Comparación de Algoritmos genéticos Aplicados a Sistemas PID y Ecuaciones 3x3

José Joaquín Restrepo Palacios, Manuel Alejandro Chalarca Botero

A genetic algorithm was programmed using the Matlab language following the instructions given by the educator, it has been evaluated using a 3x3 equation to which an attempt has been made to extract its solutions through this genetic algorithm, requiring some precision and then comparing it with the solutions given by a library specialized in genetic algorithms commonly used in Python. The process was repeated for a plant found in the databases and an attempt was made to find the most appropriate solutions for its PID system and then compare it again with the solutions provided by the Matlab library.

Keywords: Genetic Algoritm, Matlab, PID, Python

# INTRODUCTION

This report is presented as evidence of the first installment of the Intelligent Systems subject of the Metropolitan Technological Institute, on March 15, 2023.

This comparison was made by José Joaquín Restrepo Palacios and Manuel Alejandro Chalarca Botero.

2 comparaciones

2.1 Algoritmo genético y Sistema de ecuaciones lineales

Usando el diagrama de flujo otorgado por el educador se desarrolló un algoritmo genético que cumpliera con las especificaciones requeridas

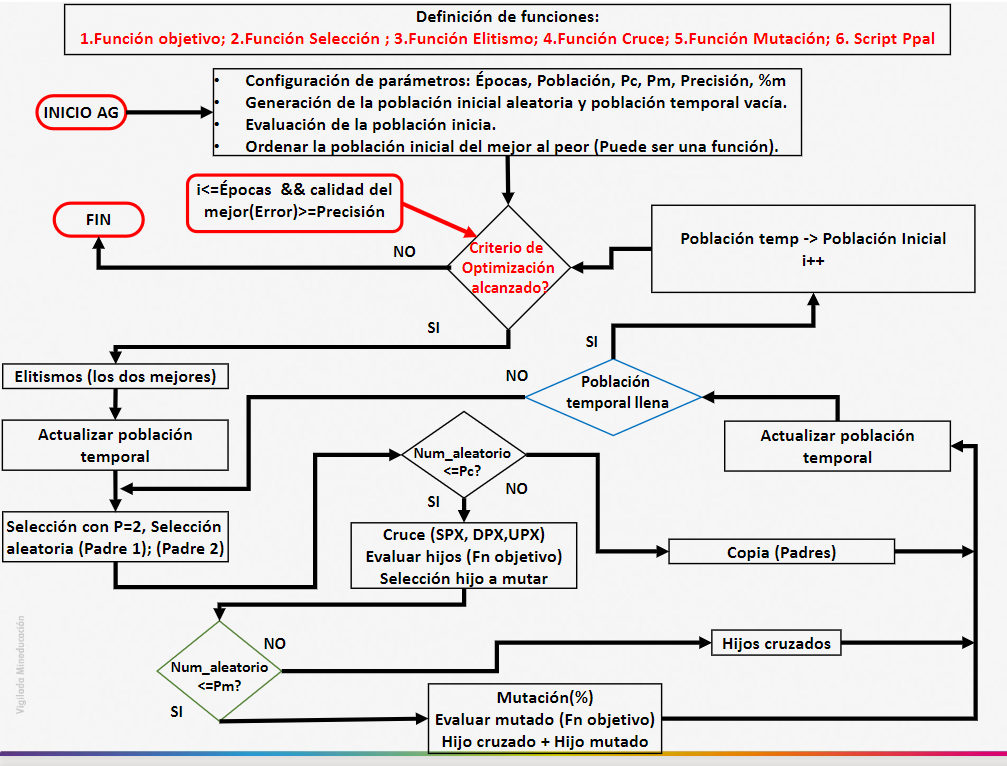


Fig 1 Recurso del curso

Basado en lo anterior se construyeron 2 códigos tanto en Python como en Matlab, los 2 se utilizaron para evaluar las siguientes ecuaciones lineales.

Siendo sus soluciones ideales X= 4.643, Y=0.821 Z=2.25

El objetivo del algoritmo genético era encontrar las soluciones x, y, z que mas se ajustara a las ecuaciones. Se decidió usar el algoritmo creado en Matlab ya que sus soluciones eran mas cercanas al ideal proporcionado, los resultados de nuestro algoritmo fueron los siguientes.

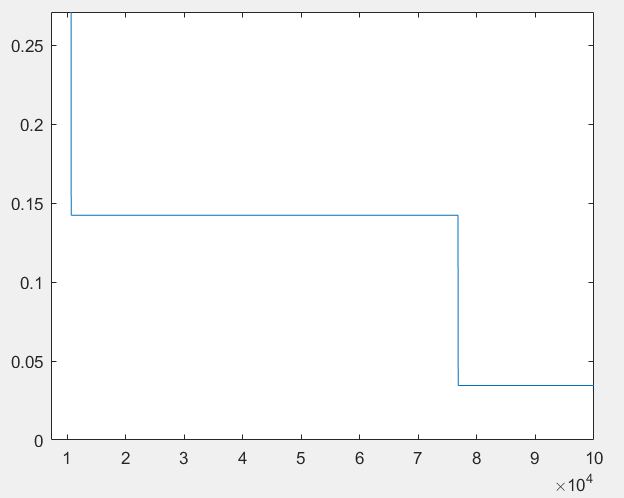


Fig 2 Recurso propio

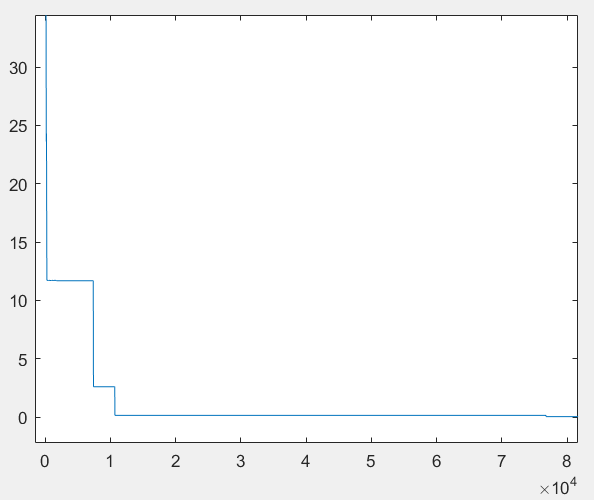


Fig 3 Recurso propio

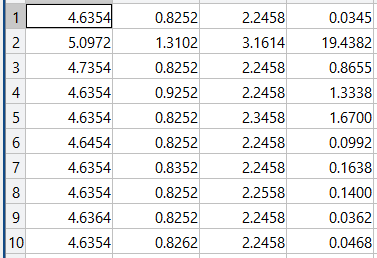


Fig 4 Recurso propio

Los fitness obtenidos se movían entre 0.5 y 0.01 por cada ejecución siendo el mejor de cada intento el primer valor de la tabla,

Nuestro algoritmo trabajo con una población de 30 individuos a un total de 100000 generaciones, si 2 mejores eran idénticos, realizaba una suma a todos los genotipos de el segundo individuo para así mantener un cruce efectivo, adicional a esto descartaba y agregaba nueva población al azar cada cierto ciclo, se nota en las gráficas como después de las 10mil generaciones es muy difícil que el fitness mejore pero en esta captura particular mejoro levemente una ultima vez antes de la generación 80.000

Los resultados obtenidos por el algoritmo realizado mediante las librerías de Matlab eran las siguientes.

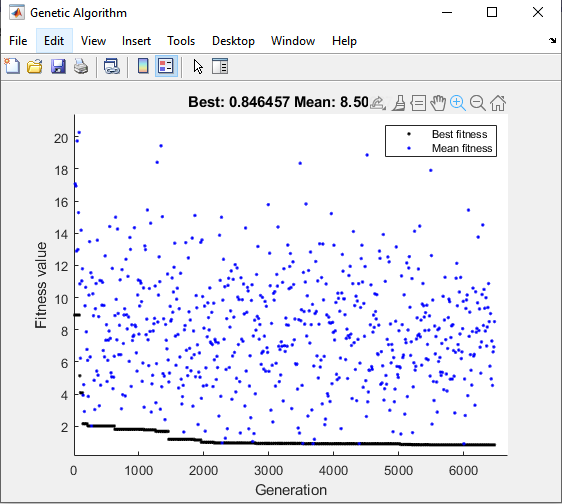


Fig 5 Recurso propio

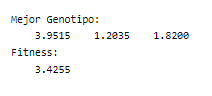


Fig 6 Recurso propio

En general los resultados del algoritmo de la librería son peores que el nuestro, pero esto puede deberse a una configuración inexacta de los parámetros del algoritmo o mala población inicial en las diferentes ejecuciones, además tiene ventajas en el procesamiento de datos ya que permite procesar y analizar los datos de una forma mas cómoda y da mas opciones al momento de detener la ejecución.

Los parámetros utilizados fueron los mismos una población de 30 individuos a 100 000 generaciones, pero se detuvo de forma autónoma después de las 60 mil por no detectar mejora sustancial.

2.2 Algoritmo genético y sistema PID

Para evaluar el sistema PID se consulto una planta a través de una tesis de grado [1], se extrajo la siguiente ecuación.

Los valores ideales del PID según el artículo son

P= 1.872, I=2, D=0.5

Los resultados al aplicar nuestro algoritmo genético a este sistema PID fueron los siguientes.

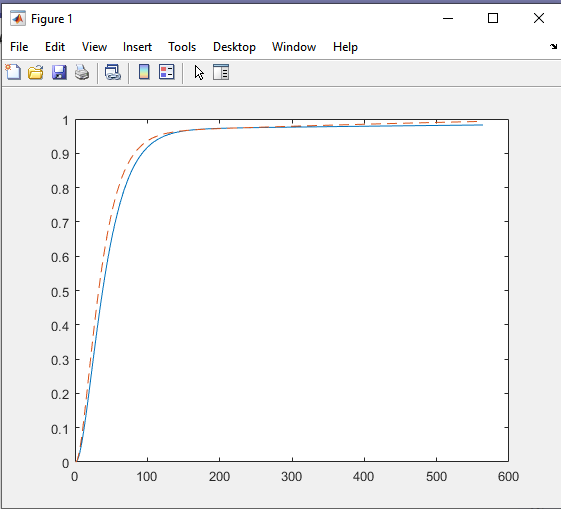


Fig 6 Recurso propio

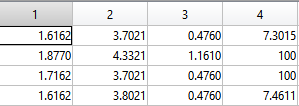


Fig 7 Recurso propio

Como se observa el elemento con mayor diferencia fue el Integral, aun así, a pesar de que su diferencia era sustancialmente mas alta no genero un efecto muy notable en la gráfica pudiéndose utilizar como otra solución aceptable. En la tabla también se muestran elementos cuyo fitness es 100, esto es una medida agregada al código para penalizar cualquier PID que no se estabilice en el tiempo requerido.

El código corrió con 30 de población inicial y 2000 generaciones, el sistema que modificaba la población cada ronda seguía siendo aplicado, debido a la forma de comparación se tuvo que reducir el número de interacciones, razón por la cual el fitness no se pudo ajustar en su totalidad.

Los resultados obtenidos del código del toolbox son los siguiente.

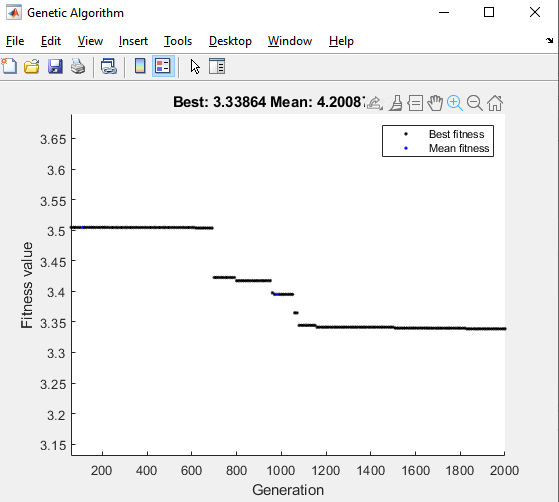


Fig 8 Recurso Propio

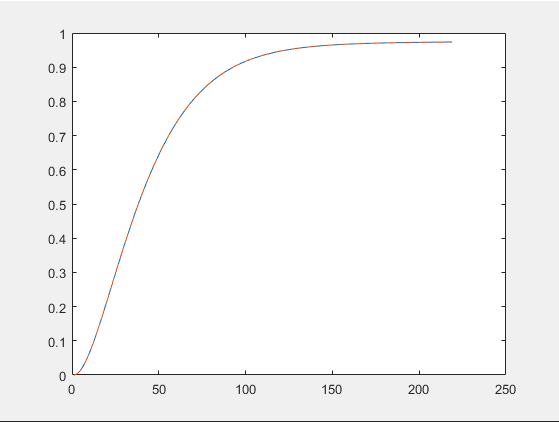


Fig 9 Recurso Propio

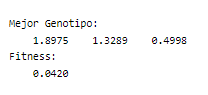


Fig 10. Recurso Propio

Los parámetros proporcionados a este sistema son los mismos que al algoritmo programado por nosotros, una población de 30 individuos a 2000 generaciones, los demás valores se dejaron por defecto.

Los datos obtenidos mediante el toolbox resultaron bastante mejor y la progresión de la mejora fue menor luego de las 1000 generaciones, quizá al aumentar el numero de generaciones el algoritmo se hubiera podio ajustar aun mas aunque se observa que al igual que en el anterior la variable Integral sigue siendo la mas desfasada con diferencia.

La diferencia mas notable es la velocidad de estabilización, se nota como este PID se puede considerar incluso mejor que la proporcionada por el articulo gracias a su velocidad de estabilización.

# Conclusion

## En general nuestros códigos tenían un mejor desempeño, pero esto se debía a que podíamos modificarlos con facilidad agregando soluciones especiales a problemas particulares fuera del diagrama de flujo otorgado al principio lo cual nos daba un mayor control de las variables y la población, los algoritmos facilitados por las toolboxes funcionaban de manera adecuada, pero fue difícil encontrar parámetros que superaran a los nuestros.

Debido a nuestras complicaciones con el algoritmo PID el toolbox tuvo un mejor desempeño en esta aplicación, aunque el tiempo de ejecución era igual de largo.

3. Referencias

[1] SARDOTH REDONDO. and CRUZ RUEDA, D.A.R.W.I.N.A.N.D.R.É.S. (2009) Desarrollo de Una Planta con control PID de temperatura, nivel y peso ... Available at: <http://repositoriodspace.unipamplona.edu.co/jspui/bitstream/20.500.12744/4194/1/Sardoth_Cruz_2019_TG.pdf>

(Accessed: March 11, 2023).