

PSO and ACO algorithms applied to the tuning of a PID controller for position control of a 200 mm hydraulic cylinder

Marcelo Quispe Ccachuco, Dr.¹, Cesar Castillo Cáceres, Msc².

¹Universidad Católica de Santa María, Perú, mquispec@ucsm.edu.pe

² Universidad Católica de Santa María, Perú, ccastill@ucsm.edu.pe

Abstract— The linear positioning of hydraulic cylinders is used in industrial applications, such as the positioning of flight fins on airplanes, injection molding processes, rock drilling, etc. One of the most used control techniques is the PID (Proportional Integral Derivative) control, the problem with this technique is the tuning of its three parameters.

In this work, the experimental identification of the transfer function of the system to be controlled was first done, using the MATLAB toolbox ident. Then the PSO (Particle Swarm Optimization) algorithm was implemented in MATLAB codes. Like any optimization algorithm it requires a performance index or cost function, in this research ITAE (Integral Time Absolute Error) was used. The codes were tested with research papers found in the literature, polished until they were ready for any transfer function. These algorithms were then tested in the transfer function previously identified, achieving satisfactory results in the simulations. Finally, those values of the PID parameters found were tested in the linear positioning module of the Oleohydraulics and Pneumatics Laboratory at the Universidad Católica de Santa María. Also achieving satisfactory results in the performance of the controlled system: minimum establishment time, minimum rise time and minimum overshoot, which matched with the values obtained by data acquisition. Finally, the Ant Colony algorithm (ACO) was tested, looking for better results.

The best results were obtained with the ant colony algorithm, for 20 ants, with 1000 nodes, and 100 tours. For the system with load the best solution was $K_p = 11.01$, $K_i = 5.51$ and $K_d = 3.71$. The results were improved by making a better experimental identification of the system. The solution was also improved by increasing the number of tours and the number of nodes, increasing the computational cost. With the controller implemented, the set-up time was reduced from 2.5 to 0.6 seconds without overshoot and with an error of less than 2 mm.

Keywords—Particle Swarm Optimization (PSO), Ant Colony, Optimization (ACO), PID control, Tuning, Positioning

Digital Object Identifier (DOI):
<http://dx.doi.org/10.18687/LACCEI2020.1.1.371>
ISBN: 978-958-52071-4-1 ISSN: 2414-6390

Algoritmos PSO y ACO aplicados a la sintonización de un controlador PID para el control de posición de un cilindro hidráulico de 200 mm

Marcelo Quispe Ccachuco, Dr.¹, Cesar Castillo Cáceres, Msc.²

¹Universidad Católica de Santa María, Perú, mquispec@ucsm.edu.pe

² Universidad Católica de Santa María, Perú, ccastill@ucsm.edu.pe

Resumen—El posicionamiento lineal de un cilindro hidráulico, es una aplicación industrial que se usa en el posicionamiento de aletas de vuelo en los aviones, procesos de moldeo por inyección, perforado de rocas, etc. Una de las técnicas de control más usada es el control PID (Proporcional Integral Derivativo), el problema de esta técnica es la sintonización de sus tres parámetros.

En este trabajo primero se hizo la identificación experimental de la función de transferencia del sistema a controlar, con el toolbox *ident* de MATLAB. Luego se implementó en códigos de MATLAB el algoritmo PSO (Particle Swarm Optimization), como todo algoritmo de optimización éste requiere un índice de desempeño o función de costo, en esta investigación se utilizó ITAE (Integral Time Absolute Error). Se probaron los códigos en trabajos de investigación anteriores, se pulió hasta que quedaron listos para cualquier función de transferencia. Luego se probaron estos algoritmos en la función de transferencia identificada previamente, logrando resultados satisfactorios en las simulaciones. Finalmente esos valores de los parámetros PID hallados se probaron en el módulo de posicionamiento lineal del Laboratorio de Oleohidráulica y Neumática de la UCSM. Logrando también resultados satisfactorios en el desempeño del sistema controlado: mínimo tiempo de establecimiento, mínimo tiempo de subida y mínimo sobreimpulso, los cuales se muestran y fueron obtenidos por adquisición de datos. Posteriormente se probó el algoritmo de Colonia de Hormigas (ACO), buscando mejores resultados.

Los mejores resultados se obtuvieron con el algoritmo de colonia de hormigas, para 20 hormigas, con 1000 nodos y 100 tours. Para el sistema con carga la mejor solución fue $K_p=11.01$, $K_i=5.51$ y $K_d=3.71$. Se mejoraron los resultados al hacer una mejor identificación experimental del sistema, también se mejoró la solución al aumentar el número de tours y el número de nodos a costa de un costo computacional. Con el controlador implementado se redujo el tiempo de establecimiento de 2.5 hasta 0.6 segundos sin sobreimpulso y con un error menor a 2 mm.

Palabras clave— Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), control PID, sintonización, posicionamiento.

I. INTRODUCCIÓN

A. La Hidráulica Proporcional

La aplicación de la hidráulica se puede ver con frecuencia en nuestras vidas, en cargadores frontales, sistemas elevadores, máquinas compactadoras, prensas, etc. Estas aplicaciones consisten principalmente en cambiar de posición a un cilindro hidráulico; es decir de una posición retraída a una posición extendida y viceversa. La dificultad surge cuando se quiere posicionar este cilindro en cualquier punto de su recorrido, como en el posicionamiento de las aletas de vuelo

en los aviones. Las válvulas hidráulicas proporcionales son ampliamente utilizadas en la mayoría de las aplicaciones de control de presión/flujo ya sea para uso industrial o hidráulica móvil. Normalmente se utilizan como actuadores de flujo para pistones hidráulicos en bucles de control de nivel superior, así como para controlar directamente válvulas hidráulicas más grandes que no pueden ser controladas directamente por un solenoide [1]. Se usan generalmente como sensor de posición un transformador diferencial variable lineal (LVDT) [1].

B. La Ingeniería de Control

El otro tema que involucra un sistema de control de posición es la técnica de control a emplear. Las metodologías de diseño de control son varias y van desde la teoría de la retroalimentación cuantitativa, control adaptativo, modelo de control predictivo por retroceso no lineal, deslizamiento adaptativo control de modo, control no lineal, hasta algoritmos de optimización heurística [1]. A pesar del desarrollo de una amplia gama de avanzadas estrategias de control en las últimas cuatro décadas, los controladores PID de lejos son los más utilizados en la industria. Su popularidad se deriva de su aplicabilidad y rendimiento robusto en una amplia variedad de escenarios operativos. Por otra parte, hay una amplia comprensión conceptual del efecto de los tres términos involucrados entre operadores de plantas no-especialistas que hace que la sintonización manual sea relativamente una sencilla tarea. La mayoría de las reglas de ajuste PID desarrolladas en los últimos 50 años usan métodos de respuesta en frecuencia [2]. Ejemplos incluyen, *Regla de Ziegler – Nichols* en 1942, *symmetric optimum rule* en 1958 por Voda & Landau, *Ziegler–Nichols’ complementary rule* en 1989 por Mantz & Tacconi, *some-overshoot rule* en 1989 por Seborg, Edgar, & Mellichamp, *no-overshoot rule* en 1989 por Seborg, *refined Ziegler–Nichols rule* en 1991 por Hang, Astrom, & Ho, *integral of squared time weighted error rule* en 1993 por Zhuang & Atherton, e *integral of absolute error rule* en 1994 por Pessen [2].

Estos métodos son de simple aplicación ya que proporcionan fórmulas simples de ajuste para determinar los parámetros del controlador PID. Sin embargo, ya que solo una pequeña cantidad de información sobre la dinámica del comportamiento del proceso se usa; en muchas situaciones no proporcionan una afinación lo suficientemente buena o producen un resultado insatisfactorio en la respuesta de lazo cerrado. Por ejemplo, en la práctica, la regla de Ziegler -

Digital Object Identifier (DOI):

<http://dx.doi.org/10.18687/LACCEI2020.1.1.371>

ISBN: 978-958-52071-4-1 ISSN: 2414-6390

Nichols a menudo conduce a una respuesta oscilatoria cuando se cambia el punto de ajuste (setpoint) [3].

Se proponen muchos enfoques de sintonización para el control PID y PI para sistemas con respuestas transitorias monótonas. Recientemente se han propuesto varios meta-algoritmos de inteligencia artificial como Algoritmo Genético (GA), Optimización de Colonia de Hormigas (ACO), Optimización de Enjambre de Partículas (PSO), etc. para estimar las ganancias optimizadas de sintonización de los controladores PID [4].

C. La Planta

Un esquema del sistema de control de posición se puede apreciar en la figura 1.

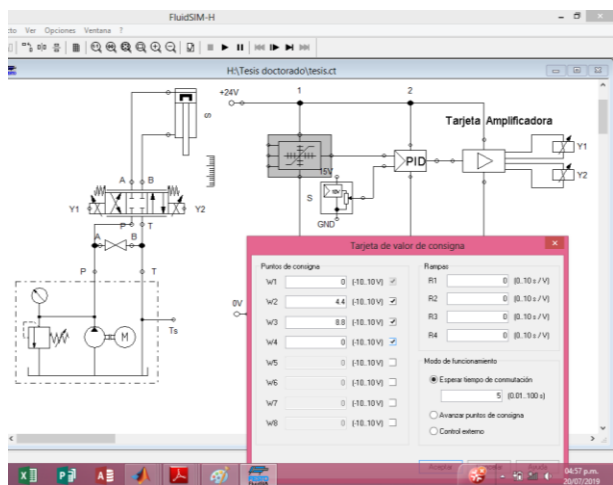


Fig. 1 Circuito electro-hidráulico para el posicionamiento lineal de un cilindro de doble efecto.

En la figura 1 se aprecia el símbolo de una bomba hidráulica de caudal constante en la parte inferior izquierda, a continuación, está la válvula proporcional eléctrica 4/3 vías. Al lado derecho de la figura se muestra el sistema de control, compuesto por la tarjeta de consigna, la tarjeta reguladora PID, el sensor de posición (potenciómetro lineal), y una tarjeta amplificadora.

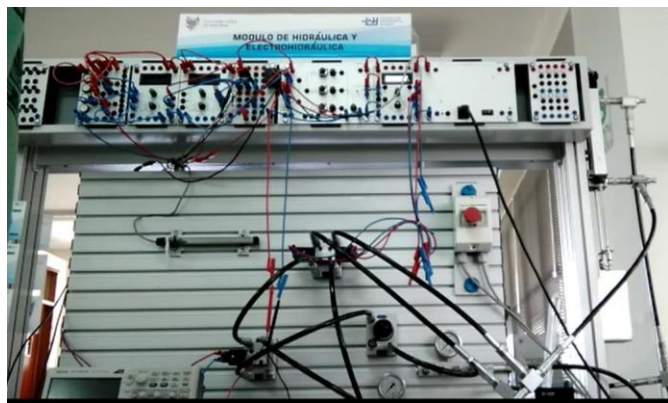


Fig. 2 Módulo de posicionamiento lineal de un cilindro hidráulico – FESTO.

En la figura 2 se aprecia el módulo de control de posición, al lado derecho de la figura se ve el cilindro hidráulico conectado a una carga de 9 kg, en la parte superior de la figura las tarjetas electrónicas y la fuente de alimentación de 24 VDC y finalmente en el centro de la figura la válvula proporcional de 4 vías y 3 posiciones, de centro cerrado. Básicamente el control de posición consiste en el movimiento del vástago del cilindro, de acuerdo a la cantidad de aceite que provee la válvula proporcional, la válvula se mueve de acuerdo al error entre la posición deseada y la posición actual del vástago del cilindro hidráulico. Esta comparación se da en términos de voltaje de 0 a 10 voltios (0 voltios indica la posición retraída del cilindro y 10 voltios es la posición extendida de 200 mm). La dificultad está en que en el sistema hay elementos no lineales como son el movimiento del fluido y la válvula proporcional, en la dinámica de sus correderas.

Todos los componentes utilizados - excepto el sensor de posición- son de la marca FESTO; es decir la bomba hidráulica, la válvula, la tarjeta PID, la tarjeta amplificadora, las mangueras, válvula de cierre. Esta marca es una de las más importantes en módulos educativos de aprendizaje de hidráulica en Latinoamérica. Se inició esta investigación debido a que en las guías de laboratorio de hidráulica en bucle cerrado de FESTO se recomendaba hacer la sintonización de Ziegler – Nichols para el control PID; pero al probar en el módulo los valores obtenidos se tenían respuestas oscilatorias, y la investigación se basa en el empleo de algoritmos metaheurísticos para sintonizar los parámetros del controlador PID en lazo cerrado.

Los parámetros K_p , K_i , K_d son valores que se fijan con potenciómetros en la tarjeta reguladora PID. Esta tarjeta es la parte más importante del módulo de aprendizaje de hidráulica en bucle cerrado del *Laboratorio de Oleohidráulica y Neumática de la UCSM (Universidad Católica de Santa María)*. Cada uno de estos parámetros puede variar de 0 hasta 1000 con un paso de 0.1; en consecuencia, las combinaciones entre los tres parámetros prácticamente son infinitas. Sintonizar estos 3 parámetros significa encontrar un juego de 3 valores K_p , K_i y K_d que permita al sistema alcanzar una referencia, satisfaciendo unas condiciones de diseño como es: un tiempo de establecimiento mínimo, un error en estado estacionario nulo y un pequeño sobreimpulso.

Existen muchos otros juegos de valores que hacen al sistema inestable. Se aplicaron para esta investigación los métodos de sintonización empíricos de Ziegler-Nichols y dos algoritmos metaheurísticos de optimización, para poder hacer una comparación.

D. Los Algoritmos Metaheurísticos ACO y PSO

Es común que este tipo de sintonización de los parámetros del controlador PID se plantee como un problema de optimización. Bajo esa perspectiva, la tarea del diseño de los parámetros de controladores PID, se convierte en encontrar el punto óptimo dentro de este conjunto de objetivos distintos. El

problema de sintonización se convierte entonces en un problema de optimización multiobjetivo [5].

Los métodos bio-inspirados se pueden agrupar en tres ramas principales: métodos evolutivos, de enjambre y de ecosistemas [5]. En el pasado reciente se han propuesto varios meta-algoritmos de inteligencia artificial como Algoritmo Genético (GA), Optimización de Colonia de Hormigas (ACO), Optimización de Enjambre de Partículas (PSO), etc. para estimar las ganancias optimizadas de sintonización de los controladores PID [6].

E. Algoritmo de Colonia de Hormigas (ACO)

Este es un algoritmo de inteligencia de enjambre desarrollado por Marco Dorigo en 1992 en su tesis de Ph.D. Este es una aproximación basada en la población, usada para resolver el problema de optimización combinatorial. El algoritmo ACO se originó del comportamiento de las hormigas y su habilidad para encontrar el camino más corto de su nido a la fuente de comida [7].

Las hormigas reales explotan una forma de comunicación indirecta mediante el medioambiente, ellas pueden seguir hasta una fuente de alimento porque, mientras caminan, ellas depositan feromona en el suelo, y ellas tienen una preferencia por caminos con mayor cantidad de feromonas.

Eventualmente, la totalidad de la colonia podría encontrar el más corto camino óptimo del nido al objetivo. En el algoritmo colonia de hormigas (ACO), el comportamiento de una hormiga real es simulado por un conjunto de hormigas artificiales. Algunos recursos computacionales se atribuyen a las colonias de hormigas artificiales para cooperar en la búsqueda de soluciones en el espacio de búsqueda del problema en cuestión. La medición de la calidad está basada en el rendimiento de una función de costo [8].

Descripción del diseño de control PID basado en ACO. En orden a resolver un problema de optimización, es muy útil describir el problema por una gráfica consistente de nodos conectados por aristas, ver figura 3 [9].

Durante el tour las hormigas deben visitar los nodos seleccionando un camino entre la fuente y el nodo de destino. El objetivo del algoritmo de colonia de hormigas, del problema de optimización de parámetros PID, consiste en minimizar la función costo entre todos los nodos, las hormigas son movidas por una regla de probabilidad para escoger su tour [10].

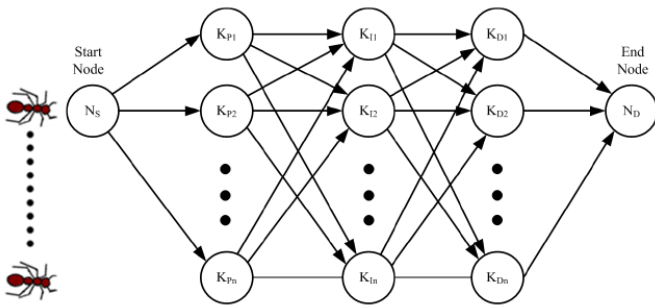


Fig.3 Representación gráfica del ACO para la sintonización PID. [9]

F. Algoritmo de Optimización de Enjambre de Partículas (PSO)

Este es un algoritmo metaheurístico basado en la naturaleza que adopta el comportamiento social de criaturas tales como cardúmenes de peces y bandadas de aves. Este fue desarrollado por Eberhart and Kennedy en 1995 y es una herramienta de optimización de rápido crecimiento para resolver los problemas de ingeniería y ciencia. El algoritmo PSO imita el comportamiento social animal, pero no requiere de un líder en el grupo para alcanzar el objetivo. Cuando la bandada de aves va a encontrar comida, ellas no requieren ningún líder; ellas van con uno de los miembros quien está en la posición más cercana a la comida [7].

El algoritmo PSO utiliza una población de individuos para descubrir la alta solución en un área de búsqueda entre soluciones vecinas. El individuo está definido por una partícula, que se desplaza estocásticamente bajo la guía de su posición más fina anterior y la mejor ubicación pasada del enjambre [11].

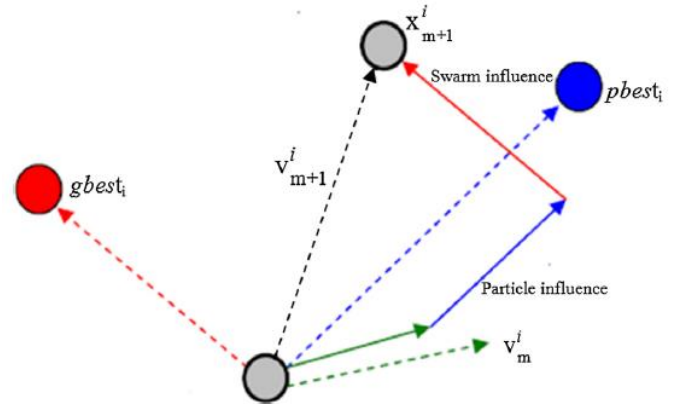


Fig. 4 Descripción de un punto de búsqueda por PSO. Vista del movimiento de una partícula en el algoritmo PSO [11].

Descripción del diseño de control PID basado en PSO. Asumiendo que el tamaño del enjambre es n y el área de búsqueda es m , a continuación la posición de la partícula i -ésima es dada por $x_i = (x_{i1}, x_{i2}, \dots, x_{id})$. Las mejores posiciones anteriores de cada i -ésima partícula son considerados como: $pbest_i = (pbest_{i1}, pbest_{i2}, \dots, pbest_{id})$ [11]. El índice de la mejor partícula entre el grupo es $gbest_d$. La velocidad de la i -ésima partícula es representada como: $v_i = (v_{i1}, v_{i2}, \dots, v_{id})$. La posición y velocidad modificada de cada partícula puede ser calculada usando el valor actual y la distancia de $pbest_{i,d}$ a $gbest_d$ (ver figura 4) como se expresa en las siguientes ecuaciones (1) y (2) [11]:

$$v_{i,m}^{t+1} = w \cdot v_{i,m}^t + C_1 * rand() * (pbest_{i,m} - x_{i,m}^t) + C_2 * rand() * (gbest_{i,m} - x_{i,m}^t) \quad (1)$$

$$X_{i,m}^{t+1} = X_{i,m}^t + V_{i,m}^{t+1} \quad i = 1, 2, \dots, n; m = 1, 2, \dots, d \quad (2)$$

G. Índice de Desempeño ITAE

En los algoritmos de optimización mencionados anteriormente, una parte fundamental es una función de costo o índices de desempeño. Un índice de desempeño es una medida cuantitativa que sirve para evaluar el desempeño de un sistema y se elige de forma que resalte las especificaciones más importantes del sistema. Dado esto, el mejor sistema es aquel que minimice o tenga el valor menor en dicho índice [12].

Los índices de desempeño se calculan con base a un tiempo finito elegido de forma arbitraria T , de forma que el sistema alcance su estado estacionario. Es conveniente elegir T como el tiempo de asentamiento (t_s), esto con propósito computacional [12].

Los índices de desempeño (todos son nombrados por sus siglas en inglés) se obtienen en función del error, definido por la diferencia entre la referencia del sistema y la salida del sistema.

Los índices más utilizados son:

IAE: integral del valor absoluto del error

IE: integral del error

ISE: integral del error cuadrático

ITAE: integral del tiempo multiplicado por el valor absoluto del error, que se define como:

$$ITAE = \int_0^{\infty} t|e(t)|dt \quad (3)$$

Al ponderar el valor absoluto del error con el tiempo, los errores grandes presentes al inicio pasan desapercibidos ya que se multiplican con tiempos muy pequeños, esto quiere decir que da poco peso a los errores grandes al inicio y da un mayor peso a los errores pequeños presentes a lo largo del tiempo. Genera sobrepaso y oscilaciones más pequeñas que el IAE y el ISE, pero es complicado de calcular analíticamente [12].

ITSE: integral del tiempo multiplicado por el error cuadrático

ISTAE: integral del tiempo al cuadrado multiplicado por el error absoluto.

ISTSE: integral del tiempo al cuadrado multiplicado por el error cuadrático.

II. MATERIALES Y MÉTODOS.

A. Identificación Experimental

Para poder hacer las simulaciones se requiere tener un modelamiento de la planta, que básicamente consiste en el ingreso de fluido a una de las cámaras del cilindro y la salida de fluido de la otra cámara. Así es como se logra el movimiento del cilindro hacia adelante o hacia atrás, logrando que varíe la posición final del vástago entre cualquier punto de su carrera que va de 0 a 200 mm (ver figura 5).

Modelar esta planta de forma analítica es complejo por las no linealidades de la válvula proporcional, por lo cual el método de trabajo que se empleó es usar un modelamiento computacional; es decir hacer una identificación experimental de acuerdo a la respuesta del sistema a una entrada escalón.

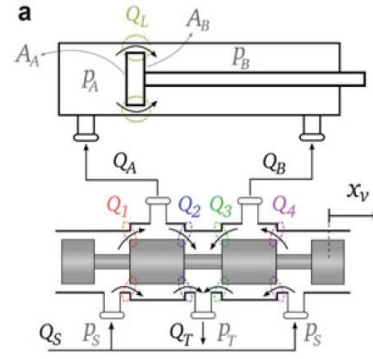


Fig. 5 Conexión cilindro-válvula con las principales variables del modelamiento. (a) Variables referidas a la hidráulica [13].

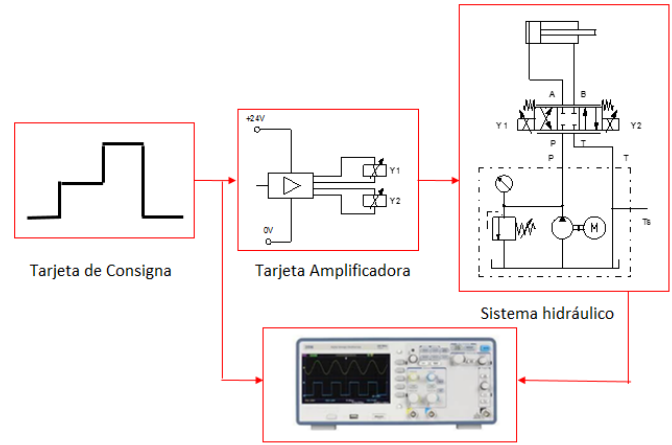


Fig. 6 Esquema de la adquisición de datos con osciloscopio.

La figura 6 muestra el esquema para la obtención de datos del sistema de posicionamiento en lazo abierto, se usó la tarjeta de adquisición de datos que está contenida en el osciloscopio Tektronix del laboratorio. Luego de este proceso los datos obtenidos se graficaron:

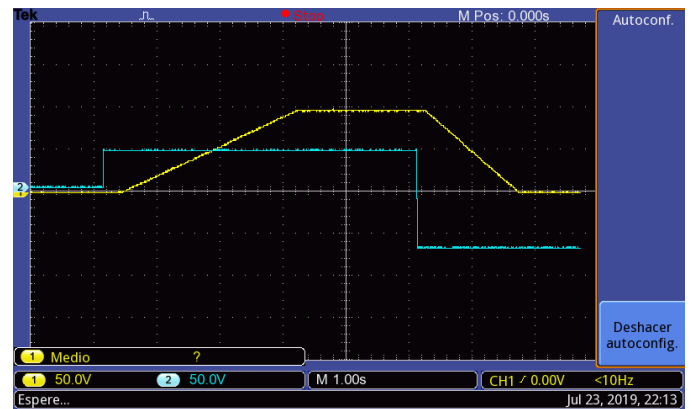


Fig. 7 Adquisición de datos en el osciloscopio, del sistema con carga para la identificación. La salida es de color amarillo.

En la figura 7 se observa un efecto de retardo o tiempo muerto de aproximadamente 0.2 segundos. Aquí se debe mencionar que al vástago del cilindro se le puede colocar una carga de 9 Kg. Entonces en esta investigación se hicieron pruebas del sistema con carga y sin carga, finalmente en este trabajo se aborda principalmente el sistema con carga (el cilindro se coloca en forma vertical y la carga ejerce una fuerza de tracción al cilindro).

Con los datos obtenidos usamos el toolbox *ident* de MATLAB, para lograr una identificación de la planta y poder hacer las simulaciones. En la figura 8, se muestra la interfaz del toolbox *ident* y las opciones que se tienen para escoger el método de regresión (ARX, ARMAX, BJ, OE, etc.). Luego del primer proceso de identificación se repitió el proceso para obtener un mejor resultado, ajustando el cero del sensor, ingresando el tiempo muerto como parte de la solución:

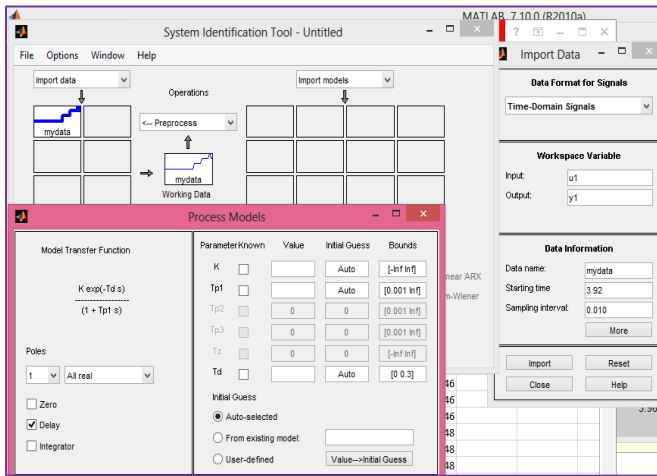


Fig. 8 Captura del método de estimación process models. Toolbox *ident* de MATLAB con los datos para la identificación.

Como se mencionó antes, este sistema tiene un tiempo muerto; por lo cual hay una no linealidad. Para poder aproximar a un sistema lineal se hizo una aproximación de la función exponencial en una serie infinita truncada en el segundo término obteniendo las siguientes funciones de transferencia:

TABLA I
RESULTADOS DE LA IDENTIFICACIÓN EXPERIMENTAL EN MATLAB

Tipo de estimación	Resultado	Best fits
Process models	$G_2(s) = \frac{1.932}{1 + 1.674s} e^{-0.6s}$	84.01%
Process models	$G_2(s)^* = \frac{1.93}{(1 + 1.6742s)(1 + 0.6s + 0.18s^2)}$	89.31%
Process models	$G_3(s) = \frac{2.8243}{(1 + 1.2326s)(1 + 1.0713s)} e^{-0.085455s}$	92.73%
Process models	$G_3(s)^* = \frac{2.8243}{(1 + 1.2326s)(1 + 1.0713s)(1 + 0.085s + 0.003s^2)}$	92.73%

G* es una aproximación lineal de G.

En la Tabla I, se muestran varios resultados pero el de mayor porcentaje de ajuste es $G_3(s)$, quiere decir que esa función de transferencia aproxima la planta y con esa función de transferencia se realizarán las simulaciones antes de probar los valores en el módulo real.

B. Implementación del Algoritmo PSO en MATLAB

El algoritmo que se codificó fue el algoritmo recuperado de [5], el diagrama de flujo se muestra en la figura 9, es un algoritmo fácil de entender y programar con las estructuras del MATLAB, lo complejo son las dimensiones de los vectores y matrices que se van creando para almacenar las mejores posiciones de cada partícula y del enjambre en cada tour.

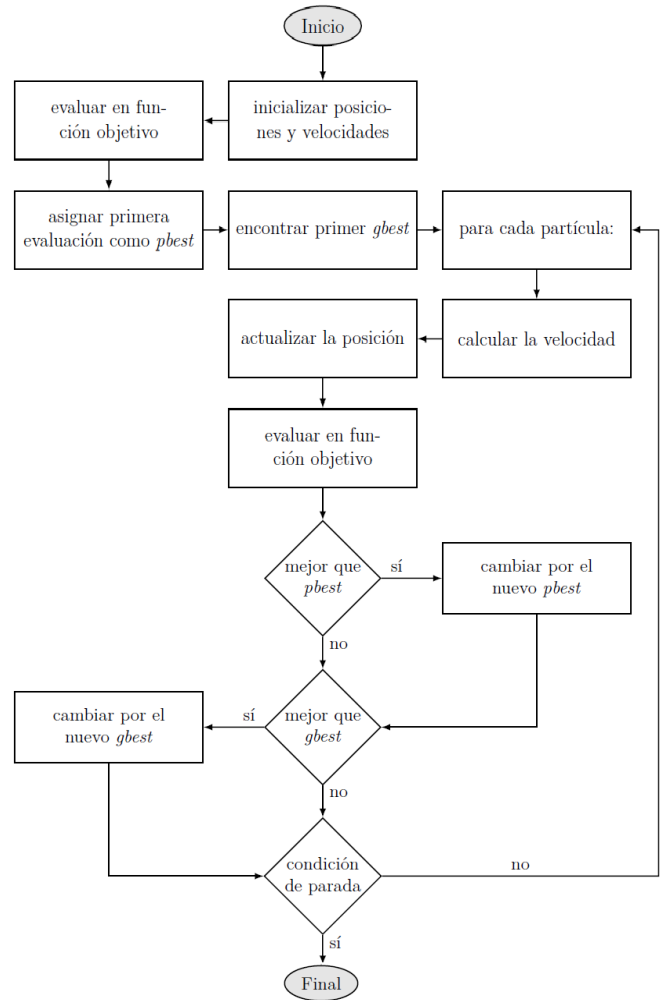


Fig. 9 Diagrama de flujo del algoritmo de PSO. Recuperado de [5].

En el código se deben ingresar los parámetros del algoritmo PSO, Los datos ingresados para las simulaciones se muestran en la Tabla II.

TABLA II
SELECCIÓN DE PARÁMETROS PARA EL ALGORITMO DE ENJAMBRE DE
PARTÍCULAS (PSO).

Parámetro	Valor
Tamaño de la población	20
Número de iteraciones	50
Número de nodos	1000
Dimensión	3
Velocidad de constante c1	2
Velocidad de constante c2	2
Tiempo de toma de datos para el ITAE	2.5seg
Periodo de muestreo del error	0.01

Al número de iteraciones también se le denomina tours.

Para ver el correcto funcionamiento del algoritmo se graficó la evolución del índice de desempeño ITAE.

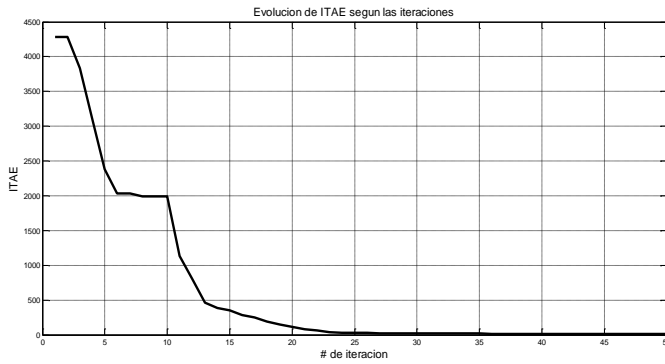


Fig. 10 Evolución del índice de desempeño ITAE con PSO con 50 iteraciones de 20 partículas.

En la figura 10 se observa que este algoritmo converge rápidamente, porque no requiere de las 50 iteraciones, sino de tan solo 25.

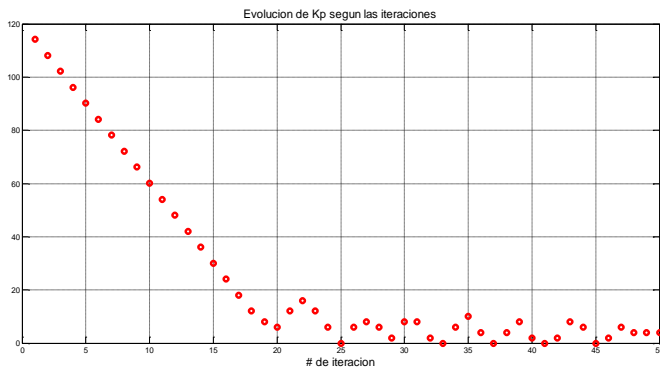


Fig. 11 Mejores soluciones de K_p con 50 iteraciones de 20 partículas.

De las figuras 11, 12 y 13 se observa que van convergiendo a un valor en la iteración 25. El valor K_i no termina de converger; pero la variación es pequeña y no afecta mucho al índice de desempeño. De esta manera se puede validar el funcionamiento del código implementado.

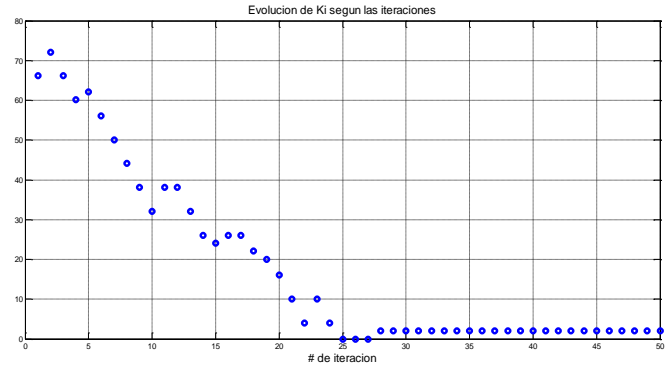


Fig. 12 Mejores soluciones de K_i con 50 iteraciones de 20 partículas.

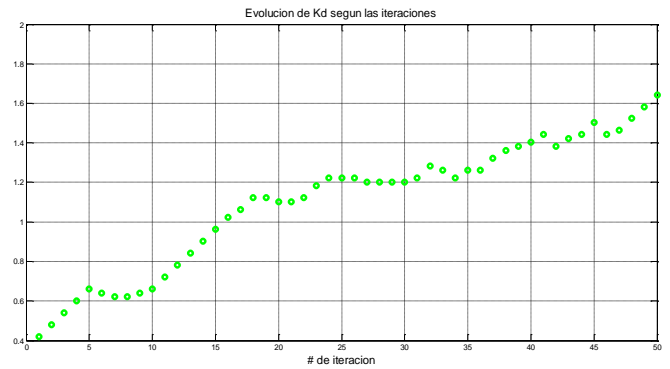


Fig. 13 Mejores soluciones de K_d con 50 iteraciones 20 partículas

C. Implementación del Algoritmo ACO en MATLAB

Para tener una comparación, también se probó con el algoritmo de colonia de hormigas; pero lo que se hizo para este trabajo es tomar un código disponible en [9], la variación que se hizo fue cambiar la función de costo de acuerdo a la función de transferencia identificada previamente:

```
function cval = ant_cost(x)
kp = x(1);
ki = x(2);
kd = x(3);
t=0:0.01:5;
u=ones(length(t),1);
G=tf(1.9324,conv([1.6742 1],[0.6^2/2 0.6 1]));%el
retardo se descompone en una sere infinita
Gc=tf([kd kp ki],[1 0]);
Gla=series(Gc,G);
Glc=feedback(Gla,1);
[Y,T,X]=lsim(Glc,u,t);
reference=u;
output=Y;
err=reference-output;
[n,~]=size(err);
Itae=0;
for i=1:n
    Itae=Itae+t(i)*abs(err(i));
end
cval=Itae;
end
```

Fig. 14 Captura de la función costo implementado en MATLAB.

Esta función de costo fue la misma que se empleó en el algoritmo PSO, se aprecia en el código que la función de transferencia $G(s)$ que representa la planta fue de 84.01% de ajuste (ver Tabla I), posteriormente se probó con otras funciones de transferencia de mayor porcentaje de ajuste. En el código se muestra cómo se va calculando el ITAE, inicialmente es cero, luego se calcula el error absoluto multiplicado por el tiempo y se va acumulando para cada iteración.

III. RESULTADOS Y DISCUSIÓN.

El primer resultado obtenido para K_p , K_i y K_d fue al aplicar el primer método de Ziegler – Nichols que consiste en encontrar una curva en forma de S y trazar una tangente al punto de inflexión a la respuesta del sistema en lazo abierto. El segundo método de Ziegler – Nichols que consiste en hacer oscilar el sistema hasta encontrar el periodo crítico y la ganancia crítica. El tercer resultado fue al aplicar el algoritmo PSO y el cuarto resultado fue al aplicar el algoritmo ACO.

TABLA III

PARÁMETROS DEL CONTROLADOR PID CALCULADOS PARA EL SISTEMA CON CARGA.

Método de sintonización	K_p	K_i	K_d	ITAE
Z-N 1er método	11.2405	17.7856	1.776	50.8602
Z-N 2do método	19.2	143.2836	0.6432	35 858.0
PSO1 50/20	4.008	8.016	1.984	8.1013
PSO2 80/20	5.01	2.004	1.984	7.8457
ACO1 100/20	16.016	21.021	5.15516	11.2989
ACO2 100/20	11.011	5.50551	3.71371	5.8954

En la tabla III se muestran diferentes resultados para los parámetros K_p , K_i y K_d , para seleccionar los mejores valores consideramos los valores más bajos del ITAE, esos valores garantizan un error muy pequeño en corto tiempo. 50/20 se refiere a 50 tours y 20 partículas

El siguiente paso es probar los valores en una simulación del sistema a una entrada escalón y ver el desempeño del controlador diseñado

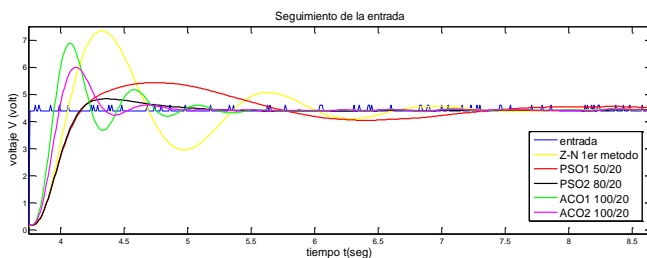


Fig. 15 Respuesta simulada del sistema con carga.

En la figura 15, no se muestra la respuesta del segundo método de Ziegler-Nichols porque resultaba inestable en la

simulación. También se observa que la peor respuesta es del primer método de Ziegler – Nichols, y la mejor respuesta en simulación es de del método de PSO con 80 tours y 20 partículas (curva de color negro), porque tiene el menor sobreimpulso y en poco tiempo se estabiliza.

El paso siguiente fue implementar los valores encontrados y probarlos en el módulo de control de posición del cilindro hidráulico, para lo cual se adquirió las señales con la tarjeta de adquisición de datos contenida en el osciloscopio, hacer coincidir el cero para poder graficar una sobre otra:

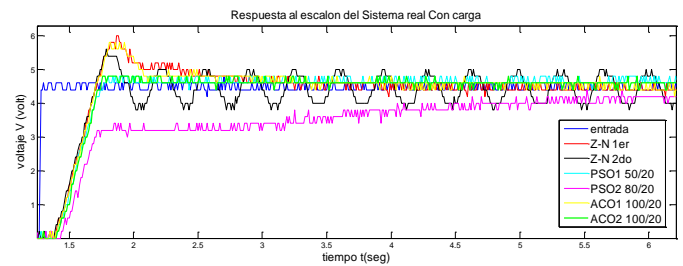


Fig. 16 Respuesta real del sistema con carga con control PID para los mismos parámetros

De la figura 16, se observa que las respuestas que tienen mayor sobreimpulso son los de Ziegler-Nichols y uno de los valores del método ACO, otra respuesta indeseable es del algoritmo PSO2, de 80/20, aunque no tiene sobreimpulso tiene un error muy grande y no llega a estabilizar. Quitando las respuestas malas a continuación se muestran los mejores valores.

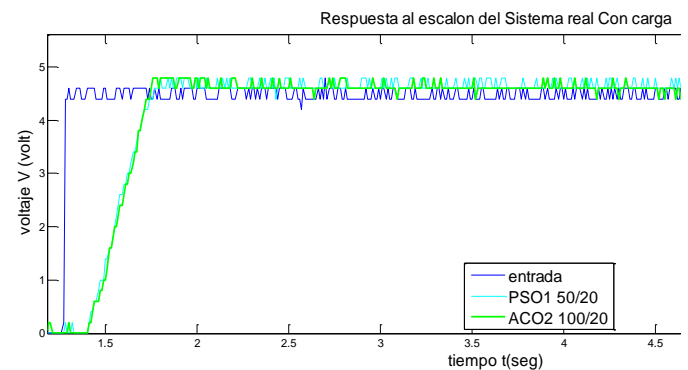


Fig. 17 Respuesta real del sistema con carga con control PID, mejores respuestas.

En la figura 17 se aprecia las mejores respuestas del sistema de control PID, al sistema real con carga de 9 Kg para el posicionamiento de un cilindro de 200 mm, se observa en la figura que no hay sobreimpulso, que no hay error en estado estacionario, y rápidamente estabiliza. Para medir el desempeño de un controlador, se debe medir los siguientes parámetros:

DESEMPEÑO DE LOS PARÁMETROS PID DEL SISTEMA REAL CON CARGA.

Método de sintonización	Mp	tr (seg)	tss (seg)	ess (volt)
Z-N (1er)	36.36%	0.45	2.73	0
Z-N (2do)	27.27%	0.45	inf	0.6
PSO1	0	0.49	0.56	0.2
PSO2	0	4.62	3.35	0.2
ACO1 100/20	27.27%	0.43	2.73	0
ACO2 100/20	0	0.47	0.48	0.1

Mp: máximo sobreimpulso, ess: error en estado estacionario, tr: tiempo de subida, tss: tiempo de establecimiento.

Un ensayo final que se hizo fue probar el sistema ante diferentes consignas de trabajo (setpoints) y los resultados fueron satisfactorios.

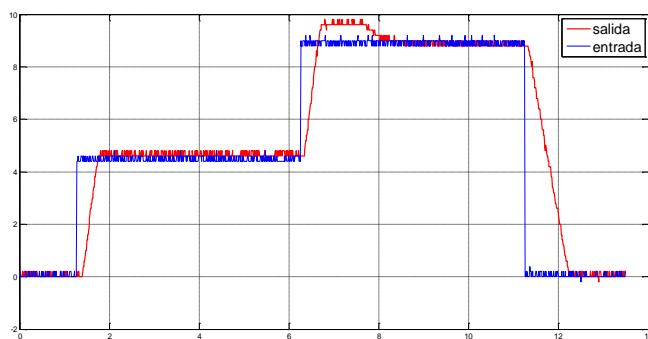


Fig. 17 Respuesta real del sistema con carga con control PID, PSO1 50/20.

IV. CONCLUSIONES

Los mejores resultados se obtuvieron con el algoritmo de colonia de hormigas, para 20 hormigas, con 1000 nodos y 100 tours. Además se redujo el espacio de búsqueda de K_p entre 0 y 500, también para K_i entre 0 y 500, para K_d entre 0 y 10. Para el sistema con carga la mejor solución fue $K_p=11.01$, $K_i=5.51$ y $K_d=3.71$, logrando alcanzar la posición deseada en tan solo 0.48 seg, sin sobreimpulso y con un error de 2 mm. Para el sistema sin carga la mejor solución fue $K_p=10.01$, $K_i=3.00$ y $K_d=4.81$, logrando alcanzar la posición deseada en tan solo 0.59 seg, sin sobreimpulso y sin error.

Se realizaron varias pruebas de identificación experimental, la mejor aproximación que se tuvo fue la de un sistema de orden 2 con retardo con un ajuste del 92.73%, esta función de transferencia fue válida para el sistema con carga y sin carga.

En esta investigación se trabajaron 4 métodos: Z-N primer y segundo método, PSO y ACO. Las mejores soluciones fueron obtenidas con el algoritmo de colonia de hormigas, este algoritmo para mejorar requiere de más de 100 tours y más nodos.

Se obtuvieron las respuestas a entradas escalones del sistema controlado con PID, con carga y sin carga. Estas respuestas se analizaron con la adquisición de datos y los

resultados se muestran en las gráficas. Las respuestas mejoraron cuando se hizo una nueva identificación del sistema para lograr un mayor porcentaje de ajuste.

El sistema sin controlador estabilizaba en 2.5 segundos y tenía un error de más del 100%, luego de aplicar el control en lazo cerrado PID con los parámetros calculados por ACO, se logró estabilizar en 0.6 segundos con un error menor al 5%. Los dos métodos de Z-N tenían sobreimpulso mayores al 25%.

Este trabajo además del aporte científico tiene la parte aplicativa en un módulo hidráulico tipo industrial; que también ha sido adquirido por muchas instituciones de educación superior en Latinoamérica.

REFERENCIAS

- [1] W. Acuña-Bravo, E. Canuto y M. Agostani, «Proportional electro-hydraulic valves: An Embedded Model Control,» *Control Engineering Practice*, pp. 22-35, 2017.
- [2] S. D. G. R. D. G. P. Liu, «Application Of Optimal-Tuning Pid Control To Industrial Hydraulic Systems,» *15th Triennial World Congress*, pp. 179-184, 2002.
- [3] G. Liu y S. Daley, «Optimal-tuning PID control for industrial systems,» *Control Engineering Practice*, pp. 1185-1194, 2001.
- [4] M. S. Sudarshan K. Valluru, «Performance investigations of APSO tuned linear and nonlinear PID controllers for a nonlinear dynamical system,» *Journal of Electrical Systems and Information Technology*, Vols. %1 de %2JESIT-189, pp. 1-11, 2017.
- [5] M. C. L. Macarena Céspedes Quesada, «Comparación de métodos de optimización bioinspirados para la sintonización de controladores PID,» Universidad de Costa Rica, Costa Rica, 2016.
- [6] M. SudarshanK.Valluru, «Performance investigations of APSO tuned linear and non linear PID controllers for a nonlinear dynamical system,» *Journal ofElectrical Systems and Information Technology*, pp. 1-11, 2017.
- [7] G. B. L. A. P. D. P. A. J. B. K. Patle, «A Review: On Path Planning Strategies for Navigation of Mobile Robot,» *Defence Technology*, pp. 1-40, 2019.
- [8] H. Boubertakh, «Knowledge-based Ant Colony Optimization Method to Design Fuzzy Proportional Integral Derivative Controllers,» *Journal of Computer and Systems Sciences International*, pp. 681-700, 2017.
- [9] M. Ünal, A. Ak, V. Topuz y H. Erdal, *Optimization of PID Controllers Using Ant Colony and Genetic Algorithms*, New York: Springer-Verlag Berlin Heidelberg, 2013.
- [10] P. V. A. S. L. L. Aruchamy Sakthivel, «Experimental investigations on Ant Colony Optimized PI control algorithm for Shunt Active Power Filter to improve Power Quality,» *Control Engineering Practice*, pp. 153-169, 2015.
- [11] S. B. A. C. S. B. Ghoulemallah Boukhalfa, «Direct torque control of dual star induction motor using a fuzzy-PSO hybrid approach,» *Applied Computing and Informatics*, pp. 1-8, 2018.
- [12] R. Martínez Trujillo, «Algoritmos bioinspirados para la sintonización de sistemas de control con retardo,» Centro de Investigación Científica y de Educación Superior de Ensenada, Baja California, 2018.
- [13] V. B. a. K. J. Jörn Malzahn, *Actuator Modeling and Simulation*, Genoa: Springer Science+Business Media, 2017.
- [14] Katsuhiko Ogata, *Ingeniería de Control Moderna*, Madrid: PEARSON, 2010, p. 11.
- [15] P. M. Marusak y S. Kuntanapreedda, «Constrained model predictive force control of an electrohydraulic actuator,» *Control EngineeringPractice*19, pp. 62-73, 2011.