

## Diseño De Técnicas De Control Clásico E Inteligente Para Regular Hornos Eléctricos De Tratamiento Térmico

*Design Of Classic And Intelligent Control Techniques To Regulate Electric Heat Treatment Ovens*

Claiver Elian Villabon Castro <sup>1\*</sup>   
Universidad de Pamplona

Fabian Enrique Casares Pavia <sup>1</sup>   
Universidad de Pamplona

Diego Andrés Rojas González <sup>1</sup>   
Universidad de Pamplona

Oscar J. Suarez <sup>1</sup>   
Universidad de Pamplona

© 2021 Universidad de Córdoba. Este es un artículo de acceso abierto distribuido bajo los términos de la licencia Creative Commons Attribution License, que permite el uso ilimitado, distribución y reproducción en cualquier medio, siempre que el autor original y la fuente se acreditan.

---

<sup>1</sup> Universidad de Pamplona, Facultad de ingenierías y arquitectura, Ingeniería Mecatrónica Pamplona, Norte de Santander, Colombia.  
E-mail: {claiver.villabon, fabian.casares, diego.rojas, oscar.suarez}@unipamplona.edu.co.

## RESUMEN

Se investigó una planta que consiste en un horno de tratamiento térmico, el cual se diseñaron diferentes estrategias de control para analizar el comportamiento en la salida de ella buscando mejorar el controlador anterior creado, entre las estrategias tenemos los distintos controladores clásicos y controladores difusos al igual que el uso de la clonación de los controladores mediante redes neuronales artificiales.

**PALABRAS CLAVE:** Tratamiento térmico, difuso, controladores, red neuronal.

## ABSTRACT

A plant consisting of a heat treatment furnace was investigated, which different control strategies were designed to analyze the behavior at the output of it, seeking to improve the previous controller created, among the strategies we have the different classic controllers and fuzzy controllers like the use of cloning controllers using artificial neural networks.

**KEYWORDS:** heat treatment, Fuzzy, Controller, neural network.

## INTRODUCCIÓN

En el presente artículo se diseñan y se implementa controladores con estrategias de control clásico y difusos para el mejoramiento de una planta de un horno eléctrico de tratamientos térmicos, los datos de la planta son adquiridos del artículo "Modelado y diseño de estrategias de control avanzado para hornos eléctricos de tratamientos térmicos" el cual nos proviene de su modelamiento para el análisis de la planta.

se implementarán los cuatro controladores clásicos y difusos, P, PI, PD y PID, analizando sus ventajas y compararlos entre ellos así también con los resultados obtenidos en el artículo base, igualmente se realiza una clonación con redes neuronales artificiales (RNA) y el uso de un código para controlador fuzzy para implementación en tarjetas embebidas.

## 1. PLANTA Y FUNCIONAMIENTO

Los tratamientos térmicos se aplican en la industria con el fin de cambiar o agregar propiedades a ciertas piezas; En los tratamientos aplicados a metales, se involucran varios procesos de calentamiento alternados con procesos de enfriamiento, lo que concluye en la modificar de características del material tales como la resistencia, dureza o ductilidad.

El control de temperatura de un horno industrial es un proceso no lineal, entonces se desarrolla una estrategia de control que se ajuste a los requerimientos de los tratamientos térmicos.

El horno referencia para el desarrollo del artículo es el horno eléctrico de resistencias TKF, fabrica Santa Bárbara; En este horno comúnmente se desarrollan tratamientos como son austenización, temple, alivio de tensiones y normalizados, sometiendo las

piezas hasta temperaturas que alcanzan los 1150°C.

El calentamiento del horno se logra por medio de resistencias eléctricas en espiral, fabricadas con alambre Kanthal AF de 4.0m.m diámetro, enrollado en núcleo de 1” diámetro.



Figura 1. Horno de tratamiento térmico.

La potencia nominal del horno es de 270KW, distribuida en ocho grupos de 33.75KW, cada grupo está conformado por dos filas laterales y una del fondo; por lo tanto, en una pared lateral se alojan cuatro grupos de resistencias y en la otra pared los otros cuatro.

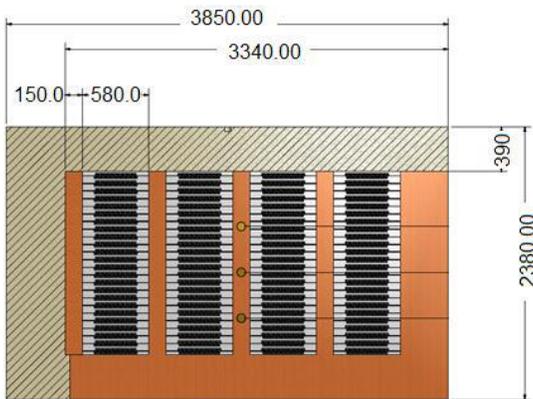


Figura 2. Resistencias actuadoras del horno de tratamiento térmico

El horno eléctrico está compuesto por una solera móvil. La superficie superior del carro está cubierta por ladrillo aislante y refractario con un espesor total de 350mm. Tiene un

ancho útil de 1.500mm y un largo útil de 3.000mm Las dimensiones del techo son: 3850mm \* 2410 mm Las dimensiones de la pared de fondo son: 2410mm \* 2380mm. [1]

## 2. MODELADO DE LA PLANTA

Debido a que la planta no es nuestra y proviene de una tesis desarrollada por Nicolef Lorena Avella Molano, se cita el artículo en el cual esta perfectamente detallada el modelado de la planta del cual se extrajo la función de transferencia y de la cual se compara los valores del controlador desarrollado.

$$G_o(s) = \frac{0.043 * s^2 + 2.056 * s + 1.511}{s^4 + 67.422 * s^3 + 1310.556 * s^2 + 8132.778 * s + 13639}$$

A partir de esta función de transferencia en la tesis se desarrolla la siguiente planta generada por el método clásico de Ziegler Nichols de lazo abierto. [1]

PARÁMETRO	KP	KI	KD
Controlador PID	137.06	22.39	209.78

Tabla 1. variables de control de la tesis base

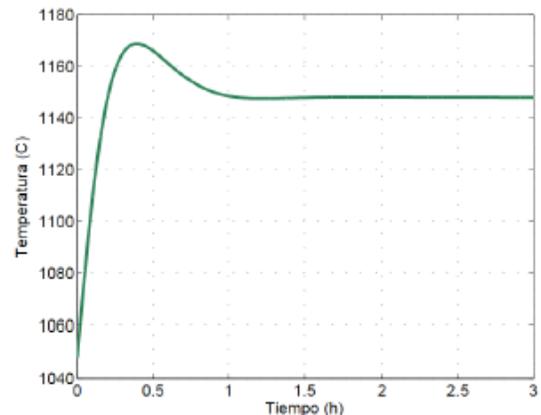


Figura 3. respuesta del controlador de la tesis base

## 3. DESARROLLO DE CONTROLADORES

El diseño de controladores es una práctica que genera diferentes resultados

dependiendo del diseñador y el tipo de controlador que se desee implementar, por lo que el comportamiento de una planta puede variar en sus ventajas, por lo que presentamos diferentes tipos de controladores para la planta y posteriormente concluir sobre los que se comportan de una buena forma.

### 3.1. Método Ziegler Nichols lazo cerrado

En este método buscamos oscilaciones sostenidas para hallar la ganancia crítica y el periodo crítico.

Los controladores desarrollados en las tablas con texto de color gris, son controladores que no generaron una respuesta satisfactoria por lo que no se graficaron. [2]

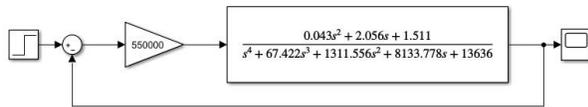


Figura 4. Diagrama simulink en lazo cerrado Ziegler Nichols

PCR	KCR
0.041657	550000

Tabla 2. Periodo crítico y ganancia crítica

PARÁMETRO	K P	KI	K D
Controlador P	27 5000	-	-
Controlador PI	24 7500	71305 09.939	-
Controlador PD	44 0000	-	2 323.2
Controlador PID	33 0000	15850 144.09	1 742.4

Tabla 3. Constantes de los controladores clásicos por Ziegler Nichols

Para el controlador PID se le modificó el Ki para obtener una mejor respuesta

PARÁMETRO	KP	KI	K D
Controlador PID	33 0000	200 0000	17 42.4

Tabla 4. Modificación del controlador PID



Figura 5. Respuesta de los controladores clásicos Ziegler Nichols



Figura 6. Mejores respuestas de los controladores de Ziegler Nichols

A los mejores controladores se le han añadido algunas variaciones.

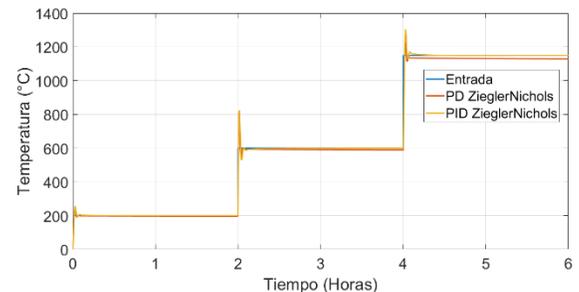


Figura 7. Variación a los mejores controladores de Ziegler Nichols

### 3.2. Método Ziegler Nichols.m

La herramienta ZIEGLERSNICHOLS.m es una librería con métodos agregados al control clásico que nos permite añadir

algunas mejoras al controlador. [3]

PARÁMETRO	KP	KI	KD
Controlador P	275000	-	-
Controlador PI	<b>247500</b>	<b>7.1297e+06</b>	-
Controlador PD	440000	-	2.2911e+03
Controlador PID 'ClassicPID'	330000	1.5844e+07	1.7184e+03
Controlador PID 'SomeOvershoot'	1.8333e+05	8.8020e+06	2.5457e+03
Controlador PID 'NoOvershoot'	110000	5.2812e+06	1.5274e+03

Tabla 5. Constantes de los controladores de ZieglerNichols.m

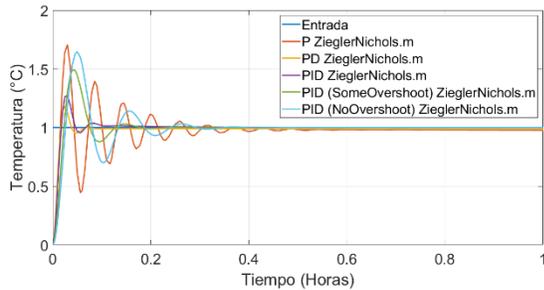


Figura 8. Respuestas de los controladores de ZieglerNichols.m

A los mejores controladores se le han añadido algunas variaciones.

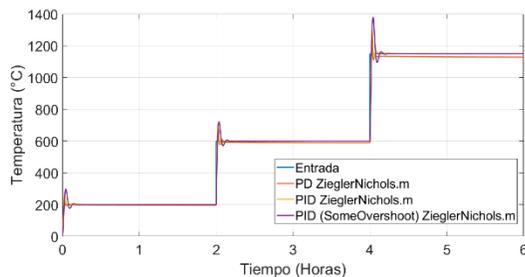


Figura 9. Variaciones a los controladores ZieglerNichols.m

### 3.3. Método pidtune de Matlab

La herramienta nativa de Matlab es pidtune al ingresar la función de transferencia ella nos genera los siguientes controladores. [4]

PARÁMETRO	KP	KI	KD
Controlador P	275000	-	-
Controlador PI	-	<b>6.4e+03</b>	-
Controlador PD	4.67e+05	-	3.46e+03
Controlador PID	-	6.29e+03	-

Tabla 6. Constantes de controladores de pidtune

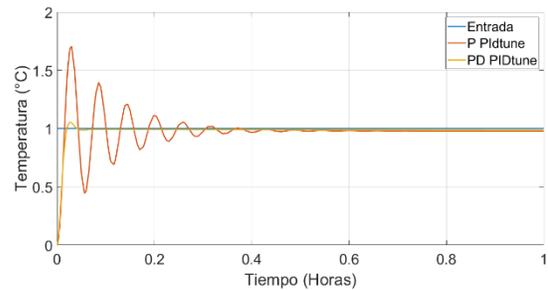


Figura 10. Respuesta de los controladores de pidtune

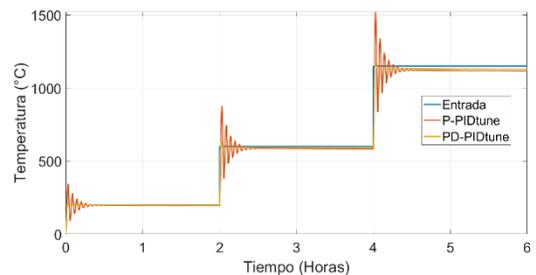


Figura 11. Variaciones de controladores por pidtune

### 3.4. Controladores difusos

Se desarrollaron diferentes controladores utilizando la herramienta de fuzzy Logic de Matlab. [5]

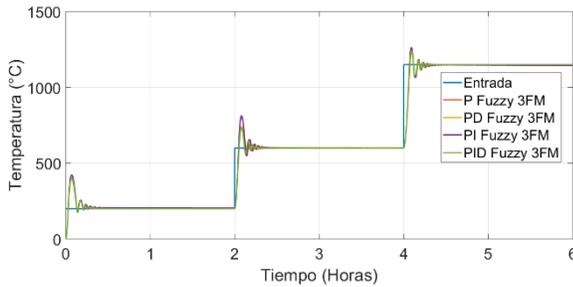


Figura 12. Respuesta con variaciones de los controladore fuzzy.

El diseño del controlador PD difuso se creó con las siguientes características

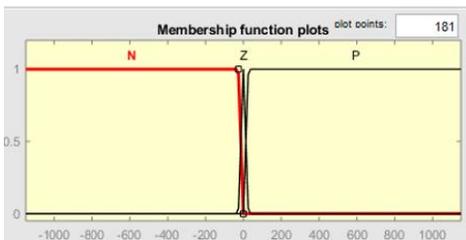


Figura 13. entrada error. PD, °C

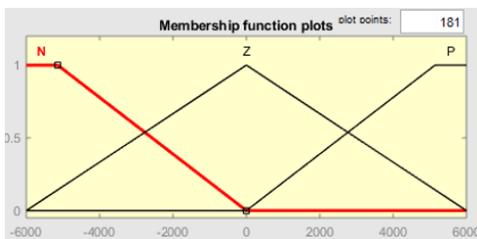


Figura 14. entrada derivada del error, PD, °C

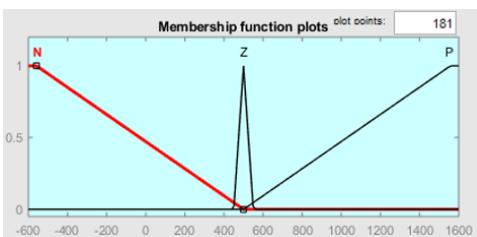


Figura 15. salida, PD, °C

1. If (Error is N) and (DevError is N) then (Salida is N) (1)
2. If (Error is N) and (DevError is Z) then (Salida is N) (1)
3. If (Error is N) and (DevError is P) then (Salida is N) (1)
4. If (Error is Z) and (DevError is N) then (Salida is Z) (1)
5. If (Error is Z) and (DevError is Z) then (Salida is Z) (1)
6. If (Error is Z) and (DevError is P) then (Salida is Z) (1)
7. If (Error is P) and (DevError is N) then (Salida is P) (1)
8. If (Error is P) and (DevError is Z) then (Salida is P) (1)
9. If (Error is P) and (DevError is P) then (Salida is P) (1)

Figura 16. reglas, PD

El diseño del controlador PID difuso se creó con las siguientes características

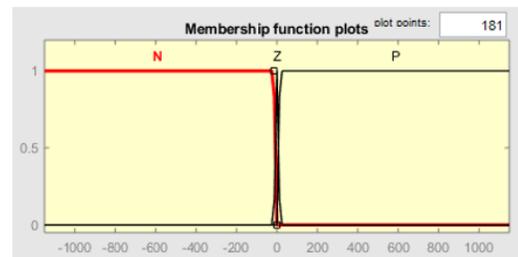


Figura 17. entrada error. PID, °C

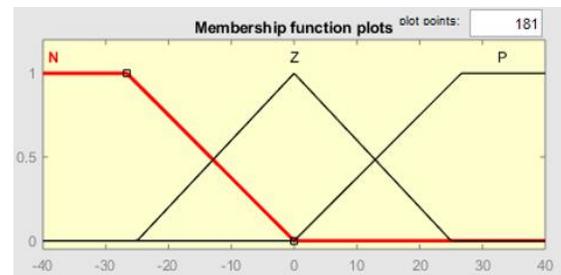


Figura 18. entrada integral, PID, °C

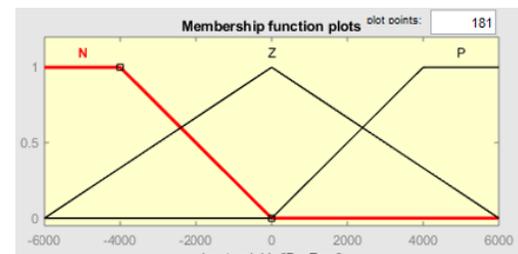


Figura 19. entrada derivada del error, PID, °C

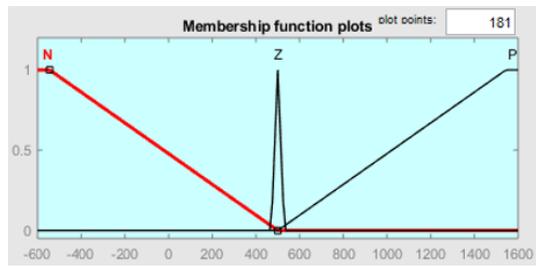


Figura 20. salida, PID, °C

1. If (Error is N) and (IntError is N) and (DevError is N) then (Salida is N) (1)
2. If (Error is N) and (IntError is N) and (DevError is Z) then (Salida is N) (1)
3. If (Error is N) and (IntError is N) and (DevError is P) then (Salida is N) (1)
4. If (Error is N) and (IntError is Z) and (DevError is N) then (Salida is N) (1)
5. If (Error is N) and (IntError is Z) and (DevError is Z) then (Salida is N) (1)
6. If (Error is N) and (IntError is Z) and (DevError is P) then (Salida is N) (1)
7. If (Error is N) and (IntError is P) and (DevError is N) then (Salida is N) (1)
8. If (Error is N) and (IntError is P) and (DevError is Z) then (Salida is N) (1)
9. If (Error is N) and (IntError is P) and (DevError is P) then (Salida is N) (1)
10. If (Error is Z) and (IntError is N) and (DevError is N) then (Salida is Z) (1)
11. If (Error is Z) and (IntError is N) and (DevError is Z) then (Salida is Z) (1)
12. If (Error is Z) and (IntError is N) and (DevError is P) then (Salida is Z) (1)
13. If (Error is Z) and (IntError is Z) and (DevError is N) then (Salida is Z) (1)
14. If (Error is Z) and (IntError is Z) and (DevError is Z) then (Salida is Z) (1)
15. If (Error is Z) and (IntError is Z) and (DevError is P) then (Salida is Z) (1)
16. If (Error is Z) and (IntError is P) and (DevError is N) then (Salida is Z) (1)
17. If (Error is Z) and (IntError is P) and (DevError is Z) then (Salida is Z) (1)
18. If (Error is Z) and (IntError is P) and (DevError is P) then (Salida is Z) (1)
19. If (Error is P) and (IntError is N) and (DevError is N) then (Salida is P) (1)
20. If (Error is P) and (IntError is N) and (DevError is Z) then (Salida is P) (1)
21. If (Error is P) and (IntError is N) and (DevError is P) then (Salida is P) (1)
22. If (Error is P) and (IntError is Z) and (DevError is N) then (Salida is P) (1)
23. If (Error is P) and (IntError is Z) and (DevError is Z) then (Salida is P) (1)
24. If (Error is P) and (IntError is Z) and (DevError is P) then (Salida is P) (1)
25. If (Error is P) and (IntError is P) and (DevError is N) then (Salida is P) (1)
26. If (Error is P) and (IntError is P) and (DevError is Z) then (Salida is P) (1)
27. If (Error is P) and (IntError is P) and (DevError is P) then (Salida is P) (1)

Figura 21. Reglas, PID

### 3.5. Comparación de controladores

En la siguiente figura podemos detallar las mejores respuestas de los controladores.

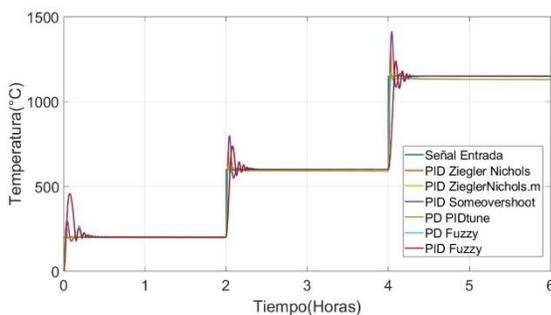


Figura 22. mejores controladores

Podemos ver que el mejor controla es el PD de la función ZieglersNichols.m, estabiliza rápido y su overshoot es el menor entre los controladores.

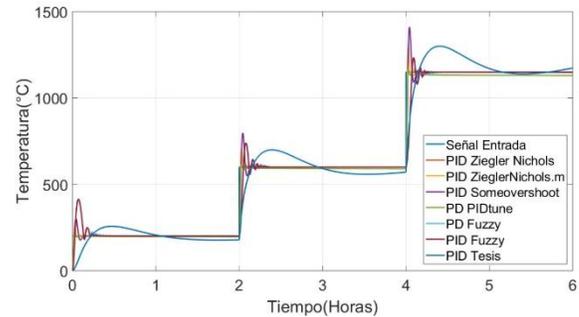


Figura 23. Mejores controladores y respuesta a mejorar

La comparación entre los controladores diseñados en el presente articulo presentan una mejoría notable en comparación al controlador original de la planta, estabilizándose 10 veces más rápido, en algunos presentando menor o igual overshoot, y menor tiempo de subida.

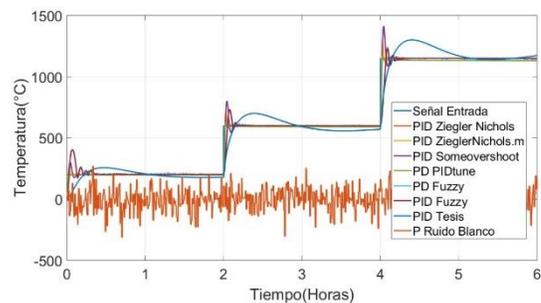


Figura 24. Respuesta de los controladores a perturbaciones

Esta planta no es susceptible a perturbaciones debido a que maneja temperaturas de miles de grados centígrados, pero en un caso simulado con perturbaciones con valores de 300 grados, notamos que los controladores diseñados son robustos para el tipo de perturbación que le estamos ingresando logrando así que regulen sin dejarse afectar en mayor medida por las situaciones externas que puedan llegar a afectar el comportamiento de la planta.

### 3.6. Clonación de controladores

La clonación de los controladores se lleva acabo con redes neuronales artificiales en

este caso escogimos el controlador PID fuzzy y el PID por el método de ZieglerNichols.m.

Una red neuronal es un modelo de computación cuya estructura de capas se asemeja a la estructura interconectada de las neuronas en el cerebro, con capas de nodos conectados. Una red neuronal puede aprender de los datos, de manera que se puede entrenar para que reconozca patrones, clasifique datos y pronostique eventos futuros. [6], [7], [8], [9].

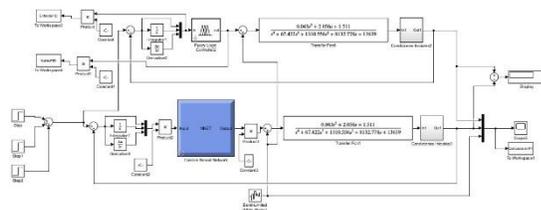


Figura 25. Diagrama de bloques, recolección de datos controlador PIDfuzzy e implementación de la RNA

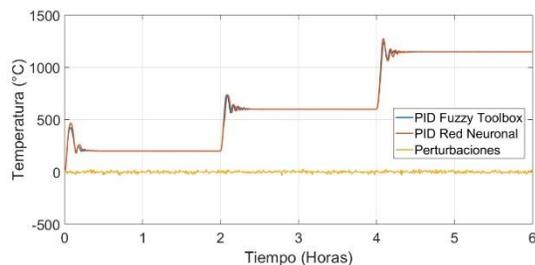


Figura 26. Respuesta de clonación de PIDfuzzy con la RNA

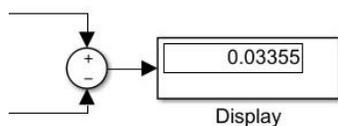


Figura 27. Error entre la clonación y el controlador PIDfuzzy original

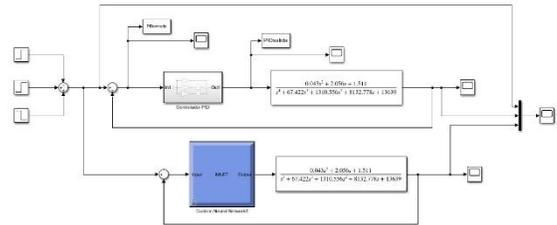


Figura 28. Diagrama de bloques, recolección de datos controlador PID Ziegler Nichols e implementación de la RNA

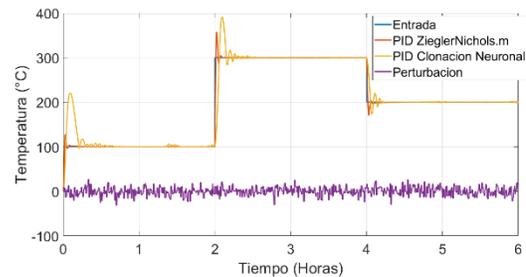


Figura 29. Respuesta de clonación de PID Ziegler Nichols con la RNA

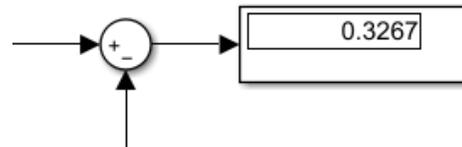


Figura 30. Error entre la clonación y el controlador PID Ziegler Nichols original

Se observan en la figura 27 y 30 unos errores en la clonación que no afectan en sobremanera el comportamiento de la planta, dando así que la RNA es una opción viable a la hora de la implementación.

### 3.7. Código controlador fuzzy

Gracias a la implementación del controlador PIDfuzzy mediante código, se puede hacer uso de este en tarjetas embebidas lo cual hace que no se requiera del toolbox de Matlab para la implementación de estos controladores.

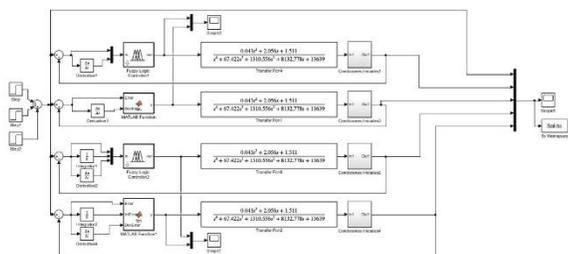


Figura 31. Diagrama de bloques, comparación entre controlador PID-PD fuzzy por código y con el uso del toolbox de Matlab

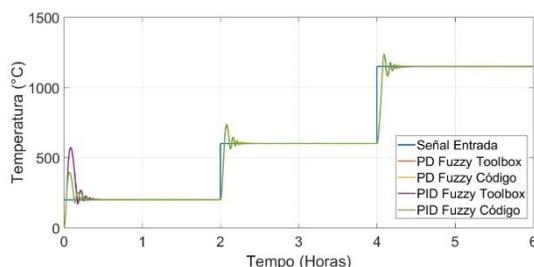


Figura 31. Respuesta de los controladores mediante código y toolbox

## 5. CONCLUSIONES

Con los mejores controladores diseñados se mejoró bastante el tiempo de respuesta que presentaba el control original, Significando un menor consumo de energía y tiempo.

En el diseño de los controladores clásicos por el método de Ziegler Nichols el de tipo PI no se obtiene una respuesta que se pueda implementar por esta razón se descartaron.

Respecto al overshoot presente, tenemos distintos puntos de vista, unos sobrepasan a la inicial pudiendo causar daños en la pieza por sobre pasar la temperatura sin embargo tenemos controladores con uno inferior generando un mejor tratamiento térmico.

Con el código se obtiene una respuesta muy similar a la obtenida mediante el uso de la herramienta Fuzzy logic de Matlab. Lo que nos da que es un controlador que se puede utilizar en tiempo real mediante tarjetas embebidas de diferente tipo lo cual hace más

versátil implementación.

El método de clonación por redes neuronales es muy útil para pasar el controlador de Matlab a otros lenguajes de programación, en los cuales se puede implementar RNA.

## REFERENCIAS

- [1]. Avella Molano, N., (2015). Modelado y diseño de estrategias de control avanzado para hornos eléctricos de tratamientos térmicos, universidad pedagógica y tecnológica de Colombia, (consultado 10 julio 2021).
- [2]. Avella Molano, N., (2015). Modelado y diseño de estrategias de control avanzado para hornos eléctricos de tratamientos térmicos, universidad pedagógica y tecnológica de Colombia, (consultado 10 julio 2021).
- [3]. Ortega, J. Y. (2016). Estructura ecológica principal de la cuenca del Río Pamplonita. Revista Ambiental Agua, Aire y Suelo. ISSN 1900-9178. Volumen (7), Numero (1). DOI: <https://doi.org/10.24054/19009178.v1.n1.2016.3258>.
- [4]. Ogata, K. (2004). Ingeniería de Control Moderna, Prentice Hall, Cuarta edición, Madrid.
- [5]. O. Suarez, A. Pardo and E. Sanchez, "Intelligent control for the exciter of a synchronous machine with closed loop voltage," World Autom. Congr. Proc., vol. 2016-Octob, no. July, pp. 1–6, 2016, doi: 10.1109/WAC.2016.7582983.

- [6]. F. A. Guasmayan y N. A. González, “Estado del arte de redes educativas para el intercambio de conocimientos en robótica educativa”, *Ingeniería E Innovación*, vol. 7, n.º 2, pp. 17-21, 2019. <https://doi.org/10.21897/23460466.1784>
- [7]. [6] PID tuning algorithm for linear plant model - MATLAB pidtune - MathWorks América Latina. (s.f.). MathWorks. <https://la.mathworks.com/help/control/ref/lti.pidtune.html>
- [8]. Fuzzy Logic Designer - MATLAB & Simulink - MathWorks América Latina. (s. f.). MathWorks. <https://la.mathworks.com/help/fuzzy/building-systems-with-fuzzy-logic-toolbox-software.html>
- Hurtado, C., Corte, C., & Triana, M. (2017). Estimación del carbono almacenado en el bosque natural en la 366 *Rev. Biol. Trop. (Int. J. Trop. Biol.)* • Vol. 69(1): 352-368, March 2021 cuenca media-baja municipio de río Quito Chocó, Colombia. *Ingeniería e Innovación*, 5(1), 32-42. DOI:10.21897/23460466.1102
- [9]. Create custom shallow neural network - MATLAB network - MathWorks América Latina. (s. f.). MathWorks. <https://la.mathworks.com/help/deeplearning/ref/network.html>
- [10]. Suarez, O. J., Vega, C. J. ., Sanchez, E. N. ., González Santiago, A. E. ., Rodríguez Jorge, O. ., & Pardo Garcia, A. (2020). Abnormal P53 degradation and apoptosis induction in P53-MDM2 network using pinning control strategy. *Revista Colombiana de Tecnologías de Avanzada*, 2(32), 1-7. <https://doi.org/10.24054/16927257.v32.n32.2018.101>
- [11]. Rios, Y. Y., García-Rodríguez, J., Sanchez, E. N. ., Alanis, A. Y. ., Ruiz Velázquez, E. ., & Durán Acevedo, C. M. . (2020). Treatment for T1DM patients using neuro-fuzzy inverse optimal control algorithm: a rapid prototyping implementation. *Revista Colombiana de Tecnologías de Avanzada*, 2(32), 26-33. <https://doi.org/10.24054/16927257.v32.n32.2018.104>
- [12]. Gualdrón Guerrero, O. E. ., Reales Osorio, J. J. ., & Acevedo Gauta, T. L. . (2016). Implementación de una estrategia de control difuso para aumentar la producción de crudo en pozo petrolero. *Revista Colombiana de Tecnologías de Avanzada*, 1(27), 98-103. <https://doi.org/10.24054/16927257.v27.n27.2016.358>