Optimizing Industrial Control Systems: A Reinforcement Learning Approach with PPID and GMVC

Daniel Abreu Macedo da Silva
SENAI Getúlio Vargas
Belém, Brasil
daniel.silva@senaiPA.org.br

Antônio da Silva Silveira
Postgraduate Program in Electrical Engineering
Federal University of Pará
Belém, Brasil
asilveira@ufpa.br

Kalil Brito de Almeida
SENAI Getúlio Vargas
Belém, Brasil
kalil.almeida@aluno.senai.br

Matheus Morais da Silva
Postgraduate Program in Electrical Engineering
Federal University of Pará
Belém, Brasil
matheus.morais@itec.ufpa.br

***Abstract*—This paper explores the use of Reinforcement Learning (RL) to autonomously tune controllers that depend on a single adjustable parameter. The proposed approach utilizes system performance and robustness indicators to optimize controller settings, thereby ensuring high efficiency and adaptability. The method was initially validated using a Tacho Generator Motor (TGM) setup to refine various control strategies, such as the Pseudo Proportional Integral Derivative (PPID) Controller and the Generalized Minimum Variance Control (GMVC) approach. Subsequently, the approach was extended to a realistic industrial setup involving a level control system using a frequency inverter-powered pump, with a PLC (Schneider M221) and an ultrasonic sensor. This demonstrated the method's applicability in controlling active industrial systems and optimizing online control via MATLAB. Both tests underscored the potential of RL-tuned controllers to achieve a favorable trade-off between performance and robustness, minimizing tracking errors while ensuring stability in diverse applications.**

***Index Terms*—Generalized Minimum Variance Control; Level Control System; Pseudo Proportional Integral Derivative Controller; Reinforcement Learning; Tacho Generator Motor.**

I. Introduction

Modern industrial production processes are increasingly complex, demanding advanced control techniques that ensure both robustness and optimization. The ability to fine-tune control parameters efficiently is critical for improving system performance while maintaining stability. Among various control strategies, model-based approaches have gained prominence due to their ability to predict system behavior and enhance accuracy [21]. However, as the complexity of processes increases, traditional tuning methods may become insufficient, necessitating more adaptive and intelligent optimization techniques.

Despite significant advancements in control theory, Proportional Integral Derivative (PID) controllers remain the dominant choice in industrial applications due to their simplicity, efficiency, and versatility [5]. However, real-world processes often exhibit high-order dynamics, delays, and nonlinearities, making manual PID tuning challenging, especially in varying operating conditions [18]. To address these challenges, alternative controller topologies such as the Pseudo Proportional Integral Derivative (PPID) Controller and Generalized Minimum Variance Control (GMVC) have been proposed. These approaches offer improved robustness and adaptability with relatively simple parameterization, making them suitable candidates for automated tuning.

A major challenge in control system design is achieving the right balance between performance and robustness. Controllers must accurately track reference signals while remaining resilient to disturbances and process uncertainties. Traditional tuning methods often rely on extensive empirical adjustments, which can be time-consuming, inefficient, and suboptimal.

In modern industry, the move towards automation, digitalization, and connected systems (known as Industry 4.0) has made adaptability and auto-tuning of controllers not only desirable but often necessary. In such environments, the time and cost associated with manual tuning, coupled with the variability of industrial processes, highlight the need for optimization solutions that can be implemented by non-specialists, with minimum process downtime, and are scalable or generalizable to various scenarios. Reinforcement Learning (RL) provides a powerful and flexible framework in this context: it enables autonomous, data-driven search for optimal or near-optimal parameters, adaptation to process or environment changes in real time, and integration with smart devices and cloud infrastructures, allowing for distributed and collaborative control tuning [32].

To address these challenges, this paper investigates the application of Reinforcement Learning (RL) for automatic controller tuning. The proposed approach was initially validated on a Tacho Generator Motor (TGM) system, modeled using the Least Squares (LS) identification method. Following this validation, the methodology was extended to an industrial

The National Industrial Apprenticeship Service (SENAI), the National Council for Scientific and Technological Development (CNPq) and the Brazilian Federal Agency for Support and Evaluation of Graduate Education (CAPES).

setup consisting of an M221 PLC, a frequency inverter, and an ultrasonic sensor, also identified using the LS method. Two single-parameter controllers—Proportional-Plus-Integral-Derivative (PPID) and Generalized Minimum Variance Control (GMVC)—were automatically tuned using RL to enhance tracking accuracy and disturbance rejection while maintaining system stability. The approach was further applied to a level control system driven by a frequency inverter-powered pump, demonstrating the versatility of RL across diverse control architectures. The RL-based tuning process achieved an effective balance between performance and robustness without the need for manual adjustments, highlighting its potential as a practical solution for automated controller optimization in industrial environments.

This work is organized as follows: The second section discusses the process composition and model achievements. In the third section, the applied controllers topologies are presented. Robustness and performance indexes, as well as the RL method used, are presented in the fourth section. The results are analyzed in the fifth section, followed by conclusions in the final section.

II. Process Composition

A. Tacho Generator Motor (TGM) Setup

The TGM is a system that converts the speed generated by a motor into voltage via a coupling connected to a generator that generates electrical energy through mechanical energy conversion. Furthermore, this model is used in industrial processes such as centrifugal pumps, conveyors, and liquid flow meters, among others. The TGM serves as a testbed to validate control strategies, as shown in Fig. 1, designed for process identification and control methods applications.

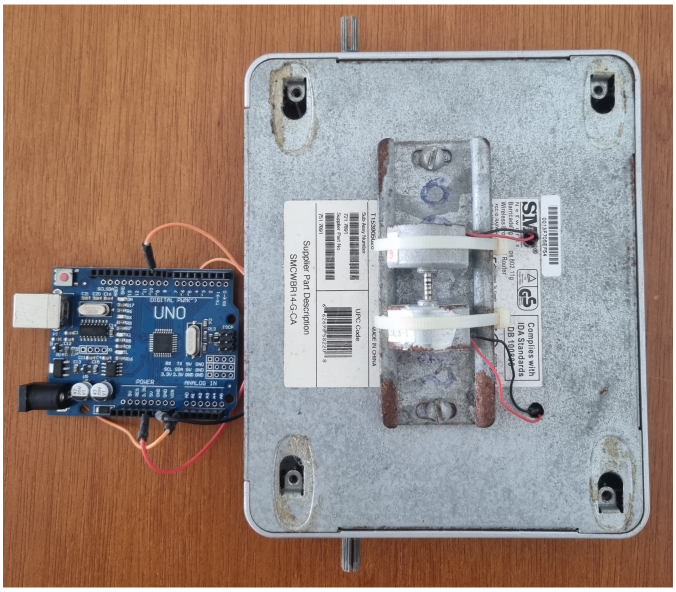

Fig. 1: TGM Didactic System.

The first motor's speed is converted into an analog signal by the designed TGM circuits, enabling the conversion of input signals from 0 V to 5 V (digital PWM signal) sent from an Arduino to the first motor. Additionally, the TGM can be controlled using potentiometers to alter the operational amplifiers' static gain. This allows for the application of plant identification techniques based on the schematic circuit shown in Fig. 2.

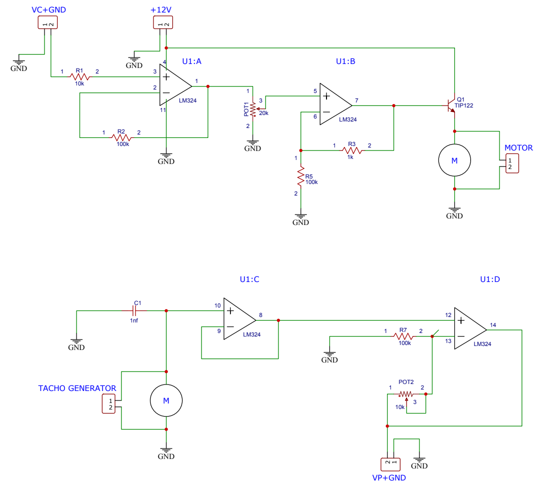

Fig. 2: TGM Schematic Circuit.

Two identical motors ensure consistent functionality across various operational points without significantly altering energy conversion. Potentiometers and the LM324 integrated circuit control the set-point, making it suitable for testing multiple motor specifications and yielding significant changes in response at the system output.

B. Industrial Level Control System

The industrial level control system is designed as a comprehensive implementation model that exemplifies practical applications in industrial environments. This setup features a frequency inverter-powered pump controlled by a Schneider M221 PLC, tasked with maintaining the water level in a tank using data from an ultrasonic sensor. This sensor, operating within a range of 0 to 60 cm (0 to 20 l) and providing an output signal of 0-10 V, facilitates precise level monitoring, a crucial aspect of industrial control systems [14].

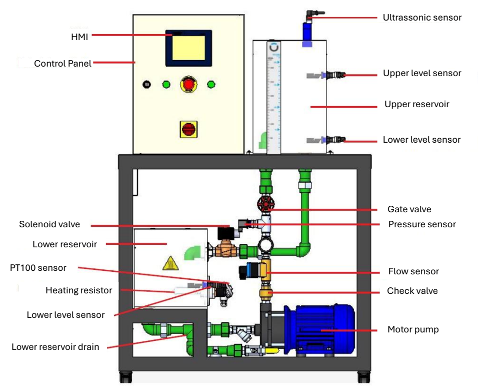

Fig. 3: The Level Control System Bench Setup illustrates the physical assembly of the industrial level control applications.

Figure 3 displays the bench setup for the level control system, showing all primary components including the tank, sensor setup, and pump, which combined facilitate practical learning and experimentation with control strategies. The PLC manages the task of modulating the frequency inverter to continuously adapt the pump speed, ensuring optimal system responses based on real-time feedback from the ultrasonic sensor [33].

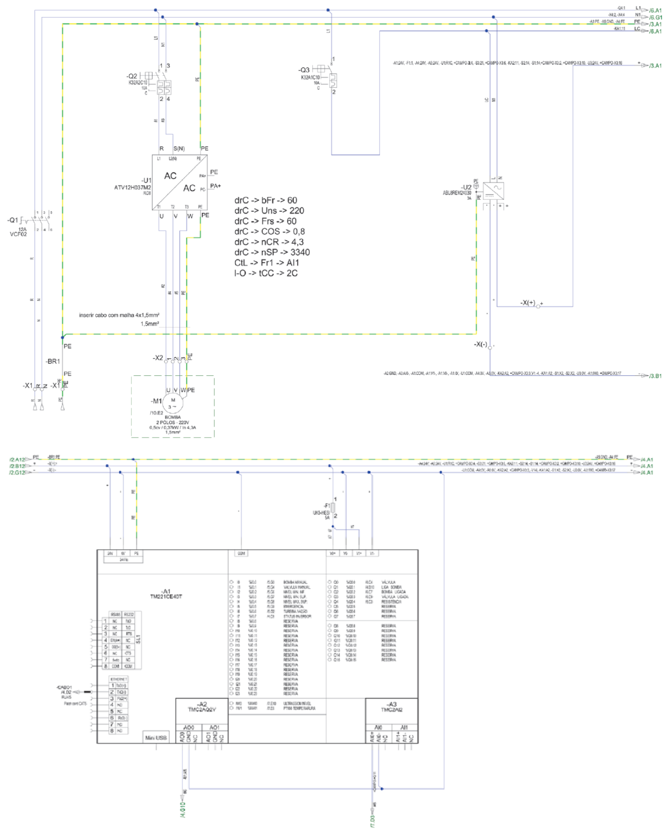

Fig. 4: System Diagram of the Level Control Setup provides a schematic representation of all interconnected components.

The schematic representation in Figure 4 highlights the workflow architecture of the system, showing the integration between the PLC, frequency inverter, pump, and sensor network, which underscores the system's ability to perform under varying operational conditions typical in industries such as wastewater management and chemical processing [2].

The real-time interfacing is carried out through an Arduino UNO connected to MATLAB, leveraging MATLAB's advanced computational capabilities to dynamically compute and refine the control law. In this setup, the sensor signal is first acquired by the PLC via its analog input. The PLC then forwards this signal through its analog output to the Arduino, which is interfaced with MATLAB running the reinforcement learning-based control algorithm. MATLAB processes the data in real time and computes the control action. This control signal is output from the Arduino as a PWM signal, which is internally converted back into an analog signal by reading it through one of the Arduino's analog inputs. The resulting analog control signal is then transmitted to the frequency inverter, which modulates the speed of the pump motor accordingly. This configuration creates a closed-loop system that enables real-time implementation, testing, and validation of advanced control strategies within a realistic industrial environment.

This setup enables a rigorous exploration of the RL-based control methodology's applicability in industrial settings, demonstrating how modern data-driven control strategies can substantially improve process performance and operational efficiency compared to conventional tuning approaches [11]. The synergy between advanced hardware and software shows this model's potential across various industrial scenarios, demonstrating its application viability and setting a solid foundation for future adaptations and enhancements.

C. Least Squares Method

For practical reasons, the Least Squares (LS) method was employed for system identification. The controllers were developed using a decentralized approach to manage the Single Input-Single Output (SISO) process, as applied in this study [2]. The SISO system identification utilizes the Auto Regressive with eXogenous inputs (ARX) model, given that the process involves one input ($u(k)$: the voltage signal applied to the first motor) and one output ($y(k)$: the voltage generated by the second motor). The relationship between the input and output signals can be expressed as follows:

$$y(k) = \frac{B(z)z^{-1}}{A(z)} u(k) \tag{1}$$

where $B(z)$ and $A(z)$ are, respectively, the z domain zeros and poles polynomials of the system.

Due to the system having an underdamped dynamic, the estimated discrete model is considered second order, (1) can be represented as a difference equation, as:

$$\begin{aligned} y(k) \quad = \quad & -a_1 y(k-1) - a_2 y(k-2) \\ & + b_0 u(k-1) + b_1 u(k-2) \end{aligned} \tag{2}$$

According to [2], Least Squares estimator is designed taking into account two factors: knowledge of the dynamics of the process, and according to the value of the squared Pearson correlation coefficient (R^2):

$$R^2 = 1 - \frac{\sum_{k=1}^{N} \left[y(k) - \hat{y}(k) \right]^2}{\sum_{k=1}^{N} \left[y(k) - \bar{y} \right]^2} \tag{3}$$

where $\hat{y}(k)$, $\bar{y}$ and N correspond to estimated output, average output and number of samples, respectively. According to [2], for many practical applications, values of R^2 between 0.8 and 1.0 can be considered sufficient. After identification via LS, a value of $R^2 = 0.94$ was obtained. For this reason, this aproach has been selected to identify the system model.

Thus, as presented in [27], using (2), the vector containing the read data (measures vector – $\boldsymbol{y}$), presented in (4), the matrix encompassing inputs and output data of the system (matrix of regressors – Φ), presented in (5), and the vector of estimated parameters (θ), presented in (6), may be determined.

$$\boldsymbol{y^T} = \begin{bmatrix} y(1) & y(2) & \dots & y(N) \end{bmatrix} \tag{4}$$

$$\Phi = \begin{bmatrix} -y(1) & 0 & u(1) \\ -y(2) & -y(1) & u(2) \\ \vdots & \vdots & \vdots \\ -y(N-1) & -y(N-2) & u(N-1) \end{bmatrix} \tag{5}$$

$$\theta^T = \begin{bmatrix} a_1 & a_2 & b_0 & b_1 \end{bmatrix} \tag{6}$$

After defining (4), (5) and (6), the following algebraic equation appears:

$$\boldsymbol{y} = \Phi \theta \tag{7}$$

According to [2], [28], to calculate θ using (7), it will be necessary that Φ is a square matrix, however Φ is a matrix of order $\Phi_{N,6}$. Thus, according to [22], it is necessary to apply the pseudo-inverse matrix. As a result, the solution of non-recursive least squares estimator was determined by computing θ as (8).

$$\theta = [\Phi^T \Phi]^{-1} \Phi^T \boldsymbol{y} \tag{8}$$

III. Control Theory

A. Pseudo Proportional Integral Derivative Controller

PID controller has different ways of being implemented because each manufacturer has its own type of adjustment, topology and filtering. The general control law of a discrete PID controller can be expressed by (9).

$$u(t) = K_c \left\{ e(k) + \frac{T_s}{T_i} \sum_{i=1}^{t} e(i) + \frac{T_d}{T_s}[e(k) - e(k-1)] \right\} \tag{9}$$

where $e(k) = y_r(k) - y(k)$ is the system error, K_c is the proportional gain, T_i is the integral time, T_d is the derivative time and T_s is the sampling period. The implementation of the incremental PID controller is given by

$$\begin{aligned} u(k) &= u(k-1) + K_c\{e(k) - e(k-1) + \tfrac{T_s}{T_i} e(k) + \\ &\quad \tfrac{T_d}{T_s}[e(k) - 2e(k-1) + e(k-2)]\} \end{aligned} \tag{10}$$

Equation 10, which is appropriate to microcontrollers applications, is present in single-loops and is understandable for digital implementation from viewpoints of operators and engineers [7]. As this controller has the peculiarity of having its proportional and derivative portions multiplied by the error signal, this implies that any considerable variation in the reference or in the error will cause a control action with high magnitude. To avoid practical problems, including loop saturation, it is kept the integral term with $e(k) = y_r(k) - y(k)$ and is substituted with the proportional and derivative terms the error with $e(k) = - y(k)$ [5]. Thus, the ideal digital PID can be rewritten as the I+PD in (11).

$$\begin{aligned} u(k) &= u(k-1) + K_c\{-y(k) + y(k-1) + \\ &\quad \tfrac{T_s}{T_i} e(k) + \tfrac{T_d}{T_s}\left[2y(k-1) - y(k) - y(k-2)\right]\} \end{aligned} \tag{11}$$

The role of the PPID is to promote stability and closed-loop performance with a simple calibration using a single parameter. Based on the relationship of [7], [26] is possible to set (12)

$$\frac{T_s}{T_i} > \frac{1}{100} \quad ; \quad T_i = [2...5]T_d \tag{12}$$

From the equations 11 and 12 it is possible to obtain the following normalized expressions:

$$\frac{T_s}{T_d} = 0.4 \;\; ; \;\; \frac{T_i}{T_d} = 4 \;\; ; \;\; \frac{T_s}{T_i} = 0.1 \tag{13}$$

After all these adaptations, the expression that defines the PPID is:

$$\begin{aligned} u(k) &= u(k-1) + K_c\{0.1y_r(k) - 3.6y(k) + \\ &\quad 6y(k-1) - 2.5y(k-2)\} \end{aligned} \tag{14}$$

The PPID controller has the following characteristics: it has only one parameter, Kc, to be tuned; can be applied to simple and complex plants (as long as they are non-linear); and its approval allows it to be integrated into digital technologies.

B. Generalized Minimum Variance Control

Another controller that will be applied is the incremental GMVC, using the ARIX (Auto-Regressive Integrated with eXogenous Inputs) model. The objective of this controller is to determine the control action u that minimizes the cost function $J = E[\phi(k+d)]$, where the output can be determined by (15).

$$\phi(k+d) = P(z)y(k+d) - T(z)y_r(k+d) + Q(z)\Delta u(k) \tag{15}$$

the Diophantine equation for the ARIX model, is expressed by (16).

$$1 = \Delta A(z)E(z) + Fz^{-1} \tag{16}$$

the polynomial $E(z)$ has value of $E(z) = 1$, and the order of the polynomial $F(z)$ changes, because $n_f = n_{\Delta a} - 1$, therefore:

$$E(z) = 1 \quad \text{and} \quad F(z) = f_0 + f_1 z^{-1} \tag{17}$$

When solving the equation 16, the F polynom can assume the values presented in (18).

$$f_0 = 1 - a_1; \quad f_1 = a_1 - a_2 \quad \text{and} \quad f_2 = a_2 \tag{18}$$

Based on [1], the control law of GMVC can be expressed as in (19).

$$\begin{aligned} \Delta u(k) &= \tfrac{1}{b_0 + q_0}[y_r(k+1) \\ &\quad -f_0 y(k) - f_1 y(k-1) \\ &\quad -f2y(k-2) - b_1 \Delta u(k-1)] \end{aligned} \tag{19}$$

Being that $\Delta u(k) = u(k) - u(k-1)$, so:

$$u(k) = u(k-1) + \Delta u(k) \tag{20}$$

IV. Performance and Robustness analysis

A. Performance analysis

The trade-off between performance and robustness is a key issue in control design [19]. Therefore, the use of a quantitative measure that evaluates the implemented controller is interesting, since when these performance indicators are minimized, the control system is considered effective or performing within the desired standards, and these are chosen with an emphasis on the specifications considered important to the system [30].

Integral Square Error (ISE) and Integral Squared control signal (ISU) are two examples of performance indexes that can measure the efficiency of a controller, those are used in discrete time domain [30]. ISE and ISU can be calculated with (21) and (22), respectively.

$$ISE = \frac{1}{N} \sum_{k=1}^{N} [e(k)]^2 \tag{21}$$

$$ISU = \frac{1}{N} \sum_{k=1}^{N} [u(k)]^2 \tag{22}$$

B. Robustness analysis

Robustness indexes are required to qualify the implemented controller as “optimal”. Among those indexes are Gain Margin (GM) and Phase Margin (PM), which are directly related to the robust stability of the process. The higher the values of these indexes, more robust (less sensitive to unwanted disturbances) the system is, on the other hand, the dynamic becomes slower [8], [30].

According to [1], the GM is defined as the required variation in the open-loop gain, necessary to make the system unstable, and the PM also provides a measure of the relative stability, indicating how much transport delay can be included in the feedback loop before instability to occur.

Other two variables are interesting to achieve the values of GM and PM on the controlled system, the Sensitivity function (S_{sen}) and the Complementary Sensitivity (T_{com}), presented in (23) and (24), respectively.

$$S_{sen}(z) = \frac{1}{1 + G_c(z)G_p(z)} \tag{23}$$

$$T_{com}(z) = \frac{G_c(z)G_p(z)}{1 + G_c(z)G_p(z)} \tag{24}$$

where $G_c(z)$ and $G_p(z)$ are, respectively, the controller and the process discrete transfer functions.

The S_{sen} characterizes the effect of a external disturbance acting on the output of the control loop, therefore indicates how the closed-loop system is sensitivity to process changes, while T_{com} is the Closed Loop Transfer Function (CLTF) for set-point changes [6], [11], [30], [33].

The maximum values of the amplitude ratio of $S_{sen}(z)$ and $T_{com}(z)$ for all frequencies, respectively, M_S and M_T (known as resonant peak), provides useful robustness measures and also shows a control system design criterion skog. These functions can be described by (25) and (26).

$$M_S \triangleq max\left[S_{sen}(z)\right] \tag{25}$$

$$M_T \triangleq max\left[T_{com}(z)\right] \tag{26}$$

According to [33], with M_S and M_T is possible to achieve GM and PM, as (27) and (28), respectively. This mathematical relation is valid for all implemented controllers of the paper.

$$GM \geq max\left[\frac{M_S}{M_S - 1}, \frac{M_T + 1}{M_T}\right] \tag{27}$$

$$PM \geq max\left[2sin^{-1}\left(\frac{1}{2M_S}\right), 2sin^{-1}\left(\frac{1}{2M_T}\right)\right] \tag{28}$$

C. Reinforcement Learning Tuning Method

Reinforcement Learning (RL) is a machine learning framework in which an agent interacts with an environment by taking actions, receiving observations, and accruing rewards, with the objective of maximizing cumulative reward [32]. In the context of controller tuning, the “agent” is the optimization algorithm (or a supervisory script), the “action” is an update in the controller’s tuning parameter (e.g., Kc for PPID or $q0$ for GMVC), and the ”environment” consists of the closed-loop process simulation and/or experiment.

Unlike brute-force search or static optimization, RL enables sequential, data-driven refinement via a reward function customized to the control objectives (in this work, minimizing IAE and TVC, and constraining PM and GM).

The agent evaluates a candidate controller configuration by running a closed-loop simulation (or experiment), calculates the relevant performance and robustness indices, and compares them to target (threshold) values. The reward structure in this study penalizes violations and increments the parameter in the direction of improved performance, as outlined in Algorithm 1 and Algorithm 2.

Compared to simple grid search or exhaustive enumeration, even this simple model-free RL setup:

- Incorporates multiple objectives (performance and robustness) in the reward;
- Is automatizable and extensible to non-linear, multi-parameter, or time-varying cases;
- Allows the design of exploration/exploitation strategies, adaptive increments, and on-line updating;
- Enables integration with more sophisticated RL algorithms (*e.g.,* policy gradients, Q-learning, Bayesian optimization) as needed.

In this work, a sequential deterministic RL variant is used for clarity and repeatability, but this approach can be naturally generalized.

Algorithm 1 Performance Tuning

Initialize $\lambda = \lambda_0$
while True **do**
  Compute IAE and TVC
  if IAE $\leq$ Target Value **and** TVC $\leq$ Target Value **then**
    $\lambda = \lambda + 0.01$
  else if IAE $>$ Target Value **and** TVC $>$ Target Value **then**
    $\lambda = \lambda + 0.01$
  else if IAE $\leq$ Target Value **and** TVC $>$ Target Value **then**
    $\lambda = \lambda + 0.01$
  else if IAE $>$ Target Value **and** TVC $\leq$ Target Value **then**
    $\lambda = \lambda + 0.01$
  end if
  if IAE $\leq$ Target Value **and** TVC $\leq$ Target Value **then**
    break
  end if
end while=0

After achieving the desired performance, a second stage focuses on robustness. This demonstrates an RL-style two-stage reward design, favoring controllers that do not just perform well but are also robust to disturbances and uncertainties.

Overall, this methodology implements the essence of reinforcement learning in the context of engineering controller synthesis: iterative, feedback-driven optimization that directly encodes the control objectives into the reward structure, enabling flexible and adaptive tuning even under changing or uncertain plant dynamics.

Algorithm 2 Robustness Tuning

0: Initialize $\lambda = \lambda_{\text{performance}}$
0: **while** True **do**
0:   Compute PM and GM
0:   **if** PM $\geq$ Target Value **and** GM $\geq$ Target Value **then**
0:     $\lambda = \lambda - 0.001$
0:   **else if** PM $<$ Target Value **and** GM $<$ Target Value **then**
0:     $\lambda = \lambda - 0.001$
0:   **end if**
0:   **if** One robustness parameter meets the target and the other does not **then**
0:     Check for a 10% tolerance
0:   **end if**
0:   **if** PM $\geq$ Target Value **and** GM $\geq$ Target Value **then**
0:     **break**
0:   **end if**
0: **end while**=0

V. Results

A. Identification

To validate the proposed methodology, experiments were conducted using both setups. The TGM served as an initial validation environment, while the level control system provided a realistic industrial application. For the TGM, a step input of 3V was applied for 2s, with a sampling period of $T_s = 0.01s$, achieving $R^2 = 0.9421$. The RL algorithm identified optimal K_c (PPID) and q_0 (GMVC) parameters, performing robust tracking and disturbance rejection.

Furthermore, the frequency inverter was excited with an input voltage of 5 V, corresponding to 30 Hz—a value which stabilized the water level at 10 liters, matching the outflow rate through the solenoid valve. This excitation was maintained for 12 seconds and served to identify the level process dynamics, using $T_s = 0.5s$, allowing for the subsequent application of the controllers. In the level control setup, the pump maintained predefined liquid levels by responding to feedback from an ultrasonic sensor. The reinforcement learning (RL)-tuned controllers demonstrated strong performance in maintaining accurate level control and mitigating flow disturbance variability, with outputs adapting in real-time, achieving $R^2 = 0.9402$.

TGM and level process identification signals are shown in figure 5 and figure 6, respectivelly. Equations (29) and (30), shows the identified transfer functions in z domain for TGM and level process, respectivelly.

$$G(z) = \frac{(-0.2175 + 0.2964z^{-1})z^{-1}}{1 - 0.8076z^{-1} - 0.1224z^{-2}} \tag{29}$$

$$G(z) = \frac{(-0.3421 + 0.4723z^{-1})z^{-1}}{1 - 0.4175z^{-1} - 0.7889z^{-2}} \tag{30}$$

B. Control

For control simulation, a few premises have been assumed. The test on TGM have a duration of $10s$, with a sample time of $0.01s$. In this time, 3 reference signals were tracked, with amplitudes of $1V$, $3V$ and $2V$, respectively. The RL algorithm achieved for PPID: $\lambda = K_c = 0.93$, and for GMVC: $\lambda = q_0 = 1.72$, since the target values for ISE, ISU, PM and GM are,

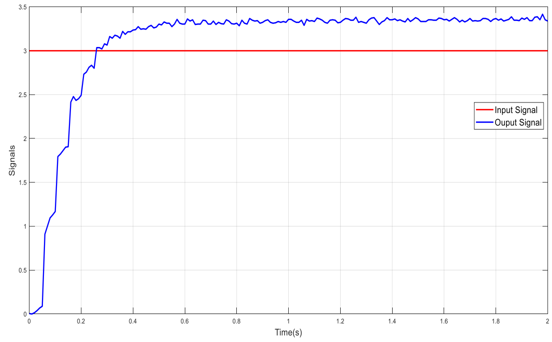

Fig. 5: TGM input and output signals obtained for identification.

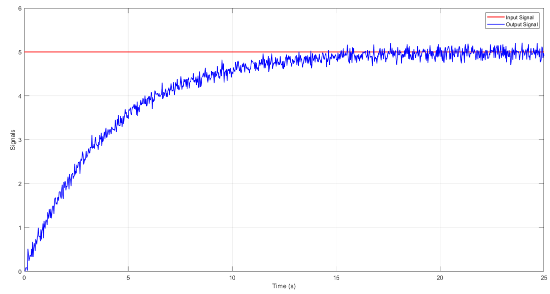

Fig. 6: Level system input and output signals obtained for identification.

respectively, 0.01, 4.5, 60 and 5 dB. The responses achieved with PPID and GMVC are shown, respectively, in Fig. 7 and Fig. 8.

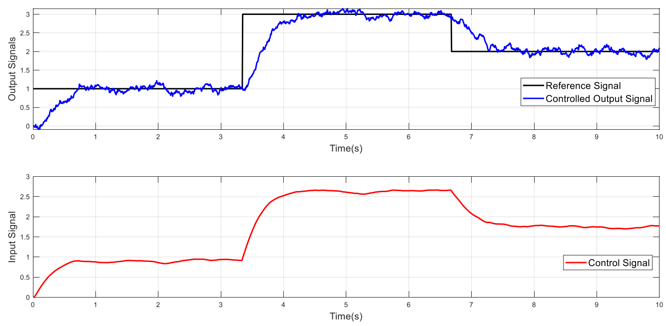

Fig. 7: TGM with PPID controller signals

The level process implementation has a duration of 60 s, with a sample time of 0.5 s. During this period, three reference signals were tracked, with amplitudes of (10 l), (12 l), and (8 l), respectively. The RL algorithm achieved for PPID $\lambda = Kc = 84.12$, and for GMVC $\lambda = q0 = 14.76$, since the target values for ISE, ISU, PM, and GM are the same as from the TGM test, but the dynamics of the process are different.

The responses achieved with PPID and GMVC are shown, respectively, in Fig 9 and Fig 10.

In Table 1, the performance and robustness indexes for both the PPID and GMVC controllers are displayed across two test processes: the TGM and Level Process. These values

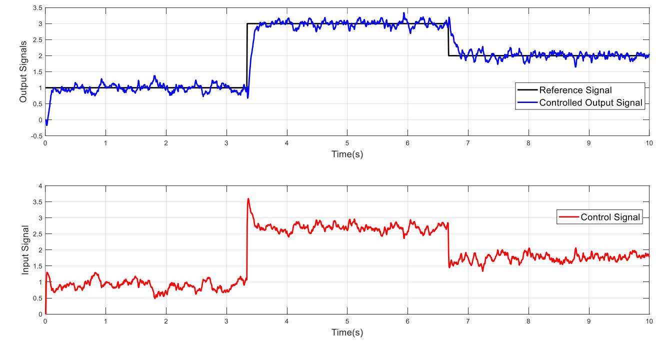

Fig. 8: TGM with GMVC controller signals.

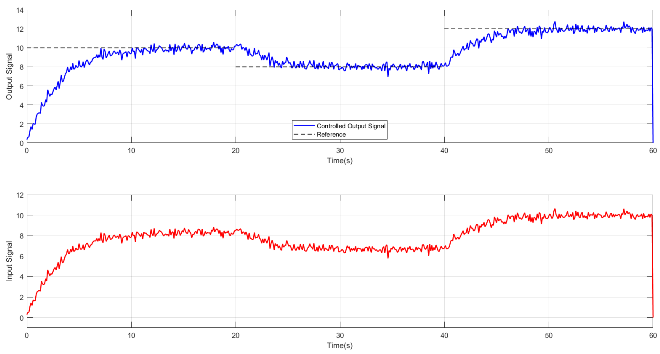

Fig. 9: Level process with PPID controller signals

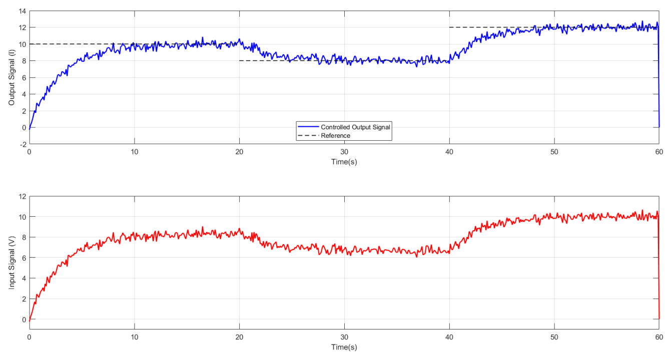

Fig. 10: Level process GMVC controller signals.

demonstrate the capabilities and resilience of each controller under varying conditions.

TABLE I: Performance and Robustness Indexes for TGM and Level Process

Process	Indexes	PPID	GMVC
TGM	ISE	0.0024	0.0052
	ISU	4.3380	3.7886
	GM	5.8591 dB	6.2442 dB
	PM	62.6446°	63.6542°
Level	ISE	0.0147	0.0248
	ISU	6.1234	4.4782
	GM	5.4123 dB	5.8471 dB
	PM	60.1014°	60.1472

VI. Conclusions

This work presented an innovative approach for optimizing industrial control systems using Reinforcement Learning (RL), focusing on the autonomous tuning of controllers characterized by a single adjustable parameter, such as the Pseudo Proportional Integral Derivative (PPID) Controller and the Generalized Minimum Variance Control (GMVC).

The experimental results demonstrated that both controllers, when tuned using the RL-based methodology, successfully met predefined performance and robustness criteria. Notably, they maintained accurate reference tracking and effective disturbance rejection in both the Tacho Generator Motor (TGM) testbed and the industrial-level control system. These findings underscore the robustness and adaptability of the proposed method across different control scenarios and dynamic processes.

This research highlighted the flexibility and practical benefits of using reinforcement learning for automatic controller tuning in complex industrial environments. By eliminating the need for manual adjustments and reducing the reliance on specialist knowledge, the methodology facilitates broader adoption of advanced control strategies, supporting the transition towards more intelligent and autonomous industrial systems aligned with Industry 4.0 standards.

Moreover, the study reinforced the relevance of computational intelligence in assisting control engineers, particularly in processes that are time-consuming or difficult to tune manually. The success of applying RL to both classical and predictive controllers confirms its versatility and generalization potential for a wide range of process control applications.

For future work, it would be valuable to expand the methodology to include a wider range of single-parameter controllers and evaluate the proposed approach in different industrial domains, such as aerospace systems or power electronics. Additionally, exploring reinforcement learning techniques for tuning multi-parameter controllers offers a promising avenue for advancing the field of adaptive control.

Acknowledgment

The authors wish to express their gratitude to SENAI, CNPQ and CAPES for their invaluable support and contributions to the development and success of this project.

References

[1] A. A. R. Coelho, D. C. Jeronymo, R. B. Araújo, *Sistemas dinâmicos: Controle clássico e preditivo discreto*. Editora UFSC, Florianópolis, 2019.

[2] A. A. R. Coelho and L. dos Santos Coelho, *Identificação de sistemas dinâmicos lineares*. Editora UFSC, Florianópolis, 2004.

[3] A. G. Bueno and R. A. Romano, "Filtro complementar aplicado a medida de inclinação de plataformas móveis," 2014.

[4] A. Simpkins, "System identification: Theory for the user, (ljung, l.; 1999)[on the shelf]," *IEEE Robotics & Automation Magazine*, vol. 19, no. 2, pp. 95–96, 2012.

[5] A. S. Silveira, A. A. R. Coelho, A. A. Franca, V. L. Knihs, "Pseudo-PID controller: design, tuning and applications," *IFAC Proceedings Volumes*, vol. 45, no. 3, pp. 542–547, 2012.

[6] A. S. Silveira, J. E. N. Rodríguez, A. A. R. Coelho, "Robust design of a 2-DOF GMV controller: A direct self-tuning and fuzzy scheduling approach," *ISA Transactions*, vol. 51, no. 1, pp. 13–21, 2012.

[7] A. Visioli, *Practical PID Control*. Springer Science & Business Media, 2006.

[8] D. A. M. da Silva, A. C. do Nascimento, R. de Barros Araújo, R. J. S. Melo, "Generalized Predictive Controller Applied in a Bidirectional DC-DC Converter," *Proc. Brazilian Power Electronics Conference (COBEP)*, pp. 1–6, IEEE, 2021.

[9] D. A. M. da Silva, A. S. Silveira, A. C. do Nascimento, "State Space Predictive Minimum Variance Controller Applied to a Tacho Generator Motor," *Proc. 14th Seminar on Power Electronics and Control (SEPOC)*, pp. 1–6, IEEE, 2022.

[10] D. A. M. da Silva, M. M. da Silva, R. de Barros Araújo, "Desenvolvimento de uma Interface para Identificação Paramétrica de Processos Utilizando a Estimação dos Mínimos Quadrados Recursivos," *Proc. 14th IEEE International Conference on Industry Applications (INDUSCON)*, pp. 969–975, IEEE, 2021.

[11] D. E. Seborg, T. F. Edgar, D. A. Mellichamp, F. J. Doyle III, *Process Dynamics and Control*. John Wiley & Sons, 2016.

[12] D. Li, F. Zeng, Q. Jin, L. Pan, "Applications of an IMC based PID Controller tuning strategy in atmospheric and vacuum distillation units," *Nonlinear Analysis: Real World Applications*, vol. 10, no. 5, pp. 2729–2739, 2009.

[13] D. Vrabie, K. Vamvoudakis, and F. L. Lewis, *Optimal Adaptive Control and Differential Games by Reinforcement Learning Principles*. IET, 2013.

[14] E. Fernandez-Camacho and C. Bordons-Alba, *Model Predictive Control in the Process Industry*. Springer, 1995.

[15] F. B. Prieto, "Um estudo sobre arquiteturas de hardware para técnicas de fusão sensorial através do EKF e da estimação de estados baseada em filtros híbridos otimizados," *Dissertação submetida ao departamento de engenharia mecânica, Universidade de Brasília*, 2018.

[16] F. C. Martins, D. S. Gontijo, E. N. Gonçalves, "Síntese de observadores PI baseada em otimização evolutiva multiobjetivo H/H2," *Simpósio Brasileiro de Automação Inteligente-SBAI*, 2019.

[17] G. F. Franklin, J. D. Powell, A. Emami-Naeini, *Feedback Control of Dynamic Systems*. Prentice Hall, 2002.

[18] K. J. Åström and T. Hägglund, "Benchmark systems for PID control," *IFAC Digital Control: Past, Present and Future of PID Control*, pp. 5–7, Terrassa, Spain, 2000.

[19] K. J. Åström and B. Wittenmark, *Computer-controlled Systems: Theory and Design*. Courier Corporation, 2013.

[20] K. J. Åström and B. Wittenmark, *Adaptive Control*. Courier Corporation, 2013.

[21] L. A. Aguirre, *Enciclopédia de Automática (vol. 3): Controle e Automação*. Editora Blucher, 2007.

[22] L. A. Aguirre, *Introdução à identificação de sistemas – Técnicas lineares e não-lineares aplicadas a sistemas reais*. Editora UFMG, 2004.

[23] L. C. Yip, "VSTAR CH-47B: A VARIABLE-STABILITY RESEARCH HELICOPTER," *NASA AMES Summer High School Apprenticeship Research Program*, pp. 133, 1984.

[24] M. A. Benjamin, R. A. Rigby, D. M. Stasinopoulos, "Generalized autoregressive moving average models," *Journal of the American Statistical Association*, vol. 98, no. 461, pp. 214–223, 2003.

[25] M. Al-Dhaifllah, D. T. Westwick, "Identification of auto-regressive exogenous Hammerstein models based on support vector machine regression," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 6, pp. 2083–2090, 2012.

[26] M. Morari and E. Zafiriou, *Robust Process Control*. Morari, 1989.

[27] N. N. N. Yamaguti, D. A. M. da Silva, B. G. Dutra, and A. S. Silveira, "Performance and robustness analysis in adaptive and non-adaptive GMVC applied to a MISO process," 2022.

[28] N. N. N. Yamaguti, B. G. Dutra, A. S. Silveira, "Development of a didactic plant and a human-machine interface to compare different digital controllers," *Simpósio Brasileiro de Automação Inteligente-SBAI*, vol. 1, 2021.

[29] P. Chen, Y. Zhang, J. Wang, A. T. Azar, I. A. Hameed, I. K. Ibraheem, N. A. Kamal, F. A. Abdulmajeed, "Adaptive Internal Model Control Based on Parameter Adaptation," *Electronics*, vol. 11, no. 23, pp. 3842, 2022.

[30] R. de Barros Araújo, "Controladores preditivos filtrados utilizando otimização multiobjetivo para garantir offset-free e robustez," *Tese, Programa de Pós-Graduação em Engenharia de Automação e Sistemas, UFSC*, 2017.

[31] R. de Barros Araújo, D. C. Jeronymo, A. R. Coelho, "Hybridization of IMC and PID Control Structures from Filtered Positional Generalized Predictive Controller," *Proceeding Series of the Brazilian Society of Computational and Applied Mathematics*, vol. 4, no. 1, 2016.

[32] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. MIT Press, 2018.

[33] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control: Analysis and Design*, 2nd Edition. Citeseer, 2007.

[34] T. M. O. Santos, R. A. Ricco, É. L. F. C. de Alvarenga, L. Rivaroli, Á. C. O. Penoni, M. F. S. Barroso, "Sintonia ótima do filtro complementar aplicado na junção de sensores inerciais," *Conferência Brasileira de dinâmica, controle e aplicações*, 2017.

[35] W. dos Santos Oliveira and E. N. Gonçalves, "Implementação em C: filtro de Kalman, fusão de sensores para determinação de ângulos," *ForScience*, vol. 5, no. 3, 2017.