



Diseño y aplicación de los algoritmos genéticos para la sintonización de un controlador PID

Design and application of genetic algorithms for tuning a PID controller

Projeto e aplicação de algoritmos genéticos para sintonia de um controlador PID

Ismael Elías Erazo-Velasco ^I

erazoismael@yahoo.com

<https://orcid.org/0000-0002-7647-4611>

Correspondencia: erazoismael@yahoo.com

Ciencias técnicas y aplicadas

Artículo de Investigación

***Recibido:** 6 de enero de 2022 ***Aceptado:** 10 de enero de 2022 * **Publicado:** 19 de enero de 2022

- I. Investigador Independiente, Experto en Sistema de Control y Automatización, Master en Energía Eléctrica y Electrotecnia en ASTRAKHAN STATE TECHNICAL UNIVERSITY, Ingeniero Eléctrico de la Universidad Técnica Luis Vargas Torres de Esmeraldas, Ecuador.

Resumen

En este trabajo se presenta una manera alternativa y moderna para calibrar, sintonizar y diseñar óptimamente un controlador PID conociendo la función de transferencia de la planta a controlar. El método que vamos a exponer se trata de una técnica de inteligencia artificial basada en los métodos evolutivos de búsqueda global de máximos y mínimos, la cual consiste en un algoritmo genético que diseñaremos con la ayuda de la herramienta Matlab con el interfaz de OPTIMIZATION TOOL como función principal para encontrar la constante de sintonización K_p , K_i y K_d del controlador PID.

Palabra clave: Controlador PID; sintonización; algoritmos genéticos.

Abstract

This paper presents an alternative and modern way to optimally calibrate, tune and design a PID controller knowing the transfer function of the plant to be controlled. The method that we are going to expose is an artificial intelligence technique based on the evolutionary methods of global search for maximums and minimums, which consists of a genetic algorithm that we will design with the help of the Matlab tool with the OPTIMIZATION TOOL interface as main function to find the tuning constant K_p , K_i and K_d of the PID controller.

Keywords: PID controller; tuning; genetic algorithms.

Resumo

Este trabalho apresenta uma forma alternativa e moderna de calibrar, ajustar e projetar de forma otimizada um controlador PID conhecendo a função de transferência da planta a ser controlada. O método que vamos expor é uma técnica de inteligência artificial baseada nos métodos evolucionários de busca global por máximos e mínimos, que consiste em um algoritmo genético que projetaremos com o auxílio da ferramenta Matlab tendo como principal interface a FERRAMENTA DE OTIMIZAÇÃO função para encontrar a constante de sintonia K_p , K_i e K_d do controlador PID.

Palavras-chave: controlador PID; sintonia; algoritmos genéticos.

Desarrollo

Controlador PID

Es un mecanismo de control que a través de un circuito de retroalimentación ayuda a regular la velocidad la temperatura la presión y el caudal entre otras variables de un proceso en su conjunto.

El controlador PID calcula la diferencia entre nuestra variable real y la variable deseada

El controlador PID está compuesto por 3 elementos que son acción proporcional, integral y derivada, debida estos 3 elementos tiene el nombre controlador PID.[1]

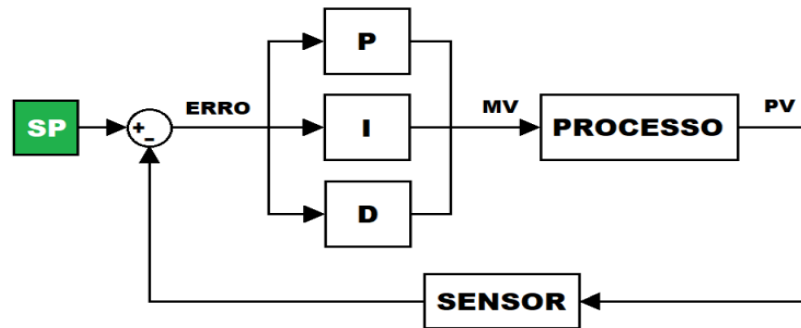


Figura 1. Circuito de retroalimentación de sistema de control.

Acción de control proporcional

Como su nombre indica esta acción de control es proporcional a la señal de error $e(t)$. Internamente la acción proporcional multiplica la señal de error por una constante K_p .

Esta verificación intenta minimizar la falla del sistema. Cuando la falla es grande la acción de control es importante y tiende a minimizar el error.

Aumentar el efecto proporcional a K_p tiene los siguientes efectos:

1. Aumenta la velocidad de respuesta del sistema.
2. Reducción de error del sistema de estado estacionario.
3. Aumenta la inestabilidad del sistema.

Acción de control integral

Esta acción de control como sugiere el nombre integra la señal de error $e(t)$, la integral se puede considerar como la suma o el producto de la señal de error.

Con el tiempo se agregan pequeños errores para hacer que la acción integral sea cada vez más grande. Con esto es posible reducir el error del sistema de estado estacionario.

La desventaja de usar la acción integral es que agrega algo de inercia al sistema y por lo tanto lo hace más inestable.

Aumentar la acción integral K_i tiene los siguientes efectos:

1. Disminuye el error del sistema en régimen permanente.
2. Aumenta la inestabilidad del sistema.
3. Aumenta un poco la velocidad del sistema.

Acción de control Derivativa

Esta acción de control es proporcional a la derivada de la señal de error $e(t)$. La derivada del error es otra forma de llamar a la "velocidad" del error la acción de control proporcional siempre intenta aumentar la posición El problema proviene de tener en cuenta la inercia.

Cuando el sistema se mueve a alta velocidad hacia el punto de referencia, el sistema se desplazará debido a su inercia. Esto crea sobre impulsos y oscilaciones alrededor de la referencia. Para evitar este problema, el controlador debe ser consciente de la rapidez con la que el sistema se acerca al punto de ajuste para poder frenar antes de alcanzar el punto de ajuste deseado y evitar que el sistema se sobrepase.

Aumentar la constante de control derivativa K_d tiene los siguientes efectos:

1. Aumenta la estabilidad del sistema controlado.
2. Disminuye un poco la velocidad del sistema.
3. El error en régimen permanente permanecerá igual.[1]

Materiales y métodos

Ecuación del controlador

La ecuación del control PID es la siguiente

$$U(t) = K_p e(t) + K_d \frac{de(t)}{dt} + K_i \int_0^T e(t)dt \quad (1)$$

Donde:

$U(t)$ = señal de control

$e(t)$ = señal de error

K_p, K_i, K_d = parámetros del controlador PID

Si aplicamos la transformada de Laplace a ambos miembros obtenemos los siguientes

$$U(s) = (K_p + s K_d + K_i \frac{1}{s}) e(s) \quad (2)$$

$$\frac{u(s)}{e(s)} = K_p + s K_d + K_i/s \quad (3)$$

Saturación y límites del controlador PID

En los sistemas reales existen limitaciones que reducen la capacidad del controlador para lograr la respuesta deseada. Por mucho que se aumente la acción proporcional llegará un momento en que el actuador se saturará y ya no podrá autoabastecerse.

Por ejemplo, en un sistema de control de temperatura el elemento calefactor puede proporcionar hasta 2000 vatios de potencia.

Si el controlador intenta suministrar más energía para lograr una tasa de calentamiento más rápida esto no será posible y el sistema no funcionará más rápido. Aunque la acción de control proporcional aumenta el límite del actuador de 2000 vatios limitará la velocidad máxima de calentamiento.

Por lo tanto, hay que tener en cuenta que la velocidad de respuesta de los sistemas reales tiene ciertos límites que el control no podrá superar.[2]

Algoritmos genéticos

Es una secuencia organizada de pasos que describe el proceso que se debe seguir para resolver un problema en particular. El algoritmo genético (o GA para abreviar) es una técnica de programación inspirada en la recreación de organismos vivos y la imitación de la evolución biológica como estrategia para resolver problemas de optimización. En general los algoritmos genéticos (AG) forman parte de lo que se conoce como inteligencia artificial; es decir resolución de problemas mediante el uso de programas informáticos que imitan el funcionamiento de la inteligencia natural. Los algoritmos genéticos (AG) funcionan entre el conjunto de soluciones a un problema llamado fenotipo y el conjunto de individuos en una población natural codificando la información de cada solución en una secuencia generalmente en forma binaria llamada cromosoma. Los símbolos que componen la secuencia se denominan genes. Cuando la representación de los cromosomas se realiza mediante cadenas de dígitos binarios hablamos de un genotipo. Los cromosomas evolucionan en muchas repeticiones llamadas generaciones. En cada generación los cromosomas se evalúan utilizando varias medidas físicas.

Las generaciones posteriores (nuevas cromosomas) se producen mediante la aplicación repetida de operadores genéticos a saber operadores de selección cruce mutación y sustitución.[3]

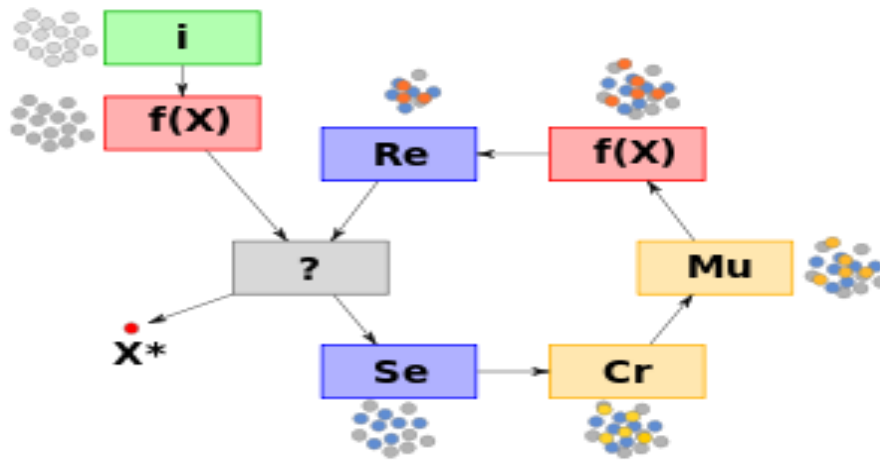


Figura 2. Programación de un algoritmo genético.

Partes de a un algoritmo genéticos

Alelo - Son los distintos valores con los cuales se puede representar un gen.

Gen. - Es el valor de un alelo dentro de un arreglo.

Cromosoma - Es una colección de genes en forma de arreglo.

Posición. - Es el lugar que ocupa un gen dentro del cromosoma.

Índice - Es la posición que tiene el individuo dentro de la población

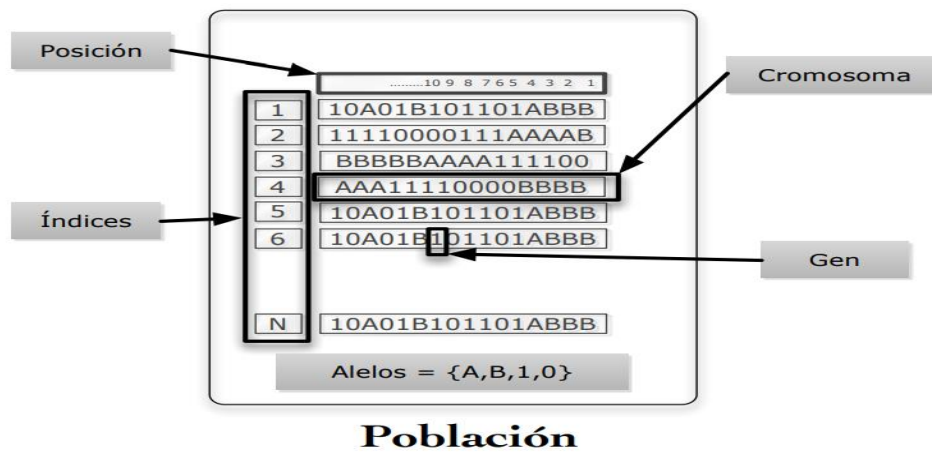


Figura 3. Partes algoritmos genéticos.

Parámetros de los algoritmos genéticos.

Para el estudio de los algoritmos genéticos hay que tener en cuenta una serie de parámetros:

1. Función coste o aptitud.
2. Tamaño de la Población.
3. Selección.
4. Operador de cruce.
5. Operador de mutación.

Función coste o aptitud

Es la función objetivo que deseamos obtener ya sea el mínimo o máximo global, esta función también nos permite evaluar todas las posibles soluciones en un espacio de búsqueda determinado, la función coste es la parte principal de los algoritmos de programación genética.

Población Inicial

Estos parámetros nos indican la cantidad de individuos los cuales están distribuidos en todo el espacio de búsqueda de solución y van estar codificados en notación binaria para luego reproducirse, cruzarse, mutar y generar de nuevo otra población la cual será evaluada por la función coste para ver si se ha encontrado el valor óptimo al problema [3]

Tamaño de la Población

Este parámetro nos indica el número de cromosomas que tenemos en nuestra población para una generación determinada. En caso de que esta medida sea insuficiente, el algoritmo genético tiene pocas posibilidades de realizar reproducciones con lo que se realizaría una búsqueda de soluciones escasa y poco óptima. Por otro lado, si la población es excesiva, el algoritmo genético será excesivamente lento.

De hecho, estudios revelan que hay un límite a partir del cual es ineficiente elevar el tamaño de la población puesto que no se consigue una mayor velocidad en la resolución del problema

Ejemplo: Una población de N individuos, dos variables y una longitud de 6 bits

n	var 1	var 2	var 1	var 2
1	101010	110011	42	51
2	111111	000001	63	1
3	011011	101110	27	46
N	101111	100111	47	39

Selección

Como ya hemos visto anteriormente es necesario hacer una selección con los individuos más capacitados para que éstos sean los que se reproduzcan con más probabilidad de acuerdo con la teoría de Darwin en la cual los más capacitados son los que deben sobrevivir y crear una nueva descendencia más facultada. Por lo tanto, una vez evaluado cada cromosoma y obtenida su puntuación, se tiene que crear la nueva población teniendo en cuenta que los buenos rasgos de los mejores se transmitan a ésta. Esta selección se puede realizar de varias formas como se verá a continuación

Selección por Rueda de Ruleta

Se crea para esta selección una ruleta con los cromosomas presentes en una generación. Cada cromosoma tendrá una parte de esa ruleta mayor o menor en función a la puntuación que tenga cada uno. Se hace girar la ruleta y se selecciona el cromosoma en el que se para la ruleta. Obviamente el cromosoma con mayor puntuación saldrá con mayor probabilidad.

En caso de que las probabilidades difieran mucho, este método de selección dará problemas puesto que, si un cromosoma tiene un 90% de posibilidades de ser seleccionado, el resto apenas saldrá lo que reduciría la diversidad genética

Selección por Torneo

Se escogen de forma aleatoria un número de individuos de la población, y el que tiene puntuación mayor se reproduce, sustituyendo su descendencia al que tiene menor puntuación.

Selección Escalada

Al incrementarse la aptitud media de la población, la fuerza de la presión selectiva también aumenta y la función de aptitud se hace más discriminadora. Este método puede ser útil para seleccionar más tarde, cuando todos los individuos tengan una aptitud relativamente alta y sólo les distinguen pequeñas diferencias en la aptitud.

Selección Jerárquica

En esta selección, los individuos atraviesan múltiples rondas de selección en cada generación. Las evaluaciones de los primeros niveles son más rápidas y menos discriminatorias, mientras que los que sobreviven hasta niveles más altos son evaluados más rigurosamente. La ventaja de este método es que reduce el tiempo total de cálculo al utilizar una evaluación más rápida y menos selectiva para eliminar a la mayoría de los individuos que se muestran poco o nada prometedores, y sometiendo a una evaluación de aptitud más rigurosa y computacionalmente más costosa sólo a los que sobreviven a esta prueba inicial.

Otras Selecciones

Existen otras técnicas de selección que simplemente se comentarán a continuación.

Una de ellas es la selección por prueba de aptitud en las que los cromosomas con más aptitud tienen más posibilidad de ser seleccionados, pero no la certeza.

La selección generacional en la que ningún miembro de la población anterior se encuentra en la nueva.

Operador de cruce o reproducción

Los dos cromosomas padres se cortan por un punto. Se copia la información genética de uno de los padres desde el inicio hasta el punto de cruce y el resto se copia del otro progenitor. Es una de las formas clásicas de crossover.

Veamos algunos ejemplos gráficos:

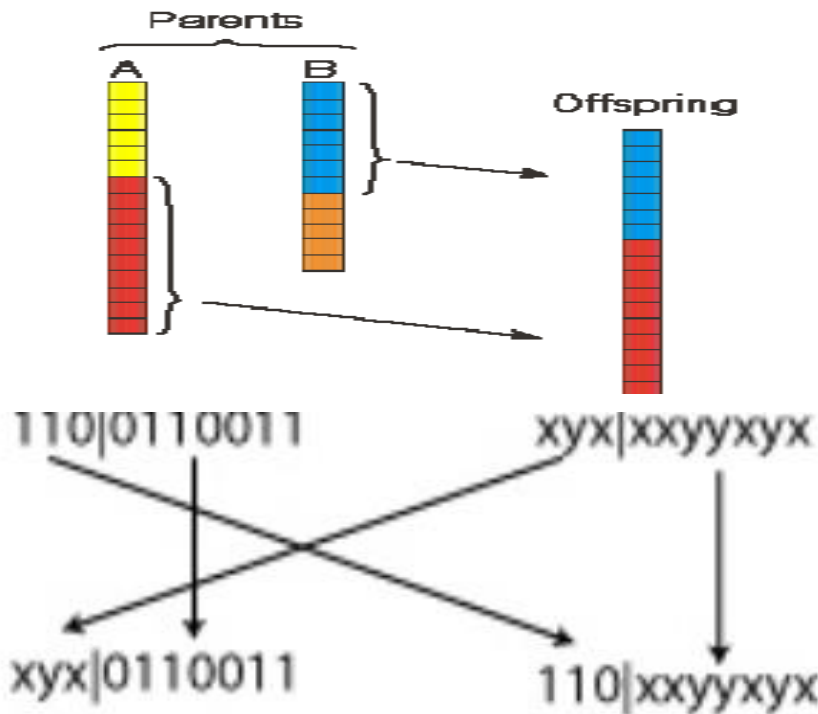


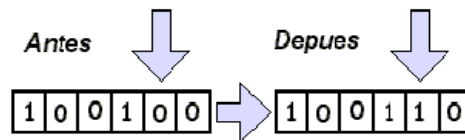
Figura 4. Operador de cruce.

Mutación

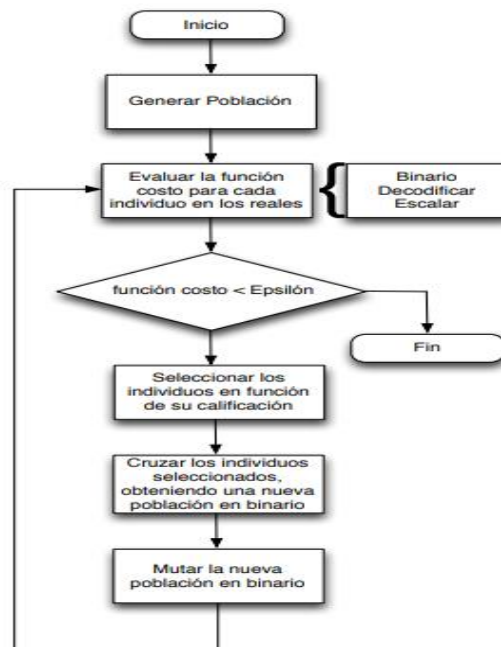
Tras el cruce, tiene lugar la mutación. Si nos referimos en términos de evolución, la mutación se manifiesta de forma extraordinaria, nada común. Las mutaciones suelen en promedio ser beneficiosas pues contribuyen a la diversidad genética de la especie. Además, previenen a las soluciones de la población de verse limitadas por un óptimo local. Por lo tanto, la mutación consiste en modificar ciertos genes de forma aleatoria atendiendo a la probabilidad de mutación establecida con anterioridad.

La mutación depende de la codificación y de la reproducción. Si se abusa de la mutación podemos caer en el uso del algoritmo genético como una simple búsqueda aleatoria. Por lo tanto, antes de aumentar las mutaciones, conviene estudiar otras soluciones que aporten diversidad a la población como podría ser el aumento del tamaño de la población o garantizar la aleatoriedad de la población inicial.

Para el caso de una codificación binaria, la mutación consiste simplemente en la inversión del gen mutado que corresponderá con un bit.



Secuencia de ejecución



Sintonización de un controlador PID

Sintonizar un sistema de control realimentado significa regular parámetros en el controlador para lograr implementar un control robusto en el proceso. Y por tanto alguien que intente o busque sintonizar un controlador PID debe entender la naturaleza dinámica del proceso que está siendo controlado.

Existen muchos métodos para poder sintonizar un controlador como pueden ser por cancelamientos de polos, Ziegler-Nichols entre otros todos estos métodos tienen una gran desventajas la cual es que más allá que encuentran la estabilidad del sistema no optimizan el funcionamiento del sistema y estos debidos a que estos métodos son basados de manera experimental en proceso de prueba error y acierto y en función de estos se realizan tablas para clasificar la función transferencia de la planta ya sea de primer, segundo o tercer orden con estas se extraen unas fórmulas para calcular las constantes K_p , K_i , K_d para calibrar el controlador.

Algoritmos genéticos para la sintonización de controladores PID

Para la implementación de los algoritmos genéticos en calibración de las constantes K_p , K_i , K_d vamos a realizar el siguiente problema. La planta a controlar está compuesta de una tolva que vierte material sobre una cinta transportadora. La cantidad de material se regula mediante un dosificador a tornillo proporcional. El peso es medido sobre la cinta con una celda de carga ubicada a una distancia determinada de la tolva. La velocidad de la cinta es, aproximadamente, constante. Por ello, existe un retardo, supuesto constante, en la medición de variaciones del peso debido a cambios en el dosificador de la tolva. Considerando los distintos elementos del conjunto y haciendo algunas aproximaciones se concluye que un modelo de la planta de cuarto orden más el retardo en la salida medida, representa bastante bien al sistema. El retardo de tiempo calculado para una cierta velocidad de la cinta es 14.6 m/seg.

En la figura número 6 se ve un bosquejo de proceso a controlar y la función transferencia del sistema y lo que buscamos es hallar las constantes K_p , K_i , K_d necesarias y óptimas para controlar y mejorar la eficiencia del proceso.

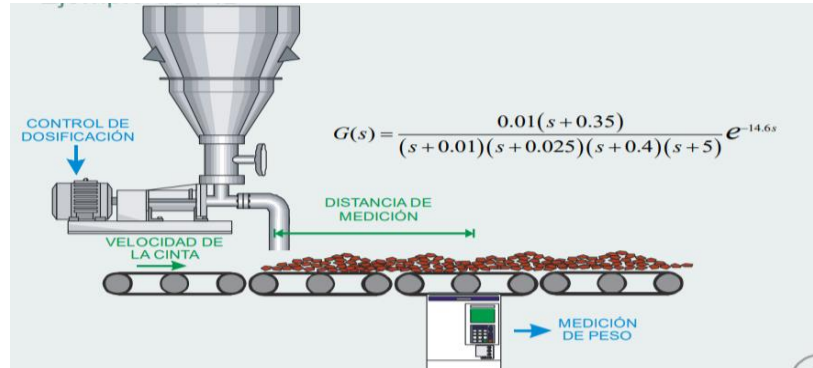


Figura 6. Planta.

Solución de del problema usando MATLAB y el solver Optimization tool

Antes de usar el solver Optimization tool primero debemos crear una función en MATLAB que represente diagrama de bloques final del proceso industrial a controlar, usaremos la fórmula del error absoluto la cual será la función coste o de aptitud para poder evaluar la mejor solución obtenidas por el solver

$$M = \sum_k t |e_k(t)| \Delta t_k$$

Donde

M -

t – Intervalo de tiempo máximo de muestreo para la optimización

Δt_k – Incremento discretizado de tiempo o partición del intervalo de muestreo

Implantación del código en MATLAB

```

1  function M = PID(x)
2  -   s= tf('s');
3  -   Planta = (0.01*(s+0.35)*exp(-14.6*s))/((s+0.01)*(s+0.025)*(s+0.4)*(s+5));
4  -   Kp=x(1)
5  -   Ki=x(2)
6  -   Kd=x(3)
7  -   Controlador= Kp + Ki/s+ Kd*s;
8  -   %step(feedback(Controlador*Planta,1))
9  -   dt= 0.01;
10 -   t=0:dt:100;
11 -   e= 1 - step(feedback(Controlador*Planta,1),t);
12 -   M= sum(t'.*abs(e)*dt);
    
```

Figura 7. Implementación de código en MATLAB.

Una vez creada la función de coste y guardada como PID, vamos a ir al interfaz Optimization tool para usar el solver de algoritmos genéticos el intervalo de búsqueda para hallar los valores de las constantes K_p , K_i y K_d límite inferior [0 0 0] y el límite superior [200 200 200]

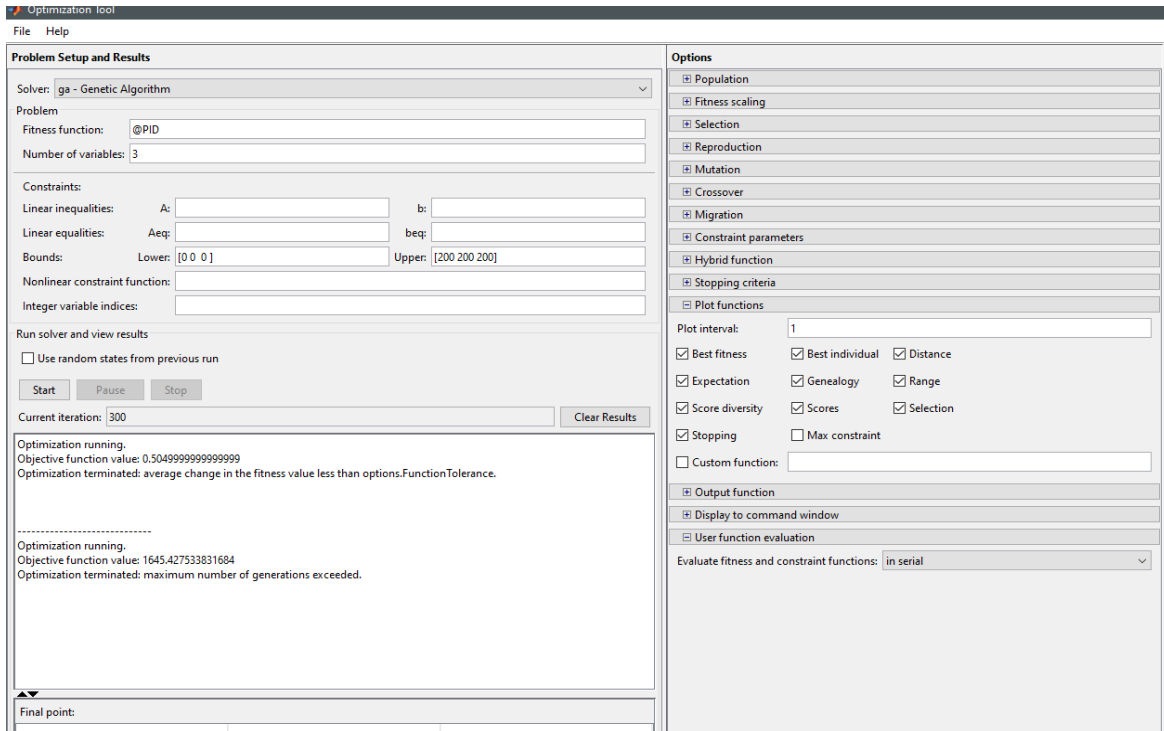


Figura 8. Introducción de las variables a Optimization tool.

Una vez ya ingresados todos los datos y elegido el solver ga – Genetic Algorithms obtuvimos los siguientes resultados

```

Command Window

Kp =

    1.1245

Ki =

    0.1534

Kd =

    65.8998

fx >>
    
```

Resultados

Gráficas obtenidas de los resultados de la optimización

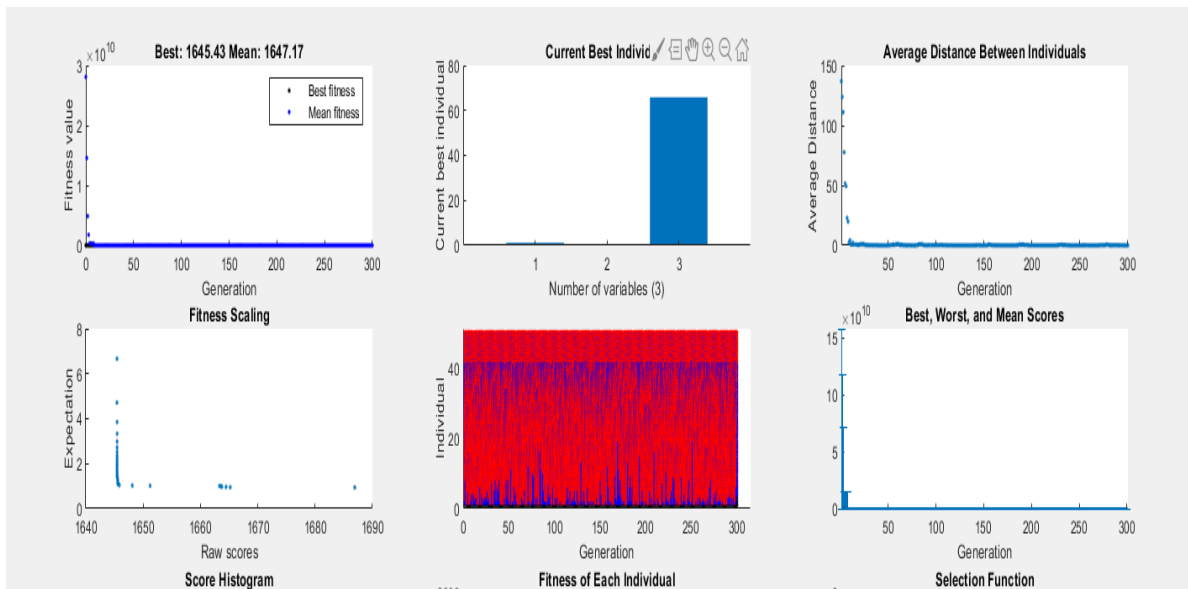


Figura 9. Resultados obtenidos.

En la figura 9 podemos 6 graficas la cuales nos indican los siguientes resultados

1. La grafica #1 muestra la mejor solución según las diferentes 300 generaciones que han pasados desde la población inicial
2. Grafica #2 el número de variables del problema las cuales son 3 K_p , K_i , K_d .
3. Representa el número de generaciones que han pasado desde la población inicial hasta encontrar la mejor solución.
4. Indica los diferentes errores que se van corriendo hasta encontrar la solución óptima.
5. Representa el número total de individuos que se fa creado en todas las generaciones.
6. Indica cómo se distribuye los diferentes individuos a lo largo de todas generaciones.

Esquema básico de control

En la figura # 10 vemos el circuito básico de control el cual consta de la función transferencia de la planta, el controlador PID con las constante ya calculadas con el solver Optimización tool, una función escalón la cual toma el valor de 1 y scope para visualizar la gráfica.

Datos:

$$K_p = 1.1245$$

$$K_i = 0.1534$$

$K_d = 65.8998$

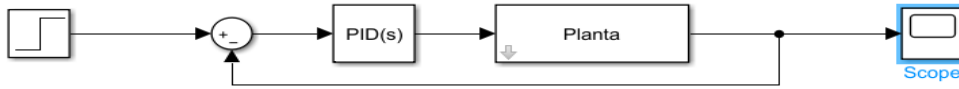


Figura 10. Esquema de control.

Respuesta de sistema

En la gráfica 11 se muestra la curva de control de la planta la cual tiene un retardo de 14.6 segundos luego de esto el sistema se estabiliza a los 400 segundos.

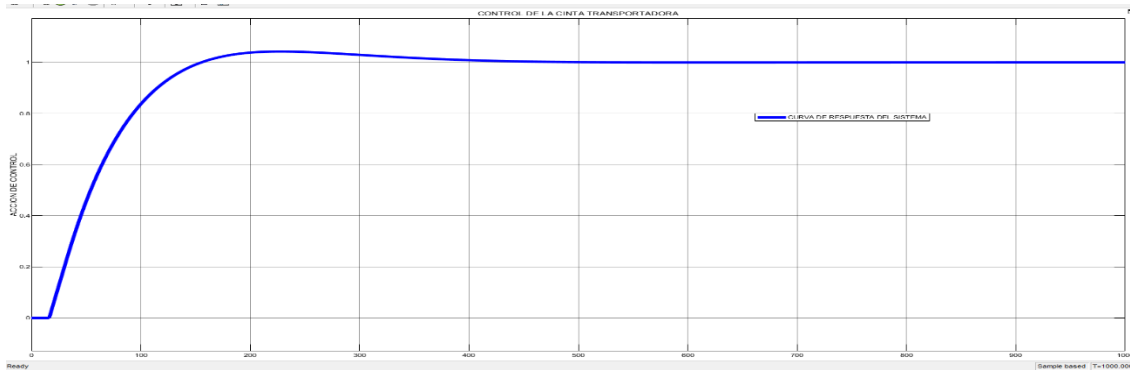


Figura 11. Curva de control.

Conclusiones

Como conclusión final podemos decir que la técnica de búsqueda de optimización de los parámetros K_p , K_i y K_d usando algoritmos genéticos es muy eficiente ya que tiene que la gran ventaja que va a calcular la mejor solución posible dentro de un intervalo búsqueda específico, pero es costo computacional de la programación es grande.

Referencias

1. C. Autom, M. Sintonizar, C. Pid, C. Usb, R. Intuilink, and D. Capture, "Tema: Sintonización de controladores PID Objetivos específicos Materiales y equipo," pp. 1–16.
2. A. E. Naconha, "No 主観的健康感を中心とした在宅高齢者における 健康関連指標に関する共分散構造分析Title," vol. 4, no. 1, p. 6, 2021.

3. L. A. Gen, "ALGORITMOS GENÉTICOS 1 Introducción," pp. 1–34, 1859.
4. Quiñónez, B. F. C., de Souza, R. C. U., de Souza, A. G. U., Peralta, A. J. M., Sandoval, J. E. E., & Bautista, J. D. M. (2021). Estudio de saturación de transformadores de corriente: análisis y simulación. *Brazilian Applied Science Review*, 5(2).
5. Báez-Zamora, J. J., Chere-Quiñónez, B. F., Martínez-Peralta, A. J., Simisterra-Quiñónez, K. Y., & Portocarrero-Rivas, F. F. (2018). Optimización de potencia en transformadores de distribución Power optimization in distribution transformers Otimização de energia em transformadores de distribuição.
6. Escobar-Sandoval, J. E., Chere-Quiñónez, B. F., & Ângulo-Guerrero, R. J. (2020). Uso del programa MATLAB/SIMULINK en la descripción de motores eléctricos de corriente alterna. *Análisis documental. Dominio de las Ciencias*, 6(5), 348-360.

©2022 por los autores. Este artículo es de acceso abierto y distribuido según los términos y condiciones de la licencia

Creative Commons Atribución-NoComercial-CompartirIgual 4.0 Internacional (CC BY-NC-SA 4.0)

(<https://creativecommons.org/licenses/by-nc-sa/4.0/>).