

Sintonización y optimización de un PID con algoritmos evolutivos

Armando Pedraza Martín, Omar Santos y Félix Castro

Universidad Autónoma del Estado de Hidalgo,
Pachuca, Hgo, México

pema09@hotmail.com, omarj@uaeh.edu.mx, fcastro@lsi.upc.edu

Resumen Actualmente la mayor parte de la industria utiliza un control automático sobre los sistemas de control de temperatura mediante un PID. Sin embargo, para que dicho control funcione adecuadamente es necesario que el PID sea sintonizado de manera eficaz para obtener los resultados deseados. El presente trabajo propone un algoritmo que logra encontrar los parámetros subóptimos de sintonización un PID que controla el proceso de una planta deshidratadora. La propuesta está basada en el algoritmo evolutivo *Hill Climbing* y tiene por objetivo encontrar los parámetros subóptimos de $\kappa\pi$, κi y $\kappa\delta$ que son programados en un PID de corte industrial para el control de temperatura de la planta, reduciendo así el consumo de energía y minimizando el error de convergencia en menor tiempo.

Palabras clave: PID, algoritmos evolutivos, Hill climbing, Ziegler-Nichols.

1. Introducción

En los últimos años hemos sido testigos de un enorme esfuerzo mundial por detener el denominado cambio climático, uno de los mayores esfuerzos ha sido el reducir la cantidad de emisiones de gases del efecto invernadero, otra tendencia ha sido el reducir el consumo de energía, ejemplo de ello es la implantación del llamado horario de verano, sin embargo, es importante contribuir en todos los ámbitos para evitar el deterioro del planeta. En este contexto, es de suma importancia reducir el consumo de energía utilizada en las plantas productivas de las empresas, en primera instancia porque la empresa reduciría sus gastos, y en segunda para contribuir con el planeta. De hecho uno de los procesos industriales que más demandan energía, es el deshidratado de alimentos. Por tal razón, en el presente artículo se trata el problema de reducir el gasto de energía en una planta deshidratadora.

El problema abordado en el presente artículo es la optimización de 3 parámetros participantes en la sintonización de un control Proporcional, Integral y Derivativo (PID), utilizando un algoritmo evolutivo. El PID se utiliza en un sistema de control de temperatura, cuyo modelo matemático se aproxima mediante un sistema lineal de primer orden con retardo en la entrada. Dicho modelo fue propuesto para sintonización de PID's en [14] y es ampliamente usado actualmente en la industria para los

mismos propósitos [15]. Debido a que la mayor parte de las industrias hoy en día utilizan controladores PID para el control de diferentes procesos [1], la implementación de algoritmos evolutivos para la obtención de parámetros subóptimos se ha vuelto una técnica de suma importancia para optimizar diferentes plantas y procesos y prueba de ello son las referencias [2-8].

La importancia de sintonizar un controlador PID en un sistema de control de temperatura radica en optimizar la energía utilizada y el desempeño del proceso en general, ya que en este tipo de procesos el consumo de energía es muy alto. Por lo que al utilizar técnicas de optimización numérica se pretenden encontrar los valores subóptimos del controlador PID para el sistema de control de temperatura, lo cual implica minimizar el uso de energía, así como optimizar el funcionamiento de la planta. Para esto, se desarrolló un prototipo experimental de deshidratadora, el cual tiene el mismo modelo (primer orden con retardo) que una deshidratadora industrial. El prototipo experimental, cuenta con una fuente de aire, una fuente de calor y un pequeño túnel de viento. La entrada de control se realiza aplicando un voltaje a los ventiladores presentes en el sistema y la temperatura actúa como la salida, que a su vez es la variable del proceso. El voltaje aplicado a los ventiladores implica producir el aire necesario para mantener una temperatura sobre una referencia establecida. Se utiliza un controlador industrial PID industrial Honeywell DC1040 para el control del sistema, el cual se optimiza numéricamente en cuanto a su sintonización. Nuestra propuesta no implica la compra de hardware o software extra, solo se optimizan las ganancias del controlador y el resultado de dicha optimización se programa directamente en el controlador.

Basándose en estudios previos y experimentación en línea, este trabajo cuenta tanto con simulaciones como resultados experimentales en tiempo real, lo que permite tener mayor veracidad en la obtención de los resultados, así como observar el comportamiento real de la planta.

En la sección 2 se describe el sistema de control de temperatura de la planta tratada en el presente artículo, así como su modelo matemático. En el apartado 3, se da una breve descripción de los algoritmos evolutivos y en particular del Hill Climbing. La solución adoptada en el artículo se presenta en la sección 4, detallando la obtención de los parámetros que permiten la sintonización correcta del PID. En el apartado 5 se presentan los resultados obtenidos en los experimentos realizados y finalmente, las conclusiones y trabajos futuros se detallan en la sección 6.

2. Sistemas de control de temperatura

La función de un sistema de control de temperatura se logra mediante una combinación de subsistemas: actuador (en este caso los ventiladores), sensor (un termopar tipo J1), controlador (en este caso un PID) y planta (deshidratadora). Dentro del lazo de control, se mide la temperatura del aire en el túnel de viento de la deshidratadora (este aire caliente deshidrata al producto), el controlador compara la temperatura del aire (Variable del Proceso, o PV de sus siglas en inglés) y envía una señal de corrección al actuador (ventilador) para introducir la cantidad de flujo de calor o enfriamiento necesario para converger al punto de referencia establecido previamente por el

usuario. Sin embargo, existen varios factores que afectan la tarea del controlador, entre ellos están, el cambio de temperatura del medio ambiente, el tiempo de retraso, la ubicación de los componentes (ventiladores, sensor, controlador) y por supuesto, la selección correcta de los parámetros del PID. La regulación de la temperatura y el ancho de banda son los más afectados por estos factores [9]. En este trabajo nos centramos principalmente en optimizar la sintonización de los tres parámetros del controlador.

2.1. Obtención del modelo de la planta

A continuación se describe brevemente como está constituida la plataforma: el prototipo de deshidratadora consistente en una caja de acrílico (1) de 30 cm de cada lado con un túnel (2) como salida. Dentro de la caja se encuentra una rejilla eléctrica (3) que se utiliza como fuente de calor y se alimenta hasta con 120 Volts CA, pero sólo se introduce un voltaje fijo de 17.5 volts CD a través de una etapa de potencia (4). Los actuadores son 3 ventiladores (5) que tienen un rango de operación de 3V - 12 V CD. El sensor de temperatura (6) es un termopar tipo J1 y se coloca en el túnel y está conectado directamente al PID (7) para recibir la temperatura dada en ese momento. Cabe mencionar que la compensación e punta fría la realiza el PID industrial, solo se tiene que programar que tipo de termopar es con el que se cuenta. En la Fig. 1 se muestran los componentes del modelo de la planta correspondientes al número ilustrado en este párrafo.

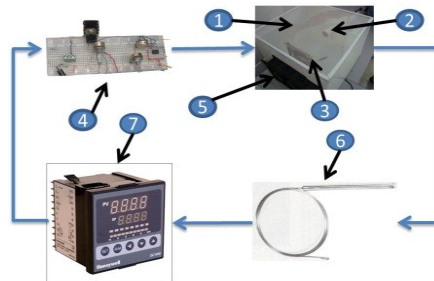


Fig. 1. Diagrama del prototipo de la planta deshidratadora

2.2. Modelo matemático

Para la obtención del modelo matemático se utilizó el método de Ziegler - Nichols en lazo abierto [14], dicho método es útil para determinar de manera experimental un modelo aproximado, mediante la respuesta de la planta en lazo abierto ante una entrada escalón unitario. La curva de respuesta se caracteriza por tres parámetros [14]:

- El tiempo de retardo, para este trabajo se emplea la variable h .
- La constante de tiempo T .
- La ganancia K del sistema.

De acuerdo con este método la planta se aproxima mediante:

$$\frac{Y(s)}{U(s)} = \frac{Ke^{-sh}}{Ts+1}. \quad (1)$$

Al utilizar este método, de manera experimental se encuentran los valores de los parámetros h , T y K , para el sistema. La temperatura externa al momento de obtener la respuesta del sistema fue de 25 °C. Utilizando el método Ziegler y Nichols se obtiene:

$$\frac{Y(s)}{U(s)} = G(s) = \frac{0.1e^{-14s}}{226s+1}. \quad (2)$$

La ecuación (1) describe un sistema lineal de primer orden con retardo en la entrada, donde $Y(s)$ es el mapeo de la temperatura y $U(s)$ es el voltaje aplicado a los ventiladores [10]. La Fig. 2 muestra la simulación de la planta (2) ante el escalón unitario, la cual es muy similar a la obtenida experimentalmente.

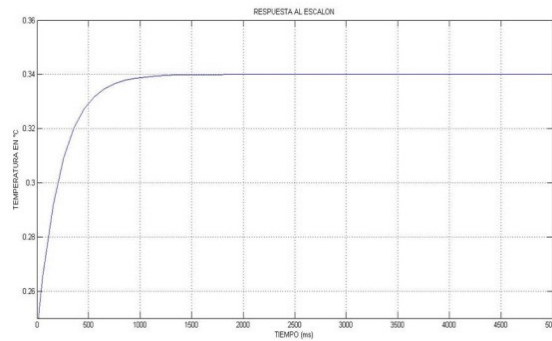


Fig. 2. Respuesta al escalón del modelo de la planta (2)

3. Algoritmos evolutivos

La idea de los algoritmos evolutivos comienza a partir de la resolución de un problema específico, por tanto se va a convertir en un proceso de búsqueda de una solución, llamado comúnmente "espacio de búsqueda". Cuando se considera la búsqueda de la mejor solución posible, se convierte en un problema de optimización. Cuando los espacios de búsqueda son pequeños o simples se pueden utilizar métodos de optimización y búsqueda clásicos como lo son métodos de búsqueda de tipo directo [11], sin embargo, éstos presentan algunas limitaciones:

1. La convergencia a una solución óptima depende de la condición inicial de la que parte la búsqueda.
2. Muchos algoritmos son propensos a verse atrapados en un óptimo local.
3. Falta de robustez, donde, un algoritmo eficaz en la resolución de un determinado problema puede no ser adecuado para otro tipo diferente de problema.
4. Algunos algoritmos no son capaces de tratar adecuadamente problemas con variables discretas.

3.1. Algoritmo de Hill Climbing

Como se mencionó, en el presente trabajo se adoptó como método de optimización el algoritmo Hill Climbing ("ascenso de colinas") por lo que a continuación se describe brevemente su funcionamiento.

Hill Climbing es un algoritmo que iterativamente permite ir obteniendo una mejor solución. Comienza con un punto en el espacio de búsqueda, una vez seleccionado este punto, si es el mejor se convierte en el punto actual, si no, es seleccionado otro punto vecino a partir del actual y es evaluado y así sucesivamente hasta encontrar la solución buscada o hasta que se satisfaga un criterio de paro. Debido a que sólo permite ir mejorando la solución, la cadena se muta, y si esta mutación proporciona un mejor punto que el actual, éste se reemplaza. El método termina cuando se encuentra un mejor punto y el actual no puede obtener mejoras o en todo caso termina cuando se alcanza el número definido de iteraciones representativamente [12].

El algoritmo Hill Climbing se considera un algoritmo voraz, debido a que es un algoritmo iterativo, donde en cada iteración se selecciona el mejor resultado para así obtener un resultado global. A este algoritmo se le considera local debido a que a partir del mejor punto actual busca únicamente en las instancias inmediatas (vecinos). De igual forma se considera una estrategia irrevocable ya que no permite regresar a otra alternativa encontrada con anterioridad [12]. De los elementos utilizados en un algoritmo evolutivo, Hill Climbing no utiliza la opción de cruce, ya que su funcionamiento se basa sólo en la mutación de un valor inmediato elegido a partir de la mejor solución actual y así evaluar cuál es la mejor solución. La Fig. 3 ilustra un diagrama con el funcionamiento general del algoritmo Hill Climbing.

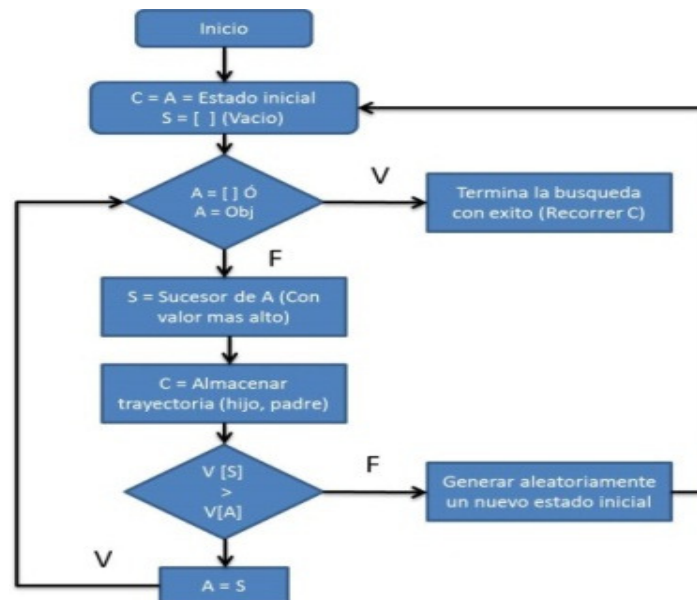


Fig. 3. Diagrama de flujo del algoritmo Hill Climbing

4. Optimización numérica del control

El objetivo principal de la optimización es encontrar los valores de los parámetros del control, los cuales dependen del sistema a controlar. En efecto, la optimización de dichos parámetros así como sus restricciones, dependen del modelo matemático de la planta, al obtener este modelo se puede utilizar un algoritmo de optimización para encontrar la solución.

Una vez elegido el algoritmo se aplica al modelo matemático en lazo cerrado con el controlador con los parámetros propuestos en cada paso del algoritmo numérico de búsqueda, aquí el usuario debe de ser capaz de reconocer si se tuvo éxito en la búsqueda de la solución. Una vez obtenidos los resultados se analizan con las condiciones de optimalidad para comprobar si el conjunto de variables es la solución al problema. Si el modelo sufre algún cambio en sus variables, el problema de optimización se debe resolver de nuevo y se repite el proceso [13].

4.1. Algoritmo Hill Climbing para la obtención de los parámetros k_p , k_i y k_d

Como ya se mencionó, Hill Climbing al no ser un algoritmo exhaustivo sólo permite encontrar una solución, en el problema abordado en el presente artículo es necesario encontrar tres parámetros, por tanto el primer valor encontrado para el controlador se genera a partir de la función de optimización; sin embargo, el segundo valor depende del resultado encontrado en el primer ciclo Hill Climbing, y para el tercero depende tanto del primero como del segundo valor encontrado, esto dentro de las funciones de optimización, por ello se realizan tres ciclos Hill Climbing, uno para k_p , otro para k_i y uno más k_d . Esto de acuerdo con el algoritmo general del PID:

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \dot{e}(t),$$

donde $e(t)$ = Referencia-Variable del Proceso. La Fig. 4 representa el diagrama de flujo del algoritmo Hill Climbing que se utiliza para describir de manera general el proceso de obtención de los tres parámetros k_p , k_i y k_d . Posteriormente, en las subsecciones siguientes se describen los pasos realizados para la obtención de los parámetros mencionados.

4.2. Ciclo Hill Climbing para k_p

A continuación se describe el proceso para la obtención del parámetro k_p .

- (a) El primer paso es inicializar las variables k_p , k_i y k_d
- (b) El siguiente paso es iniciar el primer ciclo *for*
- (c) Se simula la planta en lazo cerrado donde se obtiene el primer valor subóptimo para k_p donde el primer resultado es cero, sin embargo, cuando el valor sea mutado por el algoritmo irá cambiando dependiendo si encuentra o no una mejor solución.
- (d) Posteriormente se calcula el índice de desempeño.
- (e) A continuación se inicia un segundo ciclo *for* anidado dentro del primero.
- (f) Se realiza una mutación con un 50 % de probabilidad de cambio.

- (g) Una vez realizada la mutación se simula nuevamente la planta en lazo cerrado donde se obtiene el valor subóptimo para k_p mutado.
- (h) Se calcula el índice de desempeño para este nuevo resultado.
- (i) Ahora se comparan los resultados obtenidos entre el primer resultado obtenido y el resultado de la mutación.
- (j) Si no se encuentra una mejor solución se realiza nuevamente la mutación, en caso contrario, donde se encuentre una mejor solución se actualiza el resultado al valor subóptimo encontrado.

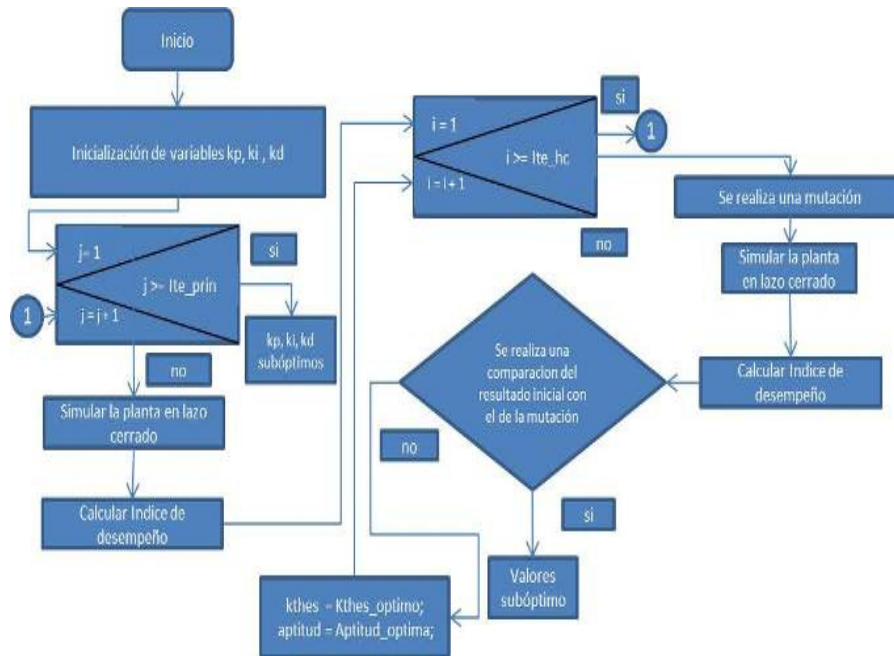


Fig. 4. Diagrama de flujo Hill Climbing

4.3. Ciclo Hill Climbing para k_i

Una vez encontrado el resultado subóptimo de k_p se realiza nuevamente el procedimiento realizado para la obtención de k_p pero ahora para k_i .

- (a) Se inicia nuevamente el primer ciclo *for* ahora para k_i .
- (b) Se simula la planta en lazo cerrado, donde, se soluciona la ecuación (2) mediante el método Ziegler – Nichols, con este resultado se crea un espacio de estado del modelo y se obtiene el lazo abierto, que a su vez es convertido a lazo cerrado, donde se obtiene la respuesta de la temperatura y el voltaje aplicado a los ventiladores.
- (c) Se calcula el índice de desempeño.

- (d) En este punto se repiten los pasos (e) y (f) realizados en el ciclo Hill Climbing para k_p .
- (e) Para obtener el resultado se repite el paso (b) realizado en este ciclo Hill Climbing para k_i , pero para el valor mutado.
- (f) Y por último se repiten los pasos (h), (i) y (j) realizados en el ciclo Hill Climbing para k_p .
- (g) Se actualiza el valor k_i subóptimo encontrado.

4.4. Ciclo Hill Climbing para kd

Una vez encontrado el resultado subóptimo de k_p y k_i , se realiza nuevamente el procedimiento realizado para la obtención de k_p pero ahora para kd .

- (a) Se inicia nuevamente el primer ciclo *for* ahora para kd .
- (b) Se simula la planta en lazo cerrado, donde, se va a realizar una discretización, esto quiere decir, convertir una señal analógica a digital, debido a que el resultado es una gráfica senoidal, se utilizan ecuaciones en diferencias para segmentar la gráfica senoidal e ir encontrando la solución.
- (c) Se calcula el índice de desempeño.
- (d) En este punto se repiten los pasos (e) y (f) realizados en el ciclo Hill Climbing para k_p .
- (e) Para obtener el resultado se repite el paso (b) realizado en este ciclo Hill Climbing para kd , pero para el valor mutado.
- (f) Y por último se repiten los pasos (h), (i) y (j) realizados en el ciclo Hill Climbing para k_p .
- (g) Se actualiza el valor kd subóptimo encontrado.

5. Resultados experimentales de control

Para culminar este trabajo se presentan los resultados experimentales obtenidos y se presenta una comparación con el método Ziegler – Nichols [14], para ello se muestran de manera gráfica los resultados.

5.1. Temperatura obtenida con Ziegler - Nichols y el método propuesto

La temperatura inicial dentro de la cámara fue de 29 °C, el valor de referencia fijado fue de 50 °C, la temperatura externa fue de 23.2 °C. Los resultados de la temperatura obtenidos con el método Ziegler - Nichols, se muestran en la Fig. 5.

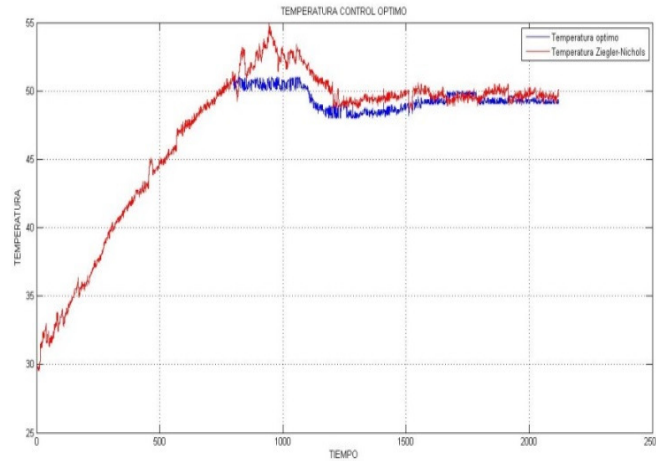


Fig. 5. Temperatura obtenida con Ziegler – Nichols y Hill Climbing

Se observa en la Fig. 5 que la temperatura se logra estabilizar con el método Ziegler – Nichols, sin embargo, una vez que se alcanza el “set point” fijado a 50 °C se llega a elevar la temperatura a los 55 °C. Los resultados obtenidos con el método propuesto estabilizan la temperatura a 50 °C, evitando sobre picos que desestabilicen la planta. Sin embargo, en un momento ésta disminuye a 49 °C, sin embargo, mantiene la estabilidad cercana a los 50 °C.

5.2. Voltaje obtenido con Z - N y el método propuesto

La Fig. 6 representa el voltaje aplicado a los ventiladores con el método Ziegler – Nichols y con el propuesto en el presente artículo.

Se observa que con el método Ziegler – Nichols el voltaje aplicado a los ventiladores se eleva hasta los 8.5 volts y logra disminuir a los cercanos 6.4 volts. Con el método propuesto se aplica un voltaje en su máximo punto de 6.4 volts, pero logra estabilizar un voltaje mínimo acercándose a los 4.5 volts. Por lo que el gasto de energía con el método propuesto es menor en comparación con Ziegler – Nichols.

Por último la Tabla 1 muestra un comparativo de los resultados obtenidos, donde se pueden observar los siguientes resultados:

- 1.- El método propuesto garantiza estabilizar la temperatura en un menor tiempo, con lo que se evita que la temperatura se pueda elevar.
- 2.- El método propuesto disminuye considerablemente el consumo de energía que, en comparación con el método Ziegler-Nichols se minimiza en 2 volts.
- 3.- Con el método propuesto se logra una convergencia del error a cero en un menor tiempo en comparación a Ziegler-Nichols.

De esta forma se demuestra que el método Ziegler-Nichols logra una estabilización en la planta; sin embargo, el método propuesto lo realiza optimizando aún más los resultados lo que indica la eficiencia del método propuesto.

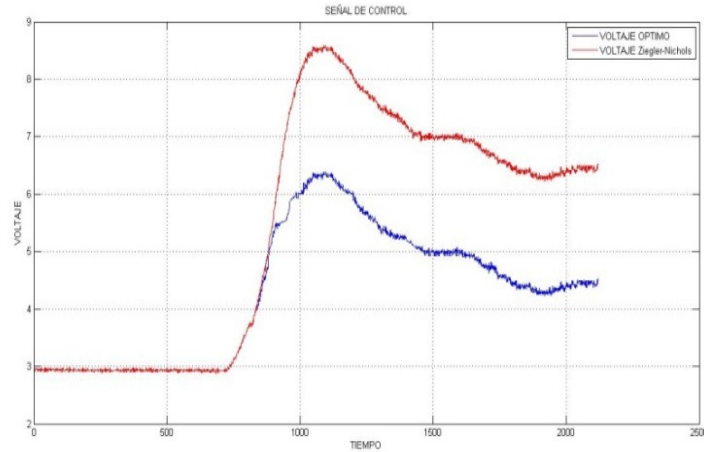


Fig. 6. Voltaje aplicado a los ventiladores con ambos métodos

Tabla 1. Resultados comparativos

	Máximo sobreimpulso	Tiempo de convergencia	Tiempo pico	Tiempo de crecimiento	Voltaje máximo aplicado
Ziegler - Nichols	5 °C	1200 seg	950 seg.	700 seg.	8.5 VCD
Método Propuesto	1 °C	700 seg	751 seg.	700 seg.	6.4 VCD

6. Conclusiones

Los resultados con el método Ziegler – Nichols y con el propuesto en el presente trabajo de investigación parecieran ser similares, sin embargo, la diferencia radica principalmente en minimizar la energía y la convergencia del error a cero, donde el método propuesto lo realiza en menor tiempo y demuestra ser subóptimo, además el método empleado, no involucra software o hardware extra en el lazo de control.

La principal aportación de este trabajo, es la metodología que permite obtener los parámetros subóptimos para un PID para poder optimizar el tiempo de respuesta y energía. Como trabajos futuros se pretende realizar experimentos en plantas más grandes, donde el gasto de energía y el tiempo de convergencia sean mayores, permitiendo una evaluación más realista de la aportación del método propuesto, así mismo se pretende comparar los resultados con otros métodos de optimización.

Referencias.

1. K J. Aström ant T. Hagglund: PID Controllers: Theory, Design and Tuning., Instrument Society of America (1995)

2. Mohammed Obaid Ali, S. P. Koh, K. H. Chong, S.K.Tiong and Zeyad Assi Obaid: Genetic Algorithm Tuning Based PID Controller for Liquid-Level Tank System. Department of Electronics and Communication Engineering, Universiti Tenaga Nasional, 11-13 October (2009)
3. Neenu Thomas, Dr. P. Poongodi: Position Control of DC Motor Using Genetic Algorithm Based PID Controller. In: Proceedings of the World Congress on Engineering 2009 Vol II, WCE 2009, London, U.K., July 1-3 (2009)
4. Musa Usman Otaru, Naji A. Al-Musabi, Samir A. Al-Baiyat: Robust PID Stabilizer Design Using Genetic Algorithms. Department of Electrical Engineering, King Fahd University of Petroleum and Minerals, Dhahran, Saudia Arabia.
5. S.Ravi, P.A.Balakrishnan: Genetic Algorithm Based Temperature Controller For Plastic Extrusion System. Department of EEE, Nandha Engineering College, Tamilnadu, India.
6. S.M.Girirajkumar, Atal.A.Kumar,N.Anantharaman: Speed Control of a Real Time D.C. Shunt Motor Using SA Based Tuning of a PID Controller. International Journal of Computer Applications (0975 .8887), Volume 5.No.11, August (2010)
7. G. Sánchez-Díaz, O. Santos-Sánchez: Suboptimal control based on hill-climbing method for time delay system, IET Control Theory Appl. 1 1441-1450 (2007)
8. A. Visioli: Optimal tuning of PID controllers for integral and unstable processes, IEEE Proceedings online no. 20010197 -Control Thieov Appl , Vol 148, No 2, March (2001)
9. Dale Patrick and Stephen W. Fardo, Industrial Process Control Systems, Delmar Publishers.
10. L. Rodríguez Guerrero: Sistema de optimización y supervisión desarrollado en Java. Caso de estudio: Proceso de temperatura con retardo, Tesis de la maestría en Automatización y Control, Universidad Autónoma del Estado de Hidalgo, Pachuca Hgo., Diciembre (2010)
11. K. Deb: Optimization for Engineering Design: Algorithms and Examples, Prentice-Hall (1995)
12. B. López Takeyas: Algoritmo Hill Climbing, Instituto Tecnológico de Nuevo Laredo, Ingeniería en Sistemas Computacionales, Octubre (2005)
13. J. Nocedal and Stephen J. Wright: Numerical Optimization, Second Edition, Springer
14. J. G. Ziegler and N. B. Nichols, Optimum Settings for Automatic Controllers, The annual meeting, New York, December 1-5 (1941)
15. Página del software industrial *Expert Tune*, <http://www.expertune.com/>, consultada en diciembre del 2012.

