



Universidad de Guadalajara
Centro Universitario de Ciencias Exactas e Ingenierías

Practica 3- Algoritmos de Optimización 2

Genetic Algorithm

Equipo:

Cornejo Chavez Edwin Joel

Veloz Alcaraz Axel Abraham

Espinoza Sucilla Samuel

Carrera: Ingeniería en computación

Materia: Seminario Inteligencia Artificial I

Maestro: Beltrán Carrillo Luis Ángel

Sección: D07 Viernes 9:00 AM a 1:00 PM.

Código: 214017739

Fecha: 17/11/2023

Introducción

Algoritmos de optimización.

En equipos:

Implementar el algoritmo elegido en cualquier lenguaje de programación. En este paso se debe experimentar con funciones matemáticas con la finalidad de visualizar y entender mejor el funcionamiento del método seleccionado.

En este caso, usaré una función matemática simple como la función de Rosenbrock para ilustrar el proceso. El objetivo es encontrar el mínimo de esta función utilizando el algoritmo genético.

```
Generación 16: Mejor valor de fitness = 0.0009505064031998709
Generación 17: Mejor valor de fitness = 0.0008551497838851341
Generación 18: Mejor valor de fitness = 0.0008551497838851341
Generación 19: Mejor valor de fitness = 0.0008551497838851341
Generación 20: Mejor valor de fitness = 0.0008551497838851341
Generación 21: Mejor valor de fitness = 0.0008551497838851341
Generación 22: Mejor valor de fitness = 0.0008551497838851341
Generación 23: Mejor valor de fitness = 0.0008551497838851341
Generación 24: Mejor valor de fitness = 0.0008551497838851341
Generación 25: Mejor valor de fitness = 0.0008551497838851341
Generación 26: Mejor valor de fitness = 0.0008344495774970788
Generación 27: Mejor valor de fitness = 0.0008551497838851341
Generación 28: Mejor valor de fitness = 0.0008551497838851341
Generación 29: Mejor valor de fitness = 0.0008551497838851341
Generación 30: Mejor valor de fitness = 0.0008551497838851341
Generación 31: Mejor valor de fitness = 0.0008551497838851341
Generación 32: Mejor valor de fitness = 0.0008551497838851341
Generación 33: Mejor valor de fitness = 0.0008551497838851341
Generación 34: Mejor valor de fitness = 0.0008551497838851341
Generación 35: Mejor valor de fitness = 0.0008551497838851341
Generación 36: Mejor valor de fitness = 0.0008551497838851341
Generación 37: Mejor valor de fitness = 0.0008551497838851341
Generación 38: Mejor valor de fitness = 0.0008551497838851341
Generación 39: Mejor valor de fitness = 0.0008551497838851341
Generación 40: Mejor valor de fitness = 0.0008551497838851341
Generación 41: Mejor valor de fitness = 0.0008551497838851341
Generación 42: Mejor valor de fitness = 0.0008551497838851341
Generación 43: Mejor valor de fitness = 0.0008551497838851341
Generación 44: Mejor valor de fitness = 0.0008551497838851341
Generación 45: Mejor valor de fitness = 0.0008551497838851341
Generación 46: Mejor valor de fitness = 0.0008551497838851341
Generación 47: Mejor valor de fitness = 0.0008551497838851341
Generación 48: Mejor valor de fitness = 0.0008551497838851341
Generación 49: Mejor valor de fitness = 0.0008551497838851341
```

Mejor solución encontrada:

Parámetros: [0.1112854 -4.04753633 4.44481288 0.46965622 0.57304377]

Valor de fitness: 1.881510307720997e-05

PS C:\Users\Nexus\Desktop\Universidad\Inteligencia artificial Tx

Aquí lo hicimos con 50 generaciones para encontrar un valor óptimo entre estas, de esta manera al final se compara entre todas cuál fue la mejor solución encontrada al finalizar el algoritmo, en este caso usamos la función matemática Rosenbrock para hacer la prueba del algoritmo ya que presenta problemas comunes del mundo real como la búsqueda en espacios de parámetros extensos y la presencia de múltiples órdenes de magnitud en las derivadas parciales.

```
    return best_individual

# Parámetros del algoritmo genético
pop_size = 100
chromosome_length = 5
generations = 50
```

En esta parte del código podemos ajustar los parámetros para encontrar o tratar de encontrar una mejor solución a la ya encontrada en nuestro algoritmo, en este caso utilizamos una longitud de 5 a un tamaño de 100 y generamos 50 pruebas.

```
Mejor solución encontrada:
Parámetros: [ 0.80862603 -2.89047088 -0.57118267  1.10658931  4.34334309]
Valor de fitness: 9.731570214517032e-05
```

Aquí hice una prueba generando 100 y podemos observar como los valores cambiaron a una distinta solución antes encontrada