UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

A Comparative Evaluation of Metaheuristic Optimization Methods For Control Applications

David Francisco García Torres

Electrónica y Automatización

Trabajo de fin de carrera presentado como requisito

para la obtención del título de

Ingeniero en Electrónica

Quito, 25 de Junio de 2023

UNIVERSIDAD SAN FRANCISCO DE QUITO USFQ

Colegio de Ciencias e Ingenierías

HOJA DE CALIFICACIÓN

DE TRABAJO DE FIN DE CARRERA

A Comparative Evaluation of Metaheuristic Optimization Methods For Control Applications

David Francisco García Torres

Nombre del profesor Tutor: Óscar Camacho, Ph.D.

Quito, 25 de Junio de 2023

© DERECHOS DE AUTOR

Nombres y apellidos:	David Francisco García Torres
Código:	00209730
C.I.:	1751294818
Fecha:	Quito, 25 de Junio de 2023

ACLARACIÓN PARA PUBLICACIÓN

Nota: El presente trabajo, en su totalidad o cualquiera de sus partes, no debe ser considerado como una publicación, incluso a pesar de estar disponible sin restricciones a través de un repositorio institucional. Esta declaración se alinea con las prácticas y recomendaciones presentadas por el Committee on Publication Ethics COPE descritas por Barbour et al. (2017) Discussion document on best practice for issues around theses publishing, disponible en `http://bit.ly/COPETheses`

UNPUBLISHED DOCUMENT

Note: The following capstone project is available through Universidad San Francisco de Quito USFQ institutional repository. Nonetheless, this project – in whole or in part – should not be considered a publication. This statement follows the recommendations presented by the Committee on Publication Ethics COPE described by Barbour et al. (2017) Discussion document on best practice for issues around theses publishing available on `http://bit.ly/COPETheses`

RESUMEN

Este trabajo compara métodos metaheurísticos para optimizar la sintonía en dos controladores aplicados a procesos con largos tiempos muertos. Se utilizó un laboratorio de control de temperatura Arduino con un retardo basado en software se utilizó para probar experimentalmente los controladores optimizados. El Smith Predictor y el PI se sintonizan utilizando tres algoritmos meta-heurísticos de optimización diferentes: Algoritmo de Optimización Ballena, Optimizador Lobo Gris, y Optimizador León Hormiga, para buscar los parámetros para el mejor rendimiento basado en el Error Cuadrático Integral como función de coste. Estos esquemas de control Estos esquemas de control se comparan cualitativamente mediante distintos índices de rendimiento para determinar cómo pueden mejorar estos algoritmos el rendimiento del proceso. estos algoritmos pueden mejorar el rendimiento del proceso mediante la búsqueda de soluciones óptimas para el ajuste de los parámetros. parámetros. Sin embargo, los resultados indican que el Smith Predictor con el algoritmo de optimizacion por ballenas entre los esquemas de control probados, es adecuado, equilibrado y funciona mejor en procesos térmicos con tiempos muertos largos. .

***Palabras clave*:** TCLAB, Smith Predictor, PI, tiempo muerto, metaheurística, algoritmos de optimización.

ABSTRACT

This work compares metaheuristic methods to optimize tuning in two controllers applied to processes with long dead time. An Arduino Temperature Control Lab with an additional software-based delay was used to test the optimized controllers experimentally. The Smith Predictor and the PI are tuned using three different meta-heuristic optimization algorithms: Whale Optimization Algorithm, Gray Wolf Optimizer, and Ant Lion Optimizer, to search parameters for the best performance based on Integral Square Error as the cost function. These control schemes are qualitatively compared using different performance indices to determine how these algorithms can enhance the process performance by seeking optimal solutions for tuning parameters. However, the findings indicated that the Smith Predictor with Whale Optimization Algorithm, among the tested control schemes, is suitable, balanced, and performs better for thermal processes with long dead time.

***Keywords*:** TCLAB, Smith Predictor, PI, dead-time, metaheuristic, optimization algorithms.

CONTENTS

List of Figures

List of Tables

DEDICATORIA

Con todo mi cariño y amor para mis padres quienes son mi mayor motivación, y en especial para mi mismo.

AGRADECIMIENTO

Quiero expresar mi sincero agradecimiento a mi tutor de tesis, Óscar Camacho, por su invaluable guía, apoyo incondicional y paciencia a lo largo de todo el proceso de investigación. Gracias a su experiencia y conocimientos, he sido moldeado de la mejor manera posible.

Además, quiero extender mi gratitud a todas las personas que han sido un apoyo constante en mi vida: mis padres, mis hermanas, mis abuelos, mi familia y mis amigos. Cada uno de ellos ha sido y seguirá siendo parte integral de mi día a día, acompañándome aunque estén lejos, ya que en mi memoria perduran sus historias, sonrisas e ideas, las cuales han contribuido a forjar la persona que soy hoy en día.

CHAPTER 1

INTRODUCTION

Dead time is commonly encountered in engineering, physics, and biology due to the inherent nature of certain processes. It manifests in various applications, including aircraft operations, chemical processes, teleoperation, and biological (Camacho and Leiva, 2020). Dead times can arise from different sources, such as delays in control input, state variables or measurements, wireless network transmission, physical transportation, and decision-making in response to stimuli that affect the overall response of the system(Camacho and Martínez, 2017, Normey-Rico and Camacho, 2007, Mejía et al., 2022).

It is crucial to recognize that dead time can significantly affect system performance, emphasizing the need for careful consideration and mitigation strategies to optimize operations and minimize undesirable effects. (Camacho and Leiva, 2020, Normey-Rico and Camacho, 2007).

Processes that exhibit a long dead time (t_0) compared to the dominant time constant (τ) are called dominant dead-time systems ($t_0 > \tau$). This phenomenon can reduce the reaction time to reject disturbances, decrease the performance and stability margins, and prolong the time response of the process (Camacho and Martínez, 2017). Long dead time systems can cause instability in the closed-loop control system (Camacho and Leiva, 2020). Moreover, achieving effective control using a PID approach becomes challenging when dealing with long dead time. Thus, various schemes and strategies have been developed to overcome this challenge. One notable solution is the Smith predictor (SP), designed as the first compensator specifically addressing dead time in feedback control systems(Mejía et al., 2022). The SP predicts the system's output based on the present input and the known dead time. This predicted output is then utilized to calculate the control signal, which is applied to the system to minimize the error between the desired and actual outputs (Camacho et al., 2020).

Over the years, diverse adjustment methods have continuously been developed to enhance delay compensation controllers (O'Dwyer, 1999). Most of these methods rely on First Order Plus Dead-time (FOPDT) models. However, some time is required for additional fine-tuning of improvements(Smith and Corripio, 2005); consequently, exploring alternative paths similar to optimization algorithms has become a topic of active research (Darwish, 2018). An optimization algorithm is a mathematical procedure that tries to find the best solution to a given problem. An optimization algorithm aims to find the optimal solution that minimizes or maximizes a given objective function. A notable example is the use of evolutionary methods inspired by living organisms, commonly called meta-heuristic optimization algorithms (Darwish, 2018). This field encompasses the study of computer science, mathematics, and biology and has gained significant attention in recent years. These methods are inspired by life in nature; these bioinspired computing optimization algorithms offer powerful problem solving strategies. In addition, they have been increasingly used in machine learning to find optimal solutions to challenging problems in the realms of science and engineering(Mejía et al., 2022, Darwish, 2018).

This work compares and evaluates two control methods for systems with long dead time. The first method uses a proportional-integral (PI) controller, and the second uses the Smith predictor. The two controllers are tuned, considering three metaheuristic optimization methods. They are whale, gray wolf, and ant-lion optimization algorithms. The evaluation uses an Arduino Temperature Control Lab (TCLab) (de Moura Oliveira et al., 2021). For comparison purposes, the controllers are initially adjusted using Dahlin's and Skogestad (Camacho et al., 2020) methods. The performance indices considered are the integral square error (ISE), the integral square controller output (ISCO), the optimization algorithm time, the settling time and the maximum overshoot. Finally, the results indicate that the Smith predictor combined with Whale optimization is particularly effective for the studied process.

The remainder of the study is organized as follows. Section II introduces fundamental concepts, Section III provides a brief description of the TCLab kit, Section IV reports the two controllers, Section V discusses the optimization algorithms, Section VI presents the results and analytical evaluation, and Section VII concludes this work.

CHAPTER 2

FUNDAMENTALS

This Section gives the background to understand the most relevant topics of this study.

2.1 Dead time

In control systems, dead time is the time it takes for an input signal to be processed and an output signal to be generated. It is also known as transport time or time delay (Normey-Rico and Camacho, 2007).

2.2 Proportional Integral Control (PI)

The PI controller combines two essential components: proportional control and integral control. Proportional control governs the strength of the control action and its responsiveness to the error. On the contrary, integral control enables the controller to address steady-state error and eliminate any deviation between the desired setpoint and the actual output (Camacho et al., 2020).

2.3 Smith Predictor (SP)

The Smith predictor is a controller designed to handle dead time in systems. It is often used with PI or PID controllers, which are adjusted to regulate the process without delay. The term "predictor" refers to its ability to estimate the system output without delay. However, it is important to note that this traditional Smith predictor approach is only suitable for stable systems (Camacho et al., 2020).

2.4 Whale Optimization Algorithm

The Whale Optimization Algorithm (WOA) is a metaheuristic algorithm that draws inspiration from the hunting behavior of humpback whales. To find optimal solutions to complex optimization problems, WOA incorporates two distinct phases: exploration and exploitation. During the exploration phase, whales randomly traverse the solution space, allowing the algorithm to explore a wide range of potential solutions. Subsequently, in the exploitation phase, whales adjust their movements based on a designated leader, gravitating towards the most favorable solution. The algorithm also incorporates the encircling of promising regions and cooperative behaviors, similar to the whales' cooperative bubble-net hunting strategy, which intensifies the search process. By harmonizing these distinct behaviors, the WOA strikes an equilibrium between global and local search, rendering it highly effective in diverse domains that require optimization (Mirjalili and Lewis, 2016).

2.5 Gray Wolf Optimizer

The Gray Wolf Optimization (GWO) algorithm emulates the leadership hierarchy and hunting strategies observed in gray wolves in the natural world. It utilizes four categories of gray wolves, alpha, beta, delta, and omega, to replicate the hierarchical structure within the pack. Additionally, to guide the optimization procedure, the algorithm incorporates the essential steps of the hunting process, including searching for prey, encircling the prey, and attacking the prey. (Mirjalili et al., 2014).

2.6 Ant Lion Optimizer

The Ant Lion Optimization (ALO) algorithm imitates the hunting behavior of natural antlions. It incorporates five key stages of prey capture, including the random movement of ants, constructing traps, ensnaring ants in the traps, capturing prey, and rebuilding traps. By emulating these steps, the ALO algorithm aims to optimize problem-solving processes like the efficient hunting mechanisms observed in antlions (Mirjalili, 2015).

CHAPTER 3

ARDUINO TEMPERATURE CONTROL LAB OVERVIEW

This section provides an overview of the TCLab kit, its fundamental components, and its behavior.

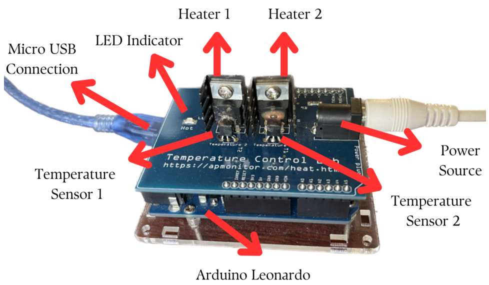
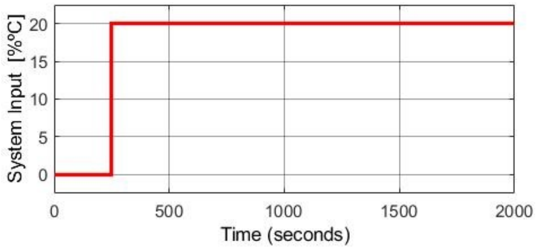


Figure 3.1: Temperature Control Laboratory

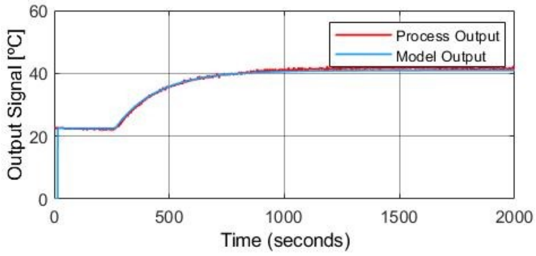
This kit is built on the Arduino Leonardo platform, as illustrated in Fig. 3.1. It requires a USB connection for communication and an electrical power supply. The kit includes two BJT heaters and their respective sensors that simulate a real-time thermal process. Also, temperature measurement is performed using voltage signals captured by the sensors, which are then converted into digital values using a 10-bit Analog-to-Digital Converter (ADC) within the Arduino. Finally, the heaters are regulated using the pulse width modulation (PWM) technique, as mentioned in reference (de Moura Oliveira et al., 2020).

3.1 TCLab Model identification

A FOPDT model was derived from the TCLab thermal process using the reaction curve procedure(Smith and Corripio, 2005). The corresponding parameters are presented in Table 1, and the validation results are illustrated in Fig. 3.2. This validation shows how effectively the model follows the behavior of the thermal process, which is useful in simulation.



(a)



(b)

Figure 3.2: Model Validation: FOPDT Model and Real process responses

Table 3.1: FOPDT Characteristic Model Parameters

Values	
Parameters	Values
K	0.9201
τ	182
t_0	15

3.2 TCLab with additional delay

The model revealed that TCLab has a dead time t_0, less than the dominant time constant τ. Considering that this study is about processes with dominant dead time, a software time delay of 185 [s] was incorporated, resulting in $t_0 = 200$ [s].

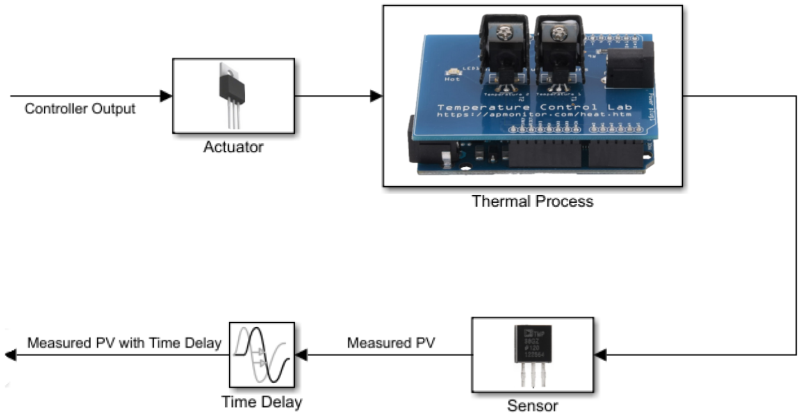


Figure 3.3: Block Diagram for the Thermal Process.

Fig. 3.3 shows the new block diagram with the additional dead time added to the original system. Therefore, the new transfer function for the FOPDT model with dominant dead time can be expressed as follows.

$$G(s) = \frac{0.92}{182s + 1} e^{-200s} \tag{3.1}$$

CHAPTER 4

CONTROLLERS APPROACHES

This section discuss the empirical approach used for implementing the Smith Predictor and Proportional Integral controllers. The complementary delay time was incorporated into the system using MathWorks Simulink, with the simulations performed on a laptop equipped with an i7 7th Generation Processor operating at 2.8 GHz and 24GB RAM. Additionally, to design the Smith Predictor, a process diagram was utilized, which is illustrated in Figure 4.1.

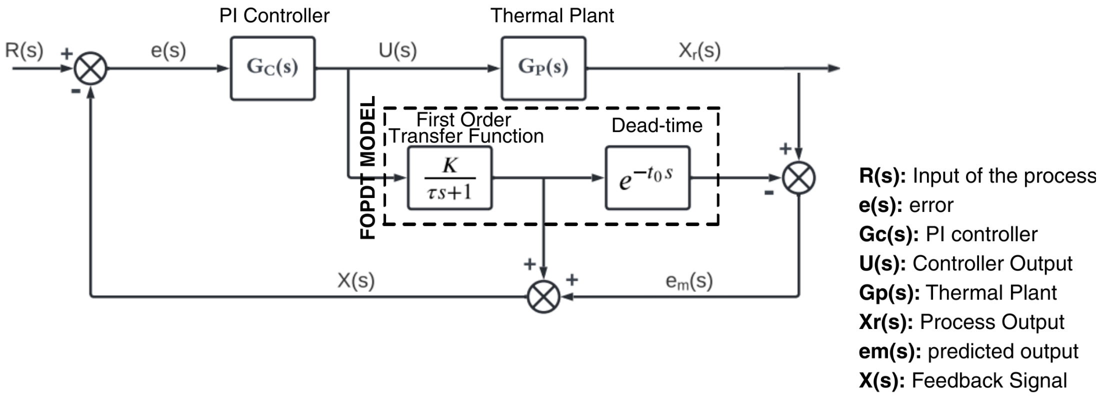


Figure 4.1: Process Diagram

4.1 PI Controller

Based on the FOPDT model, a Proportional Integral (PI) controller was implemented applying Dahlin equations to obtain the Proportional Gain (K_p) and the integral Gain (K_i). The Dahlin Method was chosen as the tuning method for the PI controller due to its frequent application in process control industries, as mentioned in (Smith and Corripio, 2005) and the calculations and parameter values obtained were specified in Table 4.1.

Moreover, the process controller output signals are depicted in Fig. 4.2, showcasing the response to a reference change.

Table 4.1: Dahlin Nominal Parameters

PI Controller		
Parameters	Equation	Values
K_p	$\frac{1}{2K}\left(\frac{\tau}{t_0}\right)$	0.5972
K_i	$\frac{1}{\tau}$	0.0054

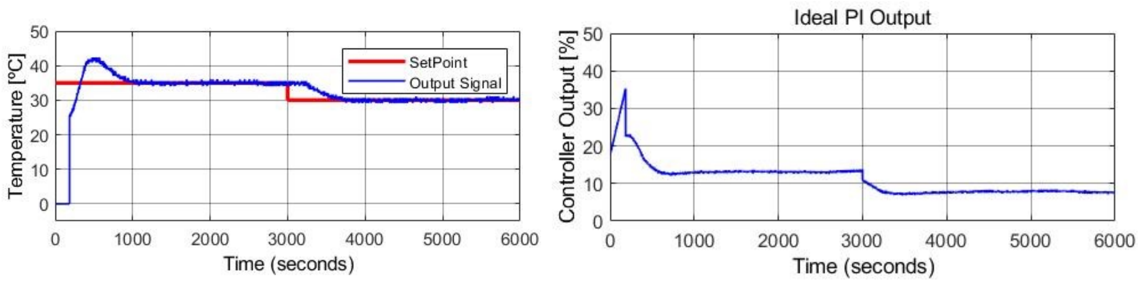


Figure 4.2: Nominal process results of the PI Controller.

Analyzing Figure 4.2, it becomes evident that the system variable is able to closely follow the set point without significant deviations or noise. Its characteristics will be seen in section VI.

4.2 Smith Predictor Controller

A second controller was developed, employing the Smith predictor approach. This controller incorporates a PI controller within the SP scheme. However, the distinction lies in the tuning method used. Equations from reference (Camacho et al., 2020), for a first-order model, with τ_c as the tuning parameter were applied, and it is noteworthy that some of the controllers discussed in (Camacho et al., 2020) consider τ_c to be equal to t_0.

Subsequently, the designed controller was tested, and the results obtained are depicted in Figure 4.3, demonstrating its satisfactory performance. The controller parameter values utilized are presented in 4.2.

Table 4.2: Nominal Values with τ_c as adjustment parameter

PI Controller for SP		
Parameters	Equation	Values
K_p	$\frac{\tau}{\tau_c}$	0.91
K_i	$\frac{1}{\tau}$	0.0054

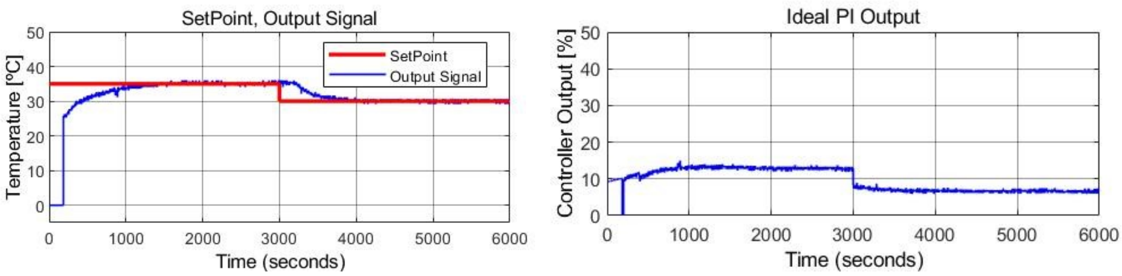


Figure 4.3: Nominal process results of Smith Predictor Controller.

Analyzing Figure 4.3, it is evident that the process variable exhibits precise tracking of the reference signal without notable deviations or noise. Additionally, the control signal effectively regulates the thermal process with a rapid settling time, displaying no overshoot or oscillation. This improved performance can be attributed to the advantageous characteristics of the Smith Predictor, which is known for its superior performance in systems with long dead-time. Its characteristics will be shown in Section VI.

Despite achieving satisfactory results, there exist notable variations in the performance characteristics between the two controllers. Consequently, in the next section, optimization algorithms are employed to explore how the compensators performance can be optimized.

CHAPTER 5

OPTIMIZATION ALGORITHMS

This section discusses the implementation of the three metaheuristic optimization algorithms described before.

The PI and SP schemes were optimized based on simulation using these algorithms to minimize the integral square error by adjusting the reference point. The optimized parameters obtained were subsequently tested using the TCLab. Additionally, in the optimization process, each procedure was assigned the same values (except for the boundaries) to compare them. This was done to obtain a new population that potentially offered improved solutions to replace the previous generation (nominal methods).(Mejía et al., 2022, Arora, 2019).

Parameters to run the optimization (Kp, Ki):

- Search Agents (N) = 30
- Max iterations (Maximum number of generations) = 30
- Lower boundaries = [0.0001, 0.0001]
- Upper boundaries: for PI = [5, 5] for SP = [1, 1]

Increasing N and Max iteration values did not yield any advantages. Therefore, a value of 30 was sufficient to optimize the process without sacrificing information. The PI controller had wider limits, as they did not affect the implementation in the thermal process. On the other hand, for the SP controller, narrower limits were established, with a difference 10% from the Nominal tuned Ki serving as a reference. This was done to prevent oscillations caused by the limitations of the TCLab PWM.

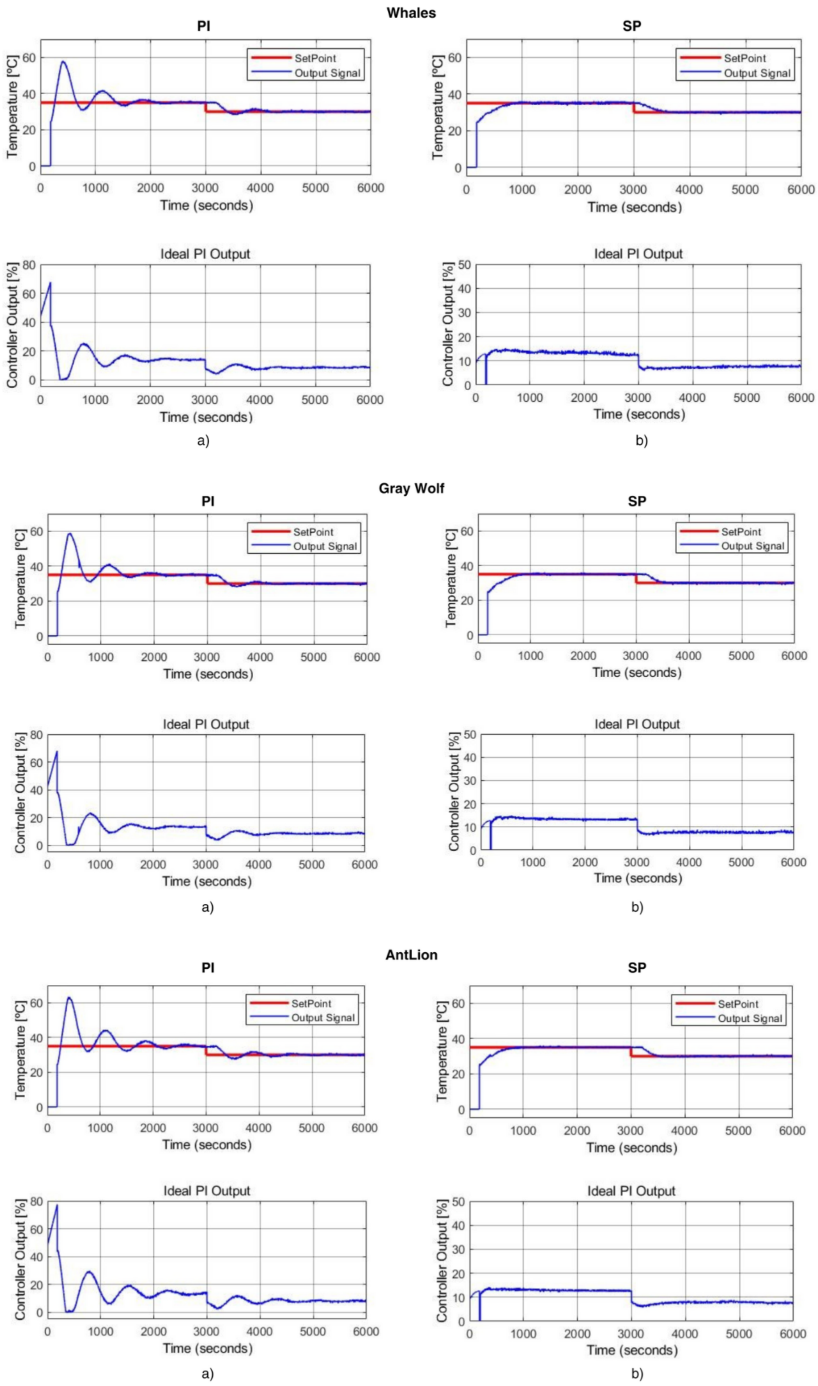


Figure 5.1: Processes Results with WOA, GWO and ALO.

Figure 5.1 presents the results of applying heuristic optimization algorithms to controllers in the real-time thermal process.

CHAPTER 6

EVALUATION AND CONTRASTS

In this section, several tests are considered for evaluation of purposes. Table IV shows the values of the tuning parameters and the time spent to obtain these optimal values, and in Table V, the performance indices ISE, $ISCO$, $Mp(\%)$ so as the ts are shown to see the dynamic performance for the response and controller action.

Table 6.1: Algorithms Comparisons

PI Scheme	K_p	K_i	Opt. Time (s)
Nominal	0.5972	0.0054	—
Whales	1.2446	0.0030	134
Gray Wolves	1.1947	0.0034	159
AntLion	1.3738	0.0033	163
SP Scheme	K_p	K_i	Opt. Time (s)
Nominal	0.91	0.0054	—
Whales	0.9	0.0098	303
Gray Wolves	0.9	0.0097	368
AntLion	0.9	0.0095	414

The SP scheme with the Whale Optimization Algorithm (WOA) achieved the fastest optimization time and the lowest Integral Square Error (ISE) in all cases. The lowest ISE value obtained was 10430, representing a significant improvement of 75% compared to the nominal PI scheme and a 23% better performance than the nominal SP method. Also, the PI scheme with WOA had a 1.41% lower ISE than GWO and 7.67% compared to ALO. On the other hand, the SP scheme with WOA had a 0.4% lower ISE compared to GWO and 1.23% compared to ALO.

These findings highlight the superior performance and control effectiveness of the Smith Predictor using WOA to minimize the integral square error.

Table 6.2: Performance Indices Comparison

PI Scheme	ISE	$ISCO$	$M_p\%$	$t_s(s)$
Nominal	4,2E+05	1.001E+07	4,63	1500
Whales	3,49E+05	1,06E+07	22,85	2485
Gray Wolves	3,54E+05	1,06E+07	25,71	2520
AntLion	3,78E+05	1,09E+07	37,14	2850
SP Scheme	ISE	$ISCO$	$M_p\%$	$t_s(s)$
Nominal	13580	1,21E+06	0	1400
Whales	10430	1,25E+06	0	1200
Gray Wolves	10470	1,25E+06	0	1000
AntLion	10560	1,25E+06	0	1135

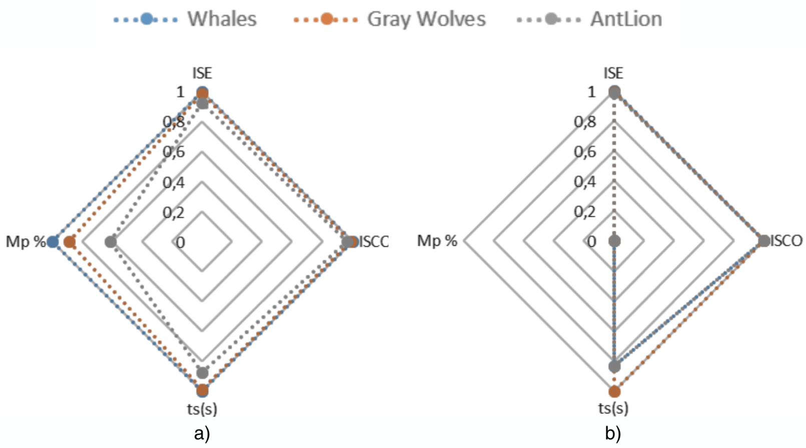


Figure 6.1: Normalized Performance Comparison a) PI b) SP: Integral Square Control Output (ISCO), Integral Square Error (ISE), Settling Time (t_s) and overshoot (M_p).

Fig. 6.1 shows the superior performance of SP in all elements. Additionally, the optimization with WOA demonstrated superiority in most aspects except for the settling time in the SP controller. However, considering that the cost function was the squared integral error, this algorithm outperformed all tests.

CHAPTER 7

CONCLUSIONS

This study aimed to explore metaheuristic algorithms for finding optimal tuning parameters for PI and SP controllers in systems with a long dead time using the TCLab kit. The analysis compared three algorithms (WOA, GWO, and ALO) and assessed their ability to improve controller performance by minimizing ISE through parameter tuning. Different performance indices were used to observe variations in behavior between schemes and their metaheuristic optimizations. The results demonstrated that all algorithms successfully found solutions. However, WOA stood out by reducing the integral square error. It is important to note that optimization with the lowest values, such as ISE, can perform a high overshot value, which is sometimes not the most suitable choice. Therefore, the cost function can be done by balancing factors that consider most aspects of performance.

BIBLIOGRAPHY

Arora, R. K. (2019). *Optimization: Algorithms and Applications*. Chapman and Hall/CRC.

Camacho, O. and Leiva, H. (2020). Impulsive semilinear heat equation with delay in control and in state. *Asian Journal of Control*, 22(3):1075–1089.

Camacho, O. and Martínez, J. (2017). *Procesos con Retardo de Tiempo Dominante. Diseño, Análisis y Comparación de Estrategias de Control*. Editorial Académica Española.

Camacho, O., Rosales, A., and Rivas, F. (2020). *Control de Procesos*. Escuela Politécnica Nacional, 1 edition.

Darwish, A. (2018). Bio-inspired computing: Algorithms review, deep analysis, and the scope of applications. *Future Computing and Informatics Journal*, 3(2):231–246.

de Moura Oliveira, P., Hedengren, J. D., and Boaventura-Cunha, J. (2021). Bridging theory to practice: Feedforward and cascade control with tclab arduino kit. In *CONTROLO 2020: Proceedings of the 14th APCA International Conference on Automatic Control and Soft Computing, July 1-3, 2020, Bragança, Portugal*, pages 23–32. Springer.

de Moura Oliveira, P., Hedengren, J. D., and Rossiter, J. (2020). Introducing digital controllers to undergraduate students using the tclab arduino kit. *IFAC-PapersOnLine*, 53(2):17524–17529. 21st IFAC World Congress.

Mejía, C., Salazar, E., and Camacho, O. (2022). A comparative experimental evaluation of various smith predictor approaches for a thermal process with large dead time. *Alexandria Engineering Journal*, 61(12):9377–9394.

Mirjalili, S. (2015). The ant lion optimizer. *Advances in Engineering Software*, 83:80–98.

Mirjalili, S. and Lewis, A. (2016). The whale optimization algorithm. *Advances in Engineering Software*, 95:51–67.

Mirjalili, S., Mirjalili, S. M., and Lewis, A. (2014). Grey wolf optimizer. *Advances in Engineering Software*, 69:46–61.

Normey-Rico, J. E. and Camacho, E. F. (2007). *Control of dead-time processes*, volume 462. Springer.

O'Dwyer, A. (1999). Pi and pid controller tuning rules for time delay processes: a summary. part 2: Pid controller tuning rules.

Smith, C. A. and Corripio, A. B. (2005). *Principles and Practices of Automatic Process Control*. John wiley & sons.