Online-Journals.org

iJOE International Journal of Online and Biomedical Engineering

iJOE | eISSN: 2626-8493 | Vol. 20 No. 14 (2024) | OPEN ACCESS

https://doi.org/10.3991/ijoe.v20i14.50995

PAPER

Evaluation of Optimization Algorithms for Use in a Mobile App Aimed at Learning Autotuning of PID Controllers for Engineering Students

Omar Chamorro-Atalaya(✉)

Universidad Nacional Tecnologica de Lima Sur, Lima, Peru

ochamorro@untels.edu.pe

ABSTRACT

Most research in the field of optimization algorithms related to automatic control, both in Hessian and quadratic methods, does not pay enough attention to computational efficiency in data processing, especially in mobile applications on smartphones where their resources are limited. This study seeks to identify the most efficient algorithm in terms of resource consumption for autotuning of PID (proportional, integrator, derivative) controllers in a mobile app aimed at learning engineering students. The BFGS (Broyden-Fletcher-Goldfard-Shanno), simulated annealing, genetic, conjugate gradient, and PSO (particle swarm optimization) algorithms were evaluated. The descriptive scope research and quasi-experimental design showed that the BFGS algorithm is highly efficient and suitable for autotuning of PID controllers in mobile applications, demonstrating consistency in the use of resources with different numbers of testing samples. This study validates a useful optimization method to develop a mobile app that simulates the tuning of PID controllers, being useful for learning topics about automatic control of industrial processes, which allows closing gaps in accessibility and educational availability for engineering students. Future studies should focus on developing mobile applications that use smartphone sensors to generate signals in real time and integrate them with augmented reality environments to build immersive and interactive autotuning systems.

KEYWORDS

mobile app, optimization algorithms, simulation, autotuning controllers, engineering students

1 INTRODUCTION

The use of mobile device technology represents new possibilities for teaching and learning, even in communities where traditional education services are limited [1]. This trend of moving from the universe of fixed technologies to mobile communication networks, and the way in which people use these wireless devices is causing an educational revolution [2]. The true value of mobile devices in education lies in

Chamorro-Atalaya, O. (2024). Evaluation of Optimization Algorithms for Use in a Mobile App Aimed at Learning Autotuning of PID Controllers for Engineering Students. *International Journal of Online and Biomedical Engineering (iJOE)*, 20(14), pp. 38–52. https://doi.org/10.3991/ijoe.v20i14.50995

Article submitted 2024-07-09. Revision uploaded 2024-08-22. Final acceptance 2024-08-22.

their ability to allow students to share information, communicate, collaborate, and innovate using easily accessible tools beyond their simple portability [3], [4]. This boom has transformed paradigms in traditional education, highlighting the importance of improving and analyzing in depth student interactions with mobile technology [5]. In the midst of this highly technological scenario, mobile learning arises, as that which uses these devices, and through them, allows the generation of new learning experiences [6]. The impact and immersion of mobile devices in society is beginning to break into education, with M-Learning standing out as a trend to be implemented in higher education [7]. When we talk about M-Learning, we refer to a type of learning that is supported by technology and that can be carried out at any time and place, not only for the simple transmission of knowledge but also to promote the development of various technological strategies [8]. In this sense, new ways of transferring knowledge together with mobile technologies better promote the assimilation of knowledge [9].

This new scenario makes the use of technologies predominant by enabling personalized and self-regulated learning in students, particularly when using mobile devices in the simulation of different topics [10], improving learning, and effectively transmitting essential content throughout the educational process [11]. For higher education students, developing their skills is not only limited to acquiring theoretical knowledge, but what is most relevant is putting these theories into practice [12]. A particular case is its application in subjects linked to the learning of automatic control, whose conceptualization provides students with cognitive and instrumental tools to contribute technological developments and applications to various productive sectors [13]. The control and supervision of industrial processes are daily operations carried out by a professional in the engineering area in real time for the exploration of a plant, which therefore requires closing gaps regarding insufficiencies in the teaching and learning process [14].

There are various industrial processes in which it is necessary to supervise and control variables, which are generally done through so-called PID controllers (proportional, integrator, derivative), and which to obtain maximum performance must be determined for each plant or process. This is called controller tuning [15]. This results in profit values being established by the trial and error method, with the consequent consumption of time and loss of production during said process [16]. In practice, it is quite difficult to meet all the desired requirements. For example, a controller that is tuned for fast responses usually results in overshoots when disturbed; on the other hand, if the control system is made robust, tuning the controller with gains conservatively, the response of the system to normal changes is slow [17]. To improve the performance of the controllers, optimization techniques have been adapted that modify traditional tuning strategies, so in the comparative analysis of these optimization algorithms the total number of iterations, the execution time, and of the algorithms, the amount of memory in use, and the integral mean square error (ISE) [18].

In this sense, the purpose of this paper is to identify the most efficient optimization algorithm in terms of resource consumption in the automatic search or autotuning of the constants of a PID controller for use in a mobile app developed for learning the tuning of controllers in engineering students. The algorithms evaluated are BFGS (Broyden-Fletcher-Goldfard-Shanno), the simulated annealing algorithm, the genetic algorithm, the conjugate gradient algorithm, and the PSO (particle swarm optimization) algorithm. The research method is descriptive in scope with a quasi-experimental design. The study is based on a comparative analysis of efficiency indicators such as “number of iterations,” “execution time,” and “amount of memory in use” that generate better performance of the PID controller based on the integral indicator of the squared error medium (ISE). Tests were performed on randomly generated synthetic data in Python and evaluated on first, second, and third-order control systems via Google Colab-T4 GPU, initially. Then the performance of the PID controller was evaluated only in the algorithm identified as the one that consumes the least resource on the server that processes the backend of the mobile app. The contribution of this study is to validate an optimization technique that is used to develop a mobile app useful in the simulation of PID controller tuning, and that contributes to closing gaps regarding the difficult accessibility to simulation tools that currently exist and that payment is required for its use. On the other hand, being a mobile app will allow students and users to practice and learn at any time, adapting to their specific educational needs and providing portability, since currently what exists are simulation tools for use on computers or laptops but not on smartphones. It is important to keep in mind that most studies concerning the topic of optimization algorithms do not pay enough attention to computational efficiency during the execution of data processing [19]. In accordance with what was described above, the following research questions are posed:

- RQ1: What is the optimization algorithm that presents the best efficiency in terms of resource consumption during the autotuning process of a PID controller?
- RQ2: What is the efficiency of PID controller autotuning when using the most efficient optimization algorithm when performed on the server processing the mobile app backend?

2 CONCEPTUAL FRAMEWORK

In this conceptual framework, we will address the theoretical aspects linked to optimization algorithms and their use in controller tuning. Figure 1 shows the architecture that illustrates how the automatic tuning of the PID controller works, where $J(\psi)$ represents the objective function, $y(t)$ is the process output, and $e(t)$ is the error between the reference and the measured output. PID is the controller structure, while K_c is the proportional gain, T_i is the integral time, and T_d is the derivative time. The optimization algorithm is incorporated for the control loop, providing optimal values for K_p, T_i, and T_d, depending on the dynamic response of the plant or process [20].

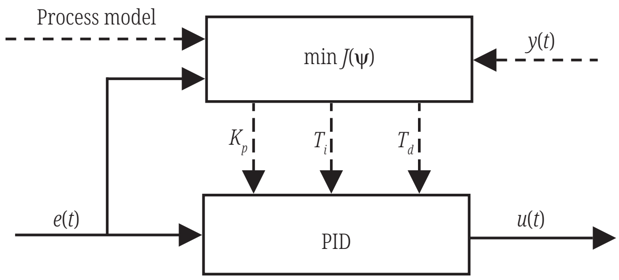

Fig. 1. Architecture of the method for automatic tuning of the PID controller

In this sense, tuning a PID controller involves solving equation 1 [14], thus identifying an optimal vector defined by $\psi = [K_p, T_i, T_d]$.

$$Min\ J(\psi) = \frac{1}{2}\sum_{i=1}^{t}\left(y_i(t) - \widehat{y_i}(t)\right)^2 \tag{1}$$

Furthermore, equation 2 [21] represents the expression of the control signal $u(t)$, which is determined by the error signal $e(t)$ and the value of the PID controller constants.

$$u(t) = K_p\left(e(t) + \frac{1}{T_i}\int_0^t e(t)dt + T_d\frac{de(t)}{dt}\right) \tag{2}$$

If the system associated with the objective function $J(\psi)$ is nonlinear, in general there is no known analytical solution to the optimization problem posed, so one way to solve it is through optimization algorithms [20]. It is important to note that there is no universal optimization algorithm, but rather a collection of algorithms in which each one is best suited to a particular type of optimization problem, and where the choice of the appropriate algorithm for a specific application will depend on whether the problem is solved quickly or slowly and, indeed, whether the solution can be obtained [22].

One of these optimization algorithms is the BFGS algorithm, defined as a type of quasi-Newton algorithm; it is based on approximating the Hessian of the objective function at each point $\nabla^2 f(x_k)$ by another matrix B_k [23]. The advantages lie in using only the first derivatives in the B_k approximation, and consequently, the search direction can be calculated with lower computational cost [24]. BFGS specifically uses the momentum implicitly incorporated in the B_k matrices, and instead of imposing conditions on the approximations of the Hessian B_k, we impose conditions on the inverse of the Hessian H_k, as shown in equation 3 [22].

$$H_{k+1}^{BFGS} = \left(1 - \rho_k s_k y_k^T\right) H_k \left(1 - \rho_k s_k y_k^T\right) + \rho_k s_k y_k^T \tag{3}$$

Where:

$$\rho_k = \frac{1}{y_k^T s_k} \tag{4}$$

Developing the BFGS update formula results in the following expression, as shown in equation 4 [22]. The solution for H_k is expressed in equation 5 [22].

$$H_{k+1}^{BFGS} = \left(1 - \frac{y_k^T H_k y_k}{s_k^T y_k}\right) \frac{s_k s_k^T}{s_k^T y_k} - \left(\frac{s_k y_k^T H_k + H_k y_k s_k^T}{s_k^T y_k}\right) \tag{5}$$

On the other hand, there is the Simulated Annealing algorithm, which is a probabilistic algorithm inspired by metallurgical annealing, which is a controlled cooling technique to reduce defects [25]. The algorithm is formally described as follows: it starts with a random solution x_p, then a random neighbor solution x_n is generated in each iteration. The difference between the values of the objective function $\Delta f = f(x_n) - f(x_p)$ is then calculated. If the neighbor solution is a better solution (i.e., the objective function is improved), then it will replace the current solution [26].

Likewise, there is also the genetic algorithm, which is a blind search method because they do not have more information about the problem to be solved than that obtained from the objective function used [27]. This algorithm works in a similar way to the reproduction of living beings; that is, an analogy is made in which each of the genes on each chromosome represents one of the variables of the solution. In this way, the objective function is evaluated with the values of the variables corresponding to the individual [28]. They carry out the solution search process through three stages: selection, crossover, and mutation; in selection, the algorithm chooses the best chromosomes that will be mutated to generate a new population of processed individuals [29]. In crossing, the best individuals are combined so that their offspring inherit characteristics from both parents, while mutation introduces random changes in some genes, exploring new areas of the search space not reached by other operators [30].

On the other hand, there is the conjugate gradient algorithm, which is widely used in various types of problems due to its rapid convergence and low computational cost, and can be applied to the resolution of linear systems, including systems of normal equations [31]. This algorithm is a variant of the maximum or gradient descent method, where the matrix is required to be symmetric and positive definite of size n × n. This allows finding a solution to the system $Ax = b$ by minimizing the function; thus, in the conjugate gradient method, a single optimal solution of the value of x must be found for the system of equation 6 [32].

$$x_{k+1} = x_k + \alpha_k d_k \tag{6}$$

In which it starts with x_0 of any value to approach the optimal solution. The direction d_0 will be equal to r_0 defined by equation 7 [32], where r is the residual value.

$$\nabla f(x) = Ax - b = -r_k \tag{7}$$

Finally, there is the PSO algorithm, which is a nature-based metaheuristic algorithm that adopts the social behavior of creatures such as schools of fish and flocks of birds [33]. The PSO algorithm imitates animal social behavior but does not require a leader in the group to achieve the objective; when the flock of birds goes to find food, they do not require any leader; they go with one of the members who is in the position closest to the food [34]. In this algorithm, in each iteration, new positions are obtained for the particles through a speed that is determined, considering the best global position and the best current position of each particle [35]. Each particle will go through the search space, based on its experience and adjusting its knowledge according to the best result of the most successful in the swarm thus, each particle symbolizes a possible solution [36].

3 METHODOLOGY

3.1 Scope and research design

The research has a descriptive scope, and its design is quasi-experimental. It is descriptive in scope because it focuses on systematically observing, calculating, and describing the averages of the resource use indicators linked to efficiency in terms of resource consumption, such as "number of iterations," "execution time," and "quantity of memory in use," that generate better performance in the automatic tuning processes of the PID controller based on the "ISE" indicator. Thus, providing a detailed comparison of optimization algorithms without seeking to establish direct causal relationships. Likewise, it is also of a quasi-experimental type in that the computational efficiency results are based on the manipulation of different optimization algorithms in the analysis of performance in the automatic tuning of PID controllers of different processes with first-and second-order transfer functions, and third-order. However, no tests were carried out manipulating all the internal parameters of the optimization algorithms under analysis.

3.2 Data collection and processing

The method used to collect data associated with the indicators of computational efficiency and performance of automatic tuning of PID controllers for different transfer functions is based on the block diagram shown in Figure 2. In the first phase, synthetic transfer functions were generated randomly through the Python program in Google Colab; in order to subject the optimization algorithms to greater complexity in an iterative manner, transfer functions of control systems of six types were generated. These types of control systems were: first order, second order, third order, second order with a first order diverter, third order with a first order diverter, and finally a third order control system with a second order derivative. The time in which the temporal responses of the control systems were generated was 20 seconds. In addition, for the randomness of the coefficients of the transfer functions, values in the range of 1 to 10 were considered. In the second phase, the optimization algorithms to be evaluated were defined, these being BFGS, simulated annealing, genetic, conjugate gradient, and PSO. In each case, the indicators to be evaluated were determined, these being the "number of iterations," the "execution time," and the "amount of memory in use." Once the indicators were identified, they were compared, and the algorithm with the best efficiency in terms of resource consumption was determined. In a third and final phase, the performance of the automatic tuning of the PID controller was determined with the optimization model that showed the best computational efficiency in the previous tests. This last evaluation was carried out on the server that processes the backend of the mobile app, with the purpose of subjecting it to a real context of the processing of the automatic tuning function of the PID controller.

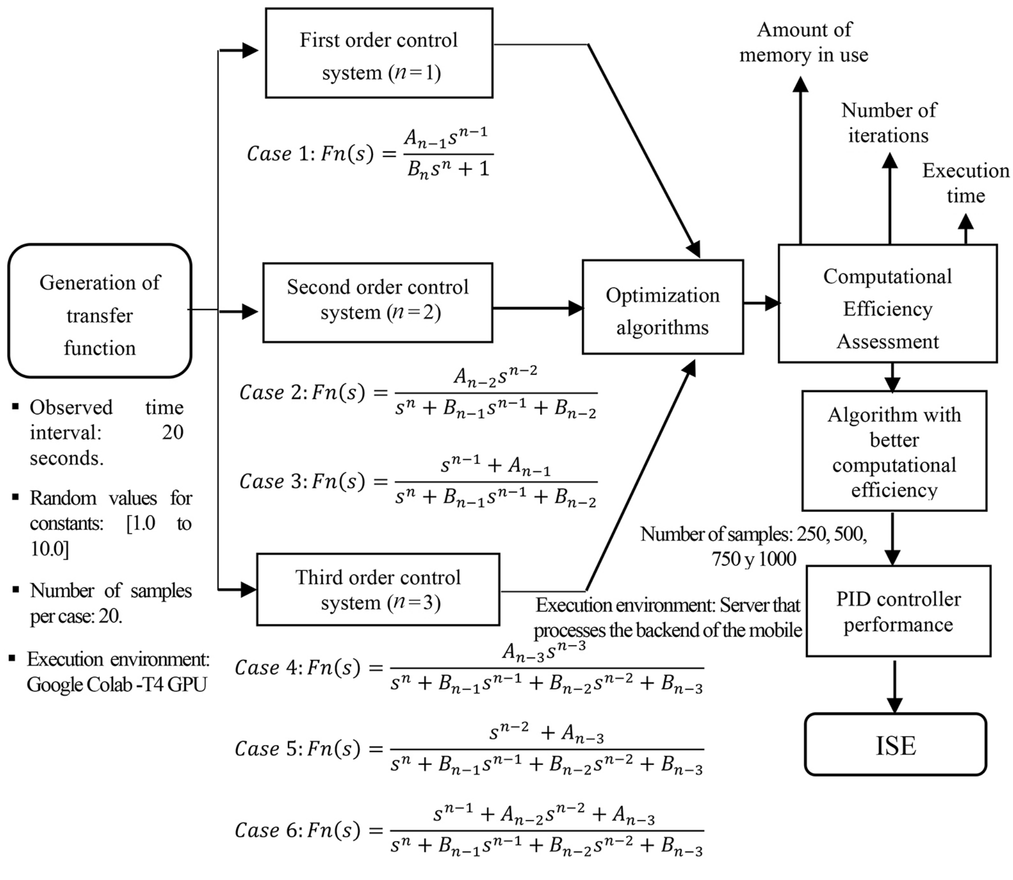

Fig. 2. Method used for data collection

4 RESULTS

4.1 Evaluation of the efficiency in terms of resource consumption of optimization algorithms

In a first evaluation of the five optimization algorithms, on the automatic tuning of a PID controller on a control system whose transfer function is first order, that is, the system with the least complexity, it was identified that the algorithm that on average generated the highest number of iterations, the highest memory consumption, and the one that took the longest execution time was the "Simulated annealing" algorithm, which is why in the following evaluations it was only carried out taking into account the other four algorithms. Table 1 shows the average results of the evaluation of efficiency in terms of resource consumption for automatic tuning of the first-order control system (case 1).

Table 1. Efficiency in terms of resource consumption for autotuning of a first-order control system (case 1)

Algorithm	Average Number of Iterations	Average Memory in Use (MB)	Average Time Used for Execution (Seconds)
BFGS	4.8	0.452	3.316
Simulated annealing	1000.0	0.536	206.949
Genetic	13.2	0.431	22.396
Conjugate gradient	1.7	0.373	5.476
PSO	11.7	0.412	19.999

Analyzing the other five cases with respect to the types of transfer functions of the control systems considered in this study, Figure 3 shows the average iterations by type of control system and by optimization algorithm, in which it was identified that the transfers function that generated the most demand or complexity for the optimization models was the third-order control system with a second-order derivative (case 6). In which the maximum number of iterations reached 241.3, corresponding to the genetic algorithm, while the minimum number of iterations reached a value of 2.6, corresponding to the conjugate gradient algorithm. It should be noted that the BFGS algorithm, even though the number of iterations developed to achieve convergence was not the lowest, was much lower compared to the Genetic and PSO algorithms.

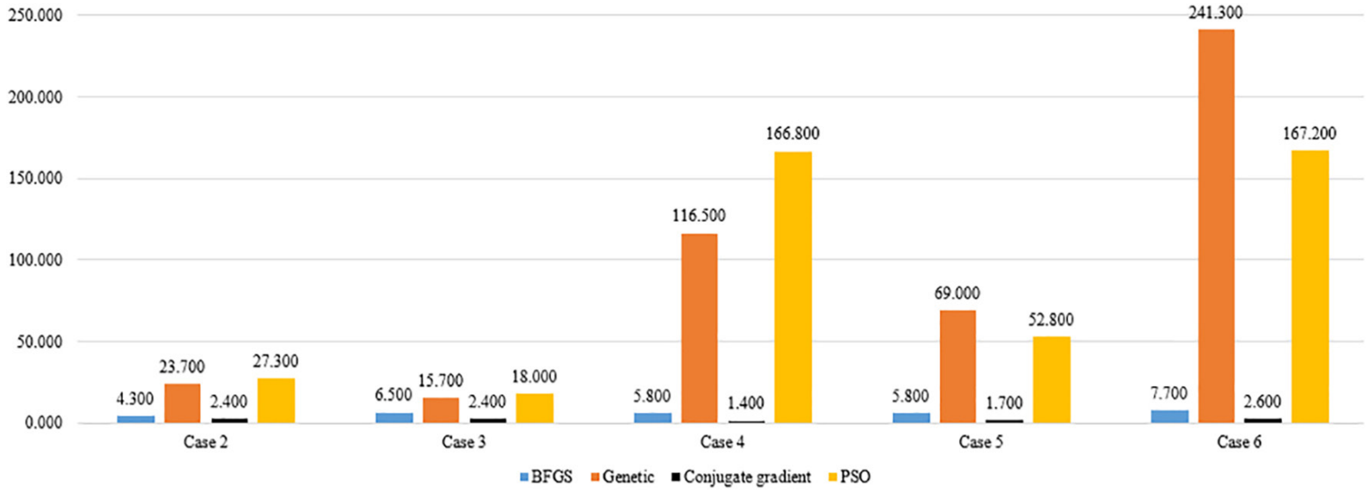

Fig. 3. Average iterations by type of control system and by optimization algorithm

Figure 4 shows the results of the evaluation of the average memory consumption by type of control system and by optimization algorithm. These results show that the control systems corresponding to cases 2 and 6 were the ones that consumed the most memory when performing automatic tuning of the PID controllers. With respect to case 2, for this type of transfer function, the genetic algorithm consumed 0.637 MB, representing the maximum value of memory consumed, while the BFGS algorithm consumed 0.430 MB, representing the minimum value of memory consumed. In relation to case 6, the algorithm that consumed the largest amount of memory was Genetic, consuming 0.621 MB, while the BFGS algorithm consumed the least amount of memory, reaching a value of 0.438 MB.

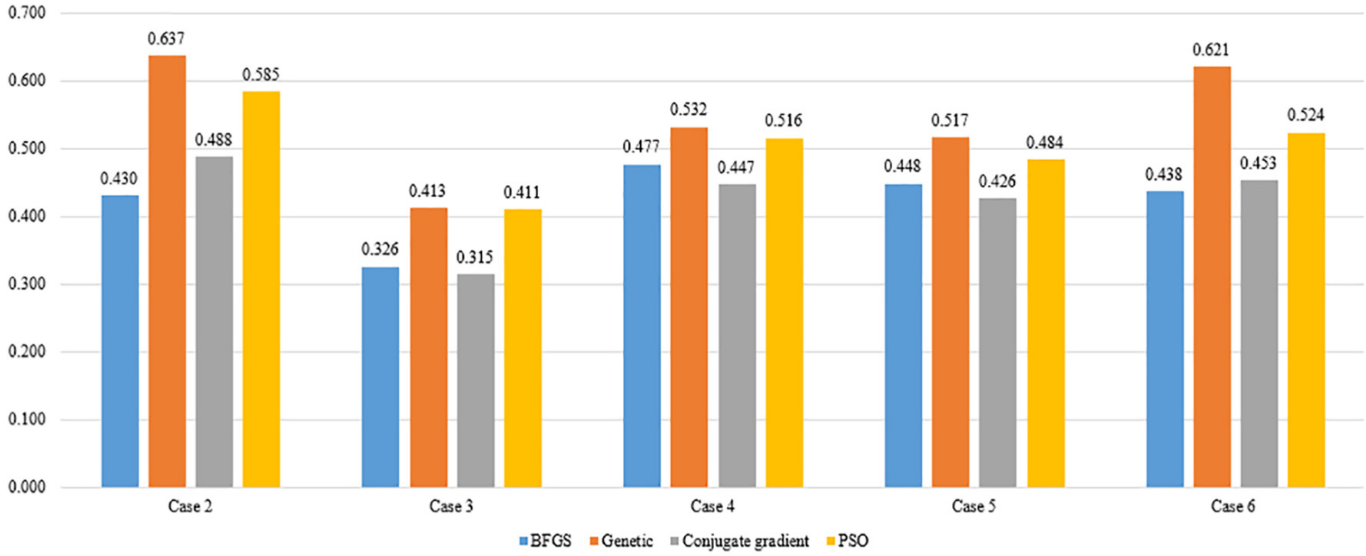

Fig. 4. Average memory consumption by type of control system and by optimization algorithm

Figure 5 shows the average execution time by type of control system and by optimization algorithm, in which the control system that required the greatest complexity or demand for the optimization algorithms to determine the automatic tuning of the controllers PID was the one related to case 6. In this case it was evident that the maximum average execution time reached the value of 502.630 seconds, the same as that corresponding to the genetic model, while the minimum average execution time reached the value of 6.643 seconds, corresponding to the BFGS optimization algorithm.

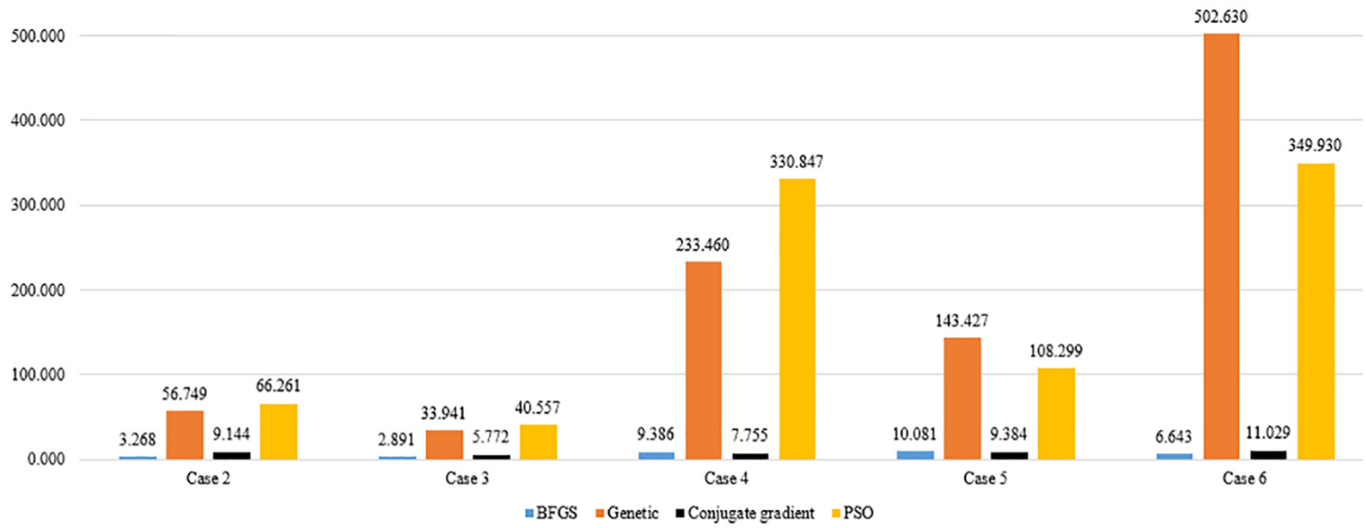

Fig. 5. Average execution time by type of control system and by optimization algorithm

4.2 Evaluation of the efficiency of the PID controller autotuning when using the optimization algorithm on the server that processes the backend of the mobile app

From the evaluation results of the five optimization algorithms considered in this study, it was identified that the BFGS algorithm is the one that generated the best indicators regarding efficiency in the transfer functions that presented the greatest demand when determining the automatic tuning of a PID controller. Table 2 shows the results obtained when developing the execution on the server that processes the backend of the mobile app, which shows that the performance of the PID controller using the BFGS optimization algorithm is consistent and efficient throughout different sample sizes. With a number of tested samples of 250, 500, 750, and 1000, the average number of iterations remained relatively stable, ranging between 17.476 and 17.546 iterations, with a standard deviation that varies slightly between 1.636 and 1.964. It is also observed that memory usage remained constant at 0.191 MB, with a very low standard deviation, evidencing efficient management of system resources. Also, regarding the execution time, the results show an average time that increases proportionally with the sample size, from 2.179 seconds for 250 samples to 2.461 seconds for 1000 samples, with standard deviations varying between 0.283 and 0.298 seconds.

Table 2. PID controller performance using the BFGS optimization algorithm

Number of Samples Tested	Number of Iterations		Memory in Use (MB)		Time Used for Execution (Seconds)	
	Average	Standard Deviation	Average	Standard Deviation	Average	Standard Deviation
250	17.476	1.689	0.191	0.010	2.179	0.283
500	17.546	1.636	0.191	0.007	2.958	0.446
750	17.347	1.877	0.190	0.006	2.404	0.298
1000	17.370	1.964	0.190	0.005	2.564	0.408

Proceeding with the performance evaluation, we now proceeded to analyze the value of the integral of the mean square error, in which, when testing with 250 samples, a concentrated distribution of the K_p, $T_{i,}$ and T_d values is observed. Most of the points are grouped in regions where the ISE is relatively low, indicating good performance of the PID controller with these parameters. By increasing to 500 samples, the dispersion of the points increases slightly, but they still remain clustered in areas with low ISE. This shows that the PID controller remains efficient and robust, despite the increase in the number of samples. In the case of 750 samples, a greater dispersion is noted in the values of K_p, T_i, and T_d. However, the points still tend to cluster in areas with low ISE, suggesting that the BFGS algorithm manages to handle an increasing number of samples while maintaining acceptable performance. Likewise, in the case of 1000 samples, the dispersion of the points is more pronounced; however, the values of K_p, $T_{i,}$ and T_d continue to show a low ISE level (see Figure 6).

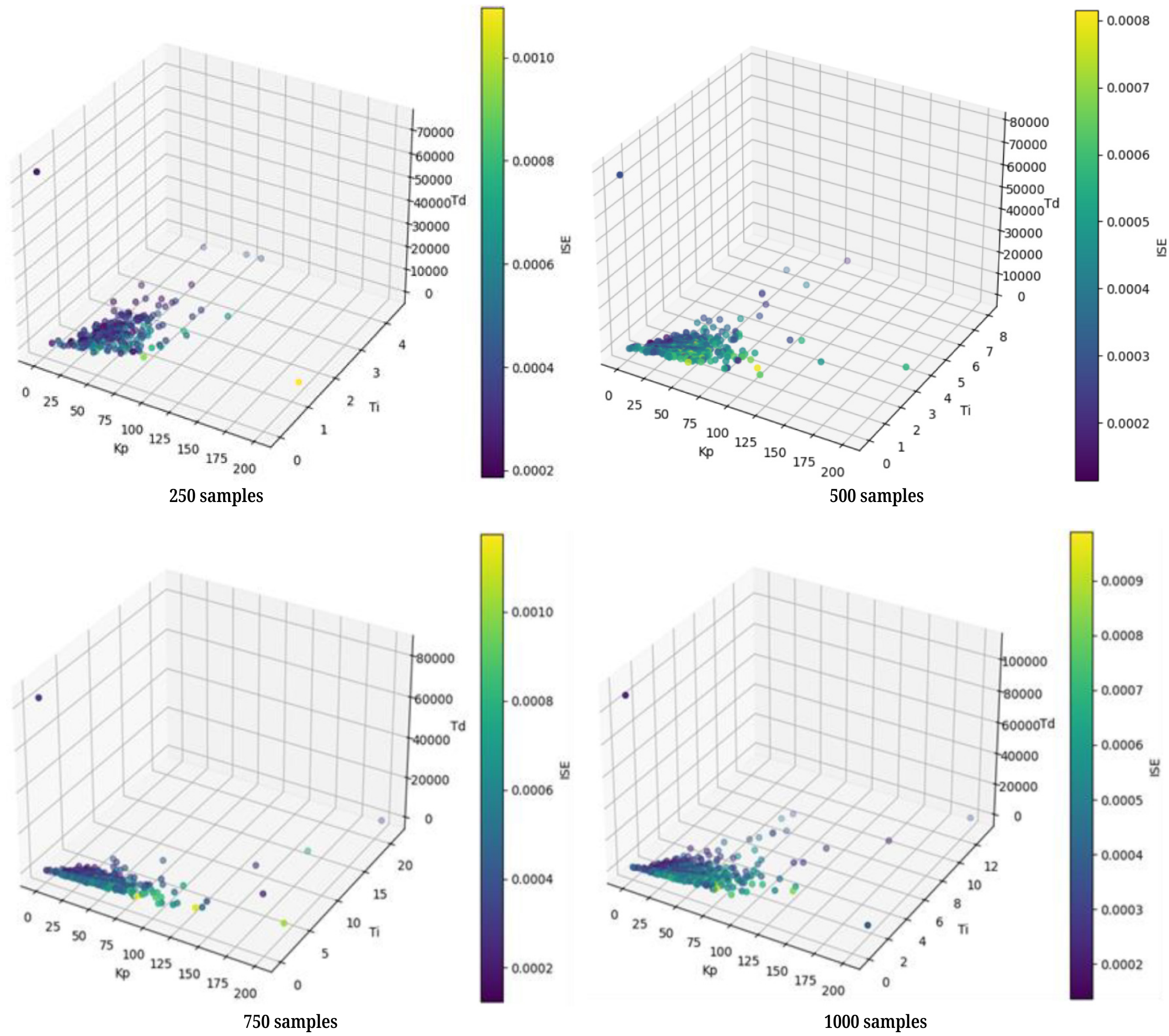

Fig. 6. ISE indicator based autotuning efficiency test for 250, 500, 750 and 1000 samples

5 DISCUSSION

In relation to the first research question, regarding which is the optimization algorithm that presents the best efficiency in terms of resource consumption during the automatic tuning process of a PID controller, regarding the evaluation, it was identified that it is the BFGS algorithm, followed slightly by the conjugate gradient algorithm. Performance is superior when performing PID controller autotuning with BFGS due to its gradient-based approach and its ability to quickly converge to differentiable control systems. In this regard, [37] points out that gradient-based methods allow optimal parameter solutions to be quickly found, improving control precision and reducing configuration time and overshoot, which is essential for nonlinear industrial applications. In contrast, metaheuristic algorithms such as simulated annealing, genetic and PSO do not depend on the differentiability of the objective function; they tend to be less efficient in terms of resource consumption. In agreement with [38], it is stated that although the Genetic and PSO algorithms can offer good performance in terms of searching for global minima and adjusting PID parameters, they present a greater consumption of gradient-based resources. Likewise, [39] highlights that optimization methods such as genetic algorithms and PSO can produce optimal settings of the PID parameters, but at the cost of greater use of resources and optimization time.

In relation to the second research question regarding evaluating the performance of the PID controller when using the most efficient optimization algorithm when performing automatic tuning on the server that processes the backend of the mobile app, the results reflect the robustness and efficiency of the algorithm. BFGS for automatic tuning of PID controllers, showing that as the number of samples and processing stress increased, the PID controller continued to function efficiently. In this regard, in [40] it is stated that although the genetic algorithm provided good results in terms of establishment time, the differential evolution algorithm shows better efficiency in terms of convergence time and mean square error when analyzed in an industrial system implemented with wastewater tanks, suggesting that gradient-based algorithms such as BGFS may be more efficient in complex and time-varying systems. In addition to the performance evaluation, it is observed that as the number of samples increases, the K_p values tend to vary in a wider range, while the Ti and Td values show less dispersion, which shows stability in the integration and derivation of the error. However, in [41], they analyze three optimization algorithms, identifying that the one that offers the best performance is the genetic algorithm, providing an ISE value equal to 0.015. It is important to reflect that the author focuses on the response of the control system and not on the analysis of computational efficiency, an aspect that is still relevant when it will be used in a mobile application environment on a smartphone, with limited resources.

Additionally, it is essential to continue optimizing the performance indicators of the mobile application in order to achieve an optimal user experience; therefore, the mobile application must be designed under an intuitive design and with optimal response times to improve the performance and adoption of the tool [42]. Positive feedback from students highlights the potential of these applications to connect theory with practice [43]. However, it is necessary to continue refining the tools to reduce the discrepancies between theoretical and practical results, improving their precision and reliability with those obtained in real scenarios [44].

6 CONCLUSIONS

In accordance with the purpose of the research and the research questions established in this study, it was identified that when evaluating the efficiency in terms of resource consumption in the search for self-adjustment of the constants of a PID controller of the five evaluated algorithms (BFGS, annealing simulated, genetic, conjugate gradient, and PSO), the one that showed the best efficiency is the BFGS algorithm. Likewise, when evaluating the performance when developing the execution on the server that processes the backend of the mobile app, it was evident that the performance is consistent and efficient across different sample sizes. That is to say, with tested samples of 250, 500, 750, and 1000, the average number of iterations remained relatively stable, ranging between 17.476 and 17.546 iterations on average. It is also observed that memory usage remained constant at 0.191 MB. While when evaluating the execution time, the results show an average time that increases proportionally with the sample size, from 2.179 seconds for 250 samples to 2.461 seconds for 1000 samples. Based on the above, it is concluded that the BFGS algorithm is highly efficient and suitable for the autotuning of PID controllers in mobile applications, demonstrating consistency in resource use when subjecting it to different numbers of testing samples. This study validates a useful optimization method to develop a mobile app that simulates the tuning of PID controllers, being useful for learning topics about automatic control of industrial processes and allowing to close accessibility gaps for education in engineering students.

7 FUTURE STUDIES

Future studies should be aimed at the development of mobile applications that make use of smartphone sensors to generate input signals in real time and integrate them with environments created with augmented reality, to build systems with automatic tuning of immersive PID controllers and with a higher degree of interactivity for students. For these studies, augmented reality SDKs (software development kits) for mobile applications such as Apple's ARKit or Google's ARCore could be used, which provide integration capacity with sensors such as accelerometer, magnetometer, gyroscope, proximity sensor, and ambient light sensor, among others.

8 REFERENCES

[1] J. F. Prado, "Mobile learning and the sustainable development goals in higher education," *Revista Universidad y Sociedad*, vol. 12, no. 4, pp. 230–233, 2020. http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2218-36202020000400230

[2] A. L. Vallina and A. G. Martínez, "Estrategia de aprendizaje con dispositivos móviles en el ministerio de educación superior de cuba," *Revista Cubana De Educación Superior*, vol. 41, no. 2, pp. 434–452, 2022. https://revistas.uh.cu/rces/article/view/186

[3] A. Hassan, H. Pathan, S. S. Kotamjani, S. M. Abbas Hamza, and R. Rastogi, "Analyzing the deep learning-based mobile environment in educational institutions," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 18, no. 9, pp. 155–166, 2024. https://doi.org/10.3991/ijim.v18i09.49029

[4] M. E. R. Rivera, G. T. Dávila, and E. R. Lizama, "Design and development of an educational mobile application for optimize communication and interaction between members of educational institutions in real time," *Industrial Data Journal*, vol. 24, no. 1, pp. 277–307, 2021. https://doi.org/10.15381/idata.v24i1.19421

[5] J. Wang, "Improvement of student interaction analysis in online education platforms through interactive mobile technology and machine learning integration," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 18, no. 9, pp. 35–49, 2024. https://doi.org/10.3991/ijim.v18i09.49291

[6] S. R. R. Soncco, "Aprendizaje móvil y las competencias del idioma inglés en la educación superior," *Comuni@cción: Revista De Investigación En Comunicación Y Desarrollo*, vol. 13, no. 2, pp. 138–148, 2022. https://doi.org/10.33595/2226-1478.13.2.571

[7] J. M. Romero-Rodríguez, I. Aznar-Díaz, F. J. Hinojo-Lucena, and G. Gómez-García, "Uso de los dispositivos móviles en educación superior: relación con el rendimiento académico y la autorregulación del aprendizaje," *Revista Complutense de Educación*, vol. 32, no. 3, pp. 327–335, 2021. https://doi.org/10.5209/rced.70180

[8] J. E. Martínez and L. A. Rodríguez, "Uso de aplicaciones móviles como herramienta de apoyo tecnológico para la enseñanza con metodología STEAM," *Revista Politécnica*, vol. 18, no. 36, pp. 75–90, 2022. https://doi.org/10.33571/rpolitec.v18n36a6

[9] R. C. Chirino-García and J. Hernández-Corona, "M-learning: Strategy for the promotion of mobile e-learning in higher education institutions," *Electronic Journal of Educational Sciences*, vol. 3, no. 5, pp. 102–121, 2020. https://fundacionkoinonia.com.ve/ojs/index.php/epistemekoinonia/article/view/684

[10] M. P. Martínez, Z. S. L. Collazo, J. S. Baranda, and A. Santos-Fuentefria, "Potentialities of the app EveryCircuit in the laboratory practices of electrical circuits in the electrical engineering career of the Universidad Tecnologica de La Habana," *Modelling in Science Education and Learning*, vol. 14, no. 2, pp. 43–50, 2021. https://doi.org/10.4995/msel.2021.15005

[11] A. Fortuna, Waskito, Purwantono, A. Kurniawan, W. Andriani, and M. Alimin, "Designing learning media using augmented reality for engineering mechanics course," *Journal of Engineering Researcher and Lecturer*, vol. 2, no. 1, pp. 18–27, 2023. https://doi.org/10.58712/jerel.v2i1.20

[12] J. Li and Q. Li, "Enhancing educational design capabilities through interactive mobile and adaptive learning platforms: An empirical study," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 18, no. 8, pp. 87–101, 2024. https://doi.org/10.3991/ijim.v18i08.48875

[13] F. J. López, "Prototype of reconfigurable mechatronic plant for teaching and learning automatic control," *ACOFI 2022 International Engineering Education Meeting*, 2022. https://doi.org/10.26507/paper.2134

[14] C. R. M. Hernández, W. G. Hernández, and G. C. Lemus, "Ability to model dynamic automatic control processes," *Chemical Education Journal*, vol. 32, no. 1, 2021. https://doi.org/10.22201/fq.18708404e.2021.1.75429

[15] E. Heinänen, "A method for automatic tuning of PID controller next the optimization of Lus-Jaakola," MS Thesis, Tampere University of Technology, Shenzhen, 2018. [Online]. Available at: https://core.ac.uk/download/pdf/196558079.pdf

[16] L. A. S. Peralta, "Research design for the implementation of autotuning of PID control parameters in a winder for voltage control and improvement in energy efficiency," Thesis, Faculty of Engineering, University of San Carlos of Guatemala, Guatemala, 2020. [Online]. Available at: http://www.repositorio.usac.edu.gt/19632/1/Luis%20Alfredo%20Salazar%20Peralta.pdf

[17] B. R. Lliuyacc, "Tuning a PID controller using particle swarm optimization for the AGC of a multi-area electrical system," M.S. Thesis, Department of Electrical Engineering, Sevilla University, Sevilla, Spain, 2014. [Online]. Available at: https://idus.us.es/handle/11441/27037

[18] E. I. Salazar Zambrano, "Comparative analysis of optimization techniques for tuning adaptive PID controllers," Thesis, Faculty of Electrical Engineering and Electronics, National Polytechnic School, Quito, Ecuador, 2020. [Online]. Available at: https://bibdigital.epn.edu.ec/handle/15000/20917

[19] J. R. Delgado, H. G. Diez, C. M. Pérez, F. J. Ferri, R. T. Rasúa, and R. Bello, "Use of mathematical optimization models in the solution of computational problems," *Annals of the Cuban Academy of Sciences*, vol. 12, no. 3, 2020. [Online]. Available at: http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2304-01062022000300009

[20] H. Valdés-González, J. L. Salazar, L. Reyes-Bozo, E. Vyhmeister, M. Gómez-Varela, and F. C. Montecinos, "Optimization based control of a direct Rotary Dryer," *UNAL Journal*, 2012. [Online]. Available at: https://revistas.unal.edu.co/index.php/dyna/article/view/20881/39113

[21] T. R. Biyanto, N. Sehamat, N. A. Sordi, and H. Zabiri, “Simultaneous optimization of tuning PID cascade control system using Duelist algorithms,” in *IOP Conference Series: Materials Science and Engineering*, vol. 458, 2018. https://doi.org/10.1088/1757-899X/458/1/012053

[22] J. G. Ventura, “Applications of Quasi-Newton optimization methods in deep learning,” MS Thesis, Basic and Environmental Sciences Area, Santo Domingo Technological Institute, Dominican, 2023. [Online]. Available at: https://josegarciav.github.io/files/MMA_TESIS_JG.pdf

[23] A. D. X. Reyes, “A non-differentiable bundle type optimization method applied to the flow of viscoplastic materials,” Thesis, Science Faculty, National Polytechnic School, Quito, Ecuador, 2022. [Online]. Available at: https://bibdigital.epn.edu.ec/handle/15000/22266

[24] B. Vitoriano and A. Ramos, “Mathematical programming: Optimization methods,” *Complutense University of Madrid*, 2023. [Online]. Available at: https://blogs.mat.ucm.es/bvitoriano/wp-content/uploads/sites/69/2023/02/MM_PMII_I_IIb.pdf

[25] R. E. S. Ruiz, R. J. E. Rodríguez, and G. C. L. Hernández, “Hybrid algorithm of artificial neural networks with simulated annealing for prediction in data mining,” *Revista Vínculos*, vol. 17, no. 2, pp. 97–103, 2020. https://doi.org/10.14483/2322939X.17232

[26] N. J. M. Acosta, “Simulated annealing and clonal selection algorithm applied to the problem of the traveling agent,” *Research in Computing Science*, vol. 149, no. 8, pp. 1211–1226, 2020. [Online]. Available at: https://rcs.cic.ipn.mx/2020_149_8/Recocido%20simulado%20y%20el%20algoritmo%20de%20seleccion%20clonal%20aplicados%20al%20problema%20del%20agente%20viajero.pdf

[27] J. A. D. Más, “Optimization of plant layout of industrial facilities using genetic algorithms: Contribution to the control of the geometry of activities,” *Universitat Politècnica de València*, 2006. https://doi.org/10.4995/Thesis/10251/135821

[28] J. A. A. Marín, S. L. R. Otero, and M. E. M. Arias, “Programación de mantenimiento preventivo usando algoritmos genéticos,” *Lámpsakos*, no. 23, pp. 37–44, 2020. https://doi.org/10.21501/21454086.3112

[29] J. A. Espitia-Mendez and G. L. Mendoza-Rojas, “Methodology based on genetic algorithm for a textil industry company production scheduling,” *Ingeniería Investigación y Tecnología*, vol. 22, no. 4, pp. 1–16, 2021. https://doi.org/10.22201/fi.25940732e.2021.22.4.032

[30] J. M. González-Martín, A. J. Sánchez-Medina, and J. B. Alonso, “Optimization of the prediction of financial problems in Spanish private healthcare companies by applying genetic algorithms,” *Gaceta Sanitaria*, vol. 33, no. 5, pp. 462–467, 2019. https://doi.org/10.1016/j.gaceta.2018.01.001

[31] I. Martín-Álvarez, M. I. Castillo, J. I. Aliaga, and D. Andrade, “Incorporation of malleability in an iterative method,” in *Advances in Computer Architecture and Technology. Minutes of the SARTECO Conference 20/21*, 2021, pp. 513–521. https://doi.org/10.5281/zenodo.8013783

[32] C. C. D. B. Neto, “Linear system solution using conjugate gradient,” Thesis, Department of Computing and Technology, Federal University of Rio Grande do Norte, Natal–RN, Brazil, 2022. [Online]. Available at: https://repositorio.ufrn.br/bitstream/123456789/50445/1/TCC_BSI_Clodoaldo.pdf

[33] A. K. Vincent and R. Nersisson, “Particle swarm optimization based PID controller tuning for level control of two tank system,” in *IOP Conf. Series: Materials Science and Engineering*, 2017, vol. 263. https://doi.org/10.1088/1757-899X/263/5/052001

[34] M. Q. Ccachuco and C. C. Cáceres, “PSO and ACO algorithms applied to the tuning of a PID controller for position control of a 200 mm hydraulic cylinder,” in *18th LACCEI International Multi-Conference for Engineering, Education Caribbean Conference for Engineering and Technology*, July 27–31, 2020. https://laccei.org/LACCEI2020-VirtualEdition/full_papers/FP371.pdf

[35] V. Quezada-Aguilar, J. C. Quezada-Quezada, S. J. C. Tuoh-Mora, and A. Cuatepotzo-Bravo, "Multiple system of production optimized by PSO," *Ingeniería Investigación y Tecnología*, vol. XXl, no. 1, pp. 1–11, 2020. https://doi.org/10.22201/fi.25940732e.2020.21n1.006
[36] V. Álvarez-Garduño, N. Guadiana-Ramírez, and A. Anzueto-Ríos, "Comparative analysis of the modification of the inertia parameter for the improvement of the PSO algorithm performance," *Científica*, vol. 25, no. 1, 2020. https://doi.org/10.46842/ipn.cien.v25n1a09
[37] E. Kosareva, A. Zenkin, I. Kirilenko, and A. Kapitonov, "Quadrotor control parameters optimization using gradient descent method," in *Majorov International Conference on Software Engineering and Computer Systems*, 2019, vol. 2590, no. 12, pp. 1–10. [Online]. Available at: https://ceur-ws.org/Vol-2590/paper12.pdf
[38] M. S. Saad, H. Jamaluddin, and I. Z. M. Darus, "PID Controller Tuning Using Evolutionary Algorithms," *WSEAS Transactions on Systems and Control*, vol. 7, no. 4, 2012. https://wseas.com/journals/sac/2012/54-890.pdf
[39] R. H. Subrata, F. Gozali, and E. Djuana, "Computational and intelligent optimization tuning method for PID controller," *Journal of Theoretical and Applied Information Technology*, vol. 100, no. 7, 2022. [Online]. Available at: https://www.jatit.org/volumes/Vol100No7/4Vol100No7.pdf
[40] V. J. M. Ramos, M. A. N. Grijalba, P. O. R. Zamudio, J. D. S. Agreda, J. H. A. Alcántara, and E. A. M. Ramos, "ANÁLISIS Y COMPARACIÓN DE LA SINTONIZACIÓN DE UN CONTROLADOR PID DE NIVEL EN UN TANQUE USANDO ALGORITMOS DE OPTIMIZACIÓN," *Revista De Investigaciones*, vol. 10, no. 4, pp. 348–359, 2021. https://doi.org/10.26788/riepg.v10i4.3499
[41] G. Yilmaz, "Comparison of different methods for optimization of PID controller gain coefficients," *Kırklareli Üniversitesi Mühendislik Ve Fen Bilimleri Dergisi*, vol. 9, no. 2, pp. 254–264, 2023. https://doi.org/10.34186/klujes.1310728
[42] L. D. P. Indacochea, "Android application for the simulation of complexive tests for the nursing course at the state University of the South of Manabí," Thesis, Faculty of Technical Sciences, 2022. [Online]. Available at: https://repositorio.unesum.edu.ec/handle/53000/4766
[43] P. M. Martínez, L. Z. S. Collazo, S. J. Baranda, and A. Santos-Fuentefria, "Potential of the EveryCircuit app in the laboratory practices of electrical circuits in the electrical engineering degree at the Technological University of Havana," *Modelling in Science Education and Learning*, vol. 14, no. 2, pp. 43–50, 2021. https://doi.org/10.4995/msel.2021.15005
[44] W. R. Ammirata, R. R. Martínez, and M. M. Pérez, "Simulation of a filtration process for the separation processes laboratory of the Metropolitan University," *TEKHNÉ Journal*, pp. 46–61, 2020. [Online]. Available at: https://revistasenlinea.saber.ucab.edu.ve/index.php/tekhne/article/view/5442/4790

9 AUTHOR

Omar Chamorro-Atalaya is a Research Professor at the National Technological University of Lima Sur, in Lima, Peru. He is a teacher of the subject of Automatic Process Control. He is a researcher qualified at Level I by CONCYTEC (National Council of Science, Technology and Technological Innovation) (E-mail: ochamorro@untels.edu.pe).