforcement Learning	22s	0%

From Figure 3, it can be observed that reinforcement learning control does not exhibit overshoot in the rising edge and small oscillation without overshoot in the falling edge, comparing with classical control methods. The Smith predictor shows a 5% overshoot with a settling time of 25 seconds, the PI control has a higher overshoot of 15% with a settling time of 35 seconds, and the cascade control exhibits a similar behavior to reinforcement learning, with no overshoot but a longer settling time by 10 seconds. Based on this analysis, it was validated that reinforcement learning control generates a better response in terms of response speed and overshoot compared to classical controllers.

4. CONCLUSIONS

The implementation of the reinforcement learning controller showed satisfactory results evaluated in a functional prototype. Additionally, it demonstrated robustness to forced disturbances while maintaining a constant reference point with a steady state error of 0.2%.

In comparison to other classical control systems, the reinforcement learning controller exhibited a shorter settling time of 22 seconds and no overshoot. It resembled the behavior of cascade control but improving the settling time.

REFERENCES

[1] Kaelbling, Leslie Pack; Littman, Michael L.; MOORE, Andrew W. Reinforcement learning: A survey. Journal of artificial intelligence research, 1996, vol. 4, p. 237-285

[2] Carlos G´omez, J., Verrastro, C. A. (s/f). Aprendizaje por refuerzo y control difuso para generar comportamiento de robots. Edu.ar. from 2022, de https: //www.f rba.utn.edu.ar/wp content/uploads/2021/02/jar8submission30.pdf

[3] Wang, Zhe; Hong, Tianzhen. Reinforcement learning for buildingcontrols: The opportunities and challenges. Applied Energy, 2020, vol.269, p. 115036.

[4] Damjanovi ´C, Ivana, et al. High Performance Computing Reinforcement Learning Framework for Power System Control. En 2023 IEEEPower Energy Society Innovative Smart Grid Technologies Conference (ISGT). IEEE, 2023. p. 1-5.

[5] V. Abad-Alcaraz, M. Castilla, J.D. Álvarez (2024) A Comparison of Classical and Reinforcement Learning-based Tuning Techniques for PI controllers, IFAC-papers OnLine Volume 58, Issue 7, 2024, Pages 180-185. doi: 10.1016/j.ifacol.2024.08.031.

[6] Gunther, J., Reichensd¨orfer, E., Pilarski, P.M., and Diepold, K. (2020). Interpretable PID parameter tuning for control engineering using general dynamic neural networks: An extensive comparison. Plos One, 15(12), e0243320.

[7] Ali, M., Firdaus, A.A., Arof, H., Nurohmah, H., Suyono, H., Putra, D.F.U., and Muslim, M.A. (2021). The comparison of dual axis photovoltaic tracking system using artificial intelligence techniques. IAES Int. J. Artif. Intell, 10(4), 901.

[8] Daye Yang, Jingcheng Wang, Huihuang Cai, Jun Rao, Chengtian Cui, (2025) Intelligent control strategy for electrified pressure-swing distillation processes using artificial neural networks-based composition controllers, Separation and Purification Technology, Volume 360, Part 2, 2025, 130991, ISSN 1383-5866.

[9] Ibtihaj Khurram Faridi, Evangelos Tsotsas, Wolfram Heineken, Marcus Koegler, Abdolreza Kharaghani, Development of a neural network model predictive controller for the fluidized bed biomass gasification process, Chemical Engineering Science, Volume 293, 2024, 120000, ISSN 0009-2509.

[10] Hong-Gui Han, Lu Zhang, Hong-Xu Liu, Jun-Fei Qiao, Multiobjective design of fuzzy neural network controller for wastewater treatment process, Applied Soft Computing, Volume 67, 2018, Pages 467-478, ISSN 1568-4946.

[11] P. Siva Krishna, P.V. Gopi Krishna Rao, Fractional-order PID controller for blood pressure regulation using genetic algorithm, Biomedical Signal Processing and Control, Volume 88, Part B, 2024, 105564, ISSN 1746-8094.

[12] Marco Paz Ramos, Axel Busboom, Andriy Slobodyan, Genetic PI Controller Tuning to Emulate a Pole Assignment Design∗∗This work was supported by the Bavarian Ministry of Economic Affairs, Regional Development and Energy (StMWi) under Grant DIK0397/03., IFAC-PapersOnLine, Volume 58, Issue 7, 2024, Pages 138-143, ISSN 2405-8963.

[13] K.G Aparna, R. Swarnalatha, Dynamic optimization of a wastewater treatment process for sustainable operation using multi-objective genetic algorithm and non-dominated sorting cuckoo search algorithm, Journal of Water Process Engineering, Volume 53, 2023, 103775, ISSN 2214-7144

[14] Himanshukumar R. Patel, Optimal intelligent fuzzy TID controller for an uncertain level process with actuator and system faults: Population-based metaheuristic approach, Franklin Open, Volume 4, 2023, 100038, ISSN 2773-1863.

[15] Slavica Prvulovic, Predrag Mosorinski, Dragica Radosav, Jasna Tolmac, Milica Josimovic, Vladimir Sinik, Determination of the temperature in the cutting zone while processing machine plastic using fuzzy-logic controller (FLC), Ain Shams Engineering Journal, Volume 13, Issue 3,2022, 101624, ISSN 2090-4479.

[16] Rios H. R, Apráez B. A., Rodriguez C. J and Tumialan. B. J. A., PI tuning based on Bacterial Foraging Algorithm for flow control, (2020) IX International Congress of Mechatronics Engineering and Automation (CIIMA), Cartagena, Colombia, 2020, pp. 1-6, doi: 10.1109/CIIMA50553.2020.9290289.

[17] Ali, M., Firdaus, A.A., Arof, H., Nurohmah, H., Suyono, H., Putra, D.F.U., and Muslim, M.A. (2021). The comparison of dual axis photovoltaic tracking system using artificial

intelligence techniques. IAES Int. J. Artif. Intell, 10(4), 901.

[18] Shuprajhaa, T., Sujit, S.K., and Srinivasan, K. (2022). Reinforcement learning based adaptive pid controller design for control of linear/nonlinear unstable processes. Applied Soft Computing, 128, 109450.

[19] Lei, Y., Zhan, S., Ono, E., Peng, Y., Zhang, Z., Hasama, T., and Chong, A. (2022). A practical deep reinforcement learning framework for multivariate occupant centric control in buildings. Applied Energy, 324, 119742.

[20] Yu, L., Xie, W., Xie, D., Zou, Y., Zhang, D., Sun, Z., Zhang, L., Zhang, Y., and Jiang, T. (2019). Deep reinforcement learning for smart home energy management. IEEE Internet of Things Journal, 7(4), 2751–2762.

[21] Matetic, I., Stajduhar, I., Wolf, I., and Ljubic, S. (2023). Improving the efficiency of fan coil units in hotel buildings through deep-learning-based fault detection. Sensors, 23(15), 6717.

[22] Andina de Integración Tecnológica (ANITCO), Manual de operación y mantenimiento unidad de entrenamiento en automatización, 1ra ed. Bogotá, Colombia: ANITCO, 2016.

[23] Mathworks, Reinforcement Learning Toolbox™ User’s Guide, Matlab and Simulink, MathWorks, Natick, MA, USA, 2020. [En Línea] Disponible en: https://www.mathworks.com/help/reinforcement-learning/

[24] Alfaro, Víctor M. Método de identificación de modelos de orden reducido de tres puntos 123c. Escuela de Ingeniería Eléctrica Universidad de Costa Rica, 2007, p. 1-7