

Instituto Politécnico Nacional

Escuela superior de Cómputo

Algoritmos Bioinspirados

Práctica No. 4: Introducción a los algoritmos genéticos

Realizada por: Calva Guzmán Alan Alexis

Profesor Jorge Luis Rosas Trigueros

Realizada: 31/03/2025

Fecha de entrega: 31/03/2025

Marco teórico:

Los algoritmos genéticos (AG) son una técnica de optimización inspirada en la evolución biológica. Introducidos por John Holland en la década de 1970, estos algoritmos simulan procesos evolutivos como selección natural, cruce y mutación para encontrar soluciones óptimas a problemas complejos. Son especialmente útiles en problemas de optimización donde el espacio de búsqueda es grande y presenta múltiples óptimos locales. A diferencia de los métodos deterministas, los AG pueden explorar diferentes regiones del espacio de soluciones de manera simultánea, aumentando la probabilidad de encontrar el óptimo global.

Un algoritmo genético opera sobre una población de posibles soluciones, llamadas individuos o cromosomas. Cada cromosoma representa una solución potencial al problema y se codifica como una cadena de valores, generalmente en forma de un vector numérico o una cadena binaria. La población inicial se genera aleatoriamente dentro del dominio del problema. En el caso de la función de Rastrigin, cada individuo es un punto en el espacio bidimensional, representado por coordenadas (x, y) . Cada individuo es evaluado según una función de aptitud (fitness), que cuantifica qué tan buena es una solución. En la optimización de Rastrigin, el objetivo es minimizar la función, por lo que el fitness se define como el inverso del valor de la función, transformándola en un problema de maximización.

Se eligen individuos para reproducirse en base a su fitness. Los métodos comunes incluyen la selección por ruleta, donde la probabilidad de selección es proporcional al fitness; el torneo, donde se seleccionan aleatoriamente varios individuos y se elige el mejor; y el elitismo, que preserva los mejores individuos para la siguiente generación. El cruce combina la información genética de dos padres para generar descendencia. En la implementación de este proyecto, se usa un cruce aritmético, donde la nueva solución es una combinación ponderada de los padres. La mutación introduce cambios aleatorios en los descendientes para aumentar la diversidad genética y evitar el estancamiento en óptimos locales. En este caso, se utiliza una distribución normal para modificar ligeramente los valores de los individuos. Tras cada generación, se reemplaza la población antigua con la nueva. El

algoritmo se detiene cuando se alcanza un número máximo de generaciones o cuando la población converge a una solución estable.

La función de Rastrigin es una función de prueba utilizada en optimización global. Se define como $f(x, y) = 20 + x^2 + y^2 - 10 (\cos(2\pi x) + \cos(2\pi y))$. Su principal desafío es que contiene múltiples mínimos locales, lo que dificulta la búsqueda del mínimo global en $(0,0)$, donde $f(0,0) = 0$.

En esta práctica, el algoritmo genético busca el mínimo de la función de Rastrigin en dos dimensiones. Los individuos representan coordenadas en el plano y evolucionan a través de generaciones hasta aproximarse a $(0,0)$. Las dificultades incluyen la convergencia en mínimos locales, lo que puede solucionarse con una mayor tasa de mutación, selección adecuada y poblaciones más grandes. Este experimento permite comprender cómo los algoritmos genéticos abordan problemas de optimización complejos y evaluar estrategias para mejorar su desempeño.

Material y equipo:

Fué utilizado como auxiliar en la escritura de código el programa Codium, fueron generados entornos de python con Anaconda, utilizando la versión 3.9 de python; todo esto corriendo en Ubuntu 24.04.2 en un equipo de escritorio con 8 Gb de memoria RAM y un procesador Ryzen de 64 bits como características principales relevantes.

Fueron requeridas librerías matplotlib y numpy.

Desarrollo de la práctica:

La realización de la práctica parece más complicada de lo que en realidad termina siendo. O al menos así fué para este caso, para el que previo no conocía la función de Rastrigin, la cual impone bastante de primera vista. Dada la naturaleza del objetivo, que requiere un valor continuo para la función fitness, resulta más conveniente y fácil codificar los genomas en forma continua que en forma binaria. No obstante, con el final de la codificación no se termina la práctica, pues la población sin los parámetros adecuados se estanca muy fácilmente en mínimos locales, por lo que se requirió hacer variar bastante la tasa mutación principalmente, para que los objetos exploren áreas fuera del mínimo local en el que se estancan de vez en cuando.

Así, se logró pasar de iterar 100 generaciones y no lograr un mínimo global a alcanzarlo en menos de 15 generaciones para múltiples ejecuciones.

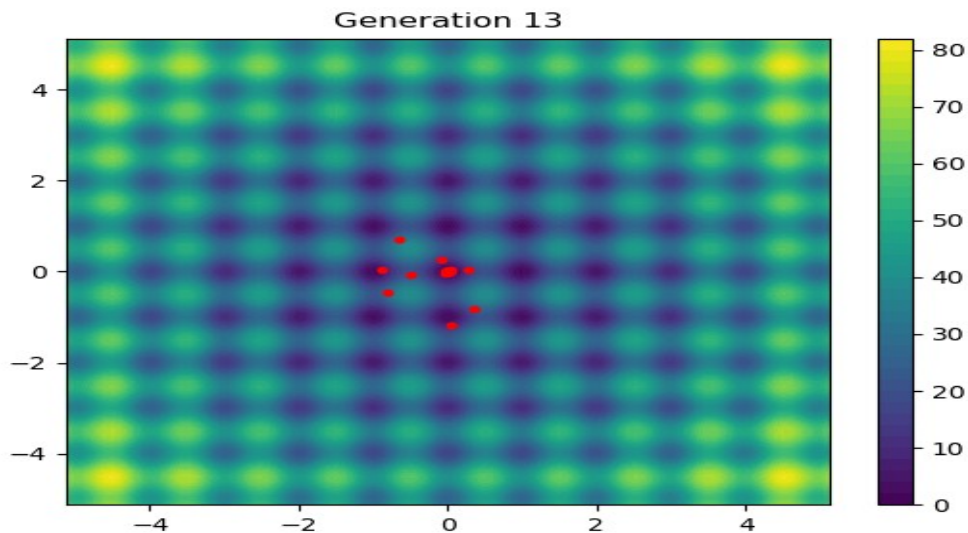


Figura 1: Mínimo global encontrado en 13 generaciones

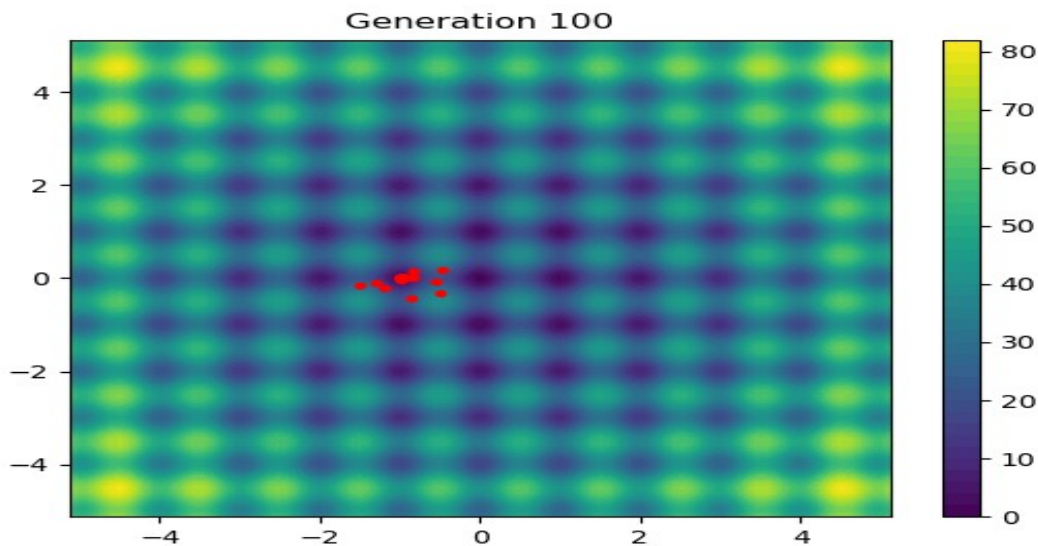


Figura 2: Población estancada en un mínimo local tras 100 generaciones

Conclusiones y recomendaciones:

Los algoritmos genéticos son bastante útiles para realizar tareas que parecen bastante complicadas. Al menos para este problema no se me ocurre una forma más fácil de hallar esta solución.

No obstante, también vale mucho la pena variar aleatoriamente (a falta de un mejor término) todos los parámetros, tales como la tasa de mutación, de ser posible experimentar con tipos diferentes de selección, y con los tamaños de las poblaciones, pues se observó que se pueden obtener soluciones muy buenas con menos generaciones al conseguir una buena combinación de estos parámetros.