

**Seminario de Ciencias de la Computación “A”
Modelación y simulación computacional
basada en agentes 2020-II**

**Práctica 3
“Algoritmos Evolutivos”**

PARTE 1. Algoritmo Genético Simple

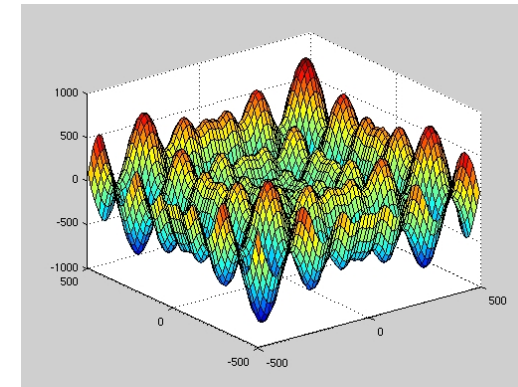
Implementar en el lenguaje de programación de su preferencia el esquema visto en clase del Algoritmo Genético Simple para maximizar funciones de dos variables considerando los siguientes puntos.

- a) Representación de los individuos.** Bit-strings.
- b) Inicialización aleatoria de la población.** A través de un generador de números pseudo-aleatorios crear la población inicial.
- c) Tamaño de la población.** Constante.
- d) Condición de término.** Usar alguna de estas dos estrategias, la primera, con base en el tiempo de ejecución, es decir, número de generaciones. La segunda con base en la aptitud (fitness) de los individuos (por ejemplo, si no existe mejoría en la aptitud de los mejores individuos en un número k de generaciones).
- e) Mecanismo de Selección.** Proporcional a su aptitud (Roulette wheel selection).
- f) Recombinación.** 1-Point crossover.
- g) Mutación.** Bit-flip. Misma probabilidad para todo el genotipo.
- h) Operador de reemplazo.** Generacional, es decir, se renueva toda la población en la generación siguiente.

Encontrar el **máximo global** para la función **Schwefel** en su versión de dos variables

$$f(\vec{x}) = \sum_{j=1}^d x_j \sin(\sqrt{|x_j|})$$

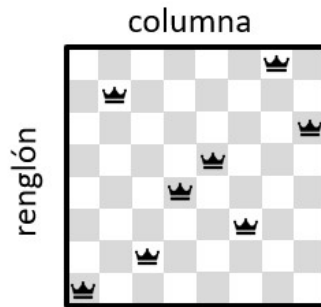
Donde $d=2$, en el intervalo $[-500,500] \times [-500,500]$.



1. Generar posibles soluciones con una precisión de 6 decimales. Escribe tu modelación.
2. Establecer una población fija de 100 individuos, la tasa de recombinación en $P_c=0.5$, la tasa de mutación en $P_m=0.1$, la condición de término en 500 generaciones. ¿Cuál fue el mejor individuo al terminar la ejecución? Reporte la estructura del genotipo, fenotipo y fitness ¿Es el máximo global? Realizar la gráfica, a) generación vs mejor aptitud de la población, b) generación vs promedio del fitness de la población, c) Grafique en el plano (x,y) , los individuos de la generación 1, 10, 50 y 100 para comparar su desempeño.
3. Genere otros experimentos cambiando la tasa de recombinación y de mutación. A partir de su experiencia trate de ajustar los parámetros para que el AGS tenga el mejor desempeño posible Graficar como en los puntos anteriores. ¿Cuáles fueron los parámetros con los que encontraron rápidamente el máximo global? Explique.

PARTE 2. El problema de las 8 reinas.

El problema de las ocho reinas consiste en colocar 8 reinas en el tablero de ajedrez de tal manera que no se “coman” entre sí. Este problema fue propuesto por Max Bezzel en 1848. Existen 92 posibles soluciones, contando las reflexiones y rotaciones.



Ejemplo de una configuración donde hay varias amenazas.

A partir del planteamiento realizado en la clase, resolver este problema a través de algoritmos evolutivos. Implemente una solución con las siguientes características:

| Componente | Tipo |
|-------------------------------|---|
| Representación | Permutación |
| Recombinación | Cut & crossfill |
| Probabilidad de recombinación | 100% |
| Mutación | Intercambio (swap) |
| Probabilidad de mutación | 80% |
| Selección de padres | Mejores 2 de 5 aleatorios |
| Operador de reemplazo | Reemplazar a los peores |
| Tamaño de la población | 100 individuos |
| Inicialización | Aleatoria |
| Condición de paro | Encontrar la solución o evaluación de 10,000 fitness. |

1. Reporte la solución: estructura del genotipo, fenotipo y fitness Realizar las gráficas: a) generación vs mejor aptitud de la población, b) generación vs promedio del fitness de la población.
2. Con su implementación resuelva el problema para n-reinas, donde $n = 30$.

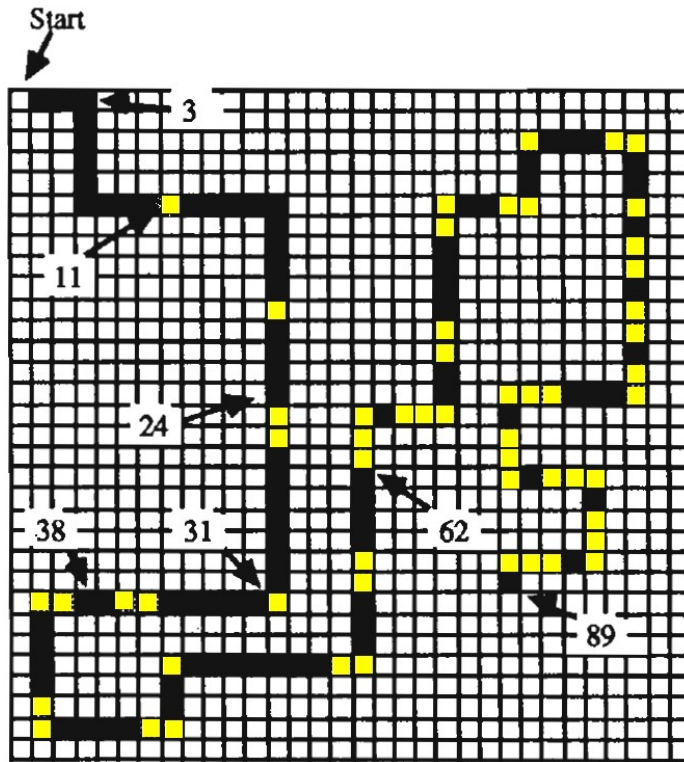
PARTE 3. Hormiga artificial

En 1991 David Jefferson y Robert Collins desarrollaron un mecanismo capaz de “adaptar” un organismo artificial a un ambiente dado. El objetivo es conducir una hormiga artificial a través de un camino irregular formado por rastros de comida.

La hormiga artificial se desarrolla en una retícula toroide de 32x32 sobre el plano. La hormiga inicia su recorrido en la celda superior izquierda identificada con las coordenadas (0,0) viendo hacia el este. El sistema de coordenadas es (renglón, columna).

El recorrido que intentará la hormiga es el llamado “*Santa Fe Trail*” el cual consiste en recolectar 89 migajas de comida distribuidas sobre la retícula como se muestra en la figura siguiente. Las celdas en negro son las migajas de comida y las celdas en amarillo son huecos donde se pierde el rastro, existen huecos únicos, huecos dobles, huecos únicos en las orillas y dobles huecos en las orillas. Este rastro fue diseñado por Christopher Langton. Los números indican el conteo de las 89 migajas de comida.

La hormiga artificial tiene una visión limitada del mundo. En particular la hormiga tiene un sensor con el cual solo puede ver lo que hay inmediatamente en la celda adyacente en la dirección que está viendo.



La hormiga puede ejecutar cualquiera de las siguientes acciones:

- RIGHT: gira la hormiga 90° a la derecha (sin moverse de su lugar).
- LEFT: gira la hormiga 90° a la izquierda (sin moverse de su lugar)
- MOVE: se mueve la hormiga hacia adelante en la dirección que está “viendo” Cuando se mueve a la celda siguiente se come la comida, si es que hay, eliminando el rastro de comida.

La hormiga también cuenta con las siguientes funciones:

- IF-FOOD-AHEAD: es un operador condicional de 2 argumentos, si tiene comida enfrente ejecuta alguna instrucción, si no, ejecuta la segunda instrucción.
- PROGN2: es un operador no condicional secuencial de 2 argumentos, ejecuta la primera instrucción e inmediatamente después la segunda instrucción.
- PROGN3: lo mismo que PROGN2 solo que con 3 argumentos.

A través del paradigma de la programación genética encontrar una estrategia adecuada para que la hormiga artificial se coma todas las migajas de comida en un tiempo finito. El problema queda determinado por la siguiente tabla:

| | |
|-------------------|--|
| Objetivo | Encontrar un programa que controle el funcionamiento de la hormiga artificial tal que encuentre las 89 migajas de comida localizadas en el “rastros Santa Fe”. |
| Conjunto Terminal | (LEFT), (RIGHT), (MOVE) |
| Conjunto Función | IF-FOOD-AHEAD, PROGN2, PROGN3 |
| Casos del Fitness | Un solo caso |
| Raw Fitness | Número de piezas de comida levantadas antes de 400 operaciones, es decir, antes de la aplicación de 400 instrucciones del conjunto de terminales. |
| Parámetros | Población 500, Generaciones=n, recomendado n = 51. |

Ejercicios:

1. Encontrar el programa (árbol) más adecuado (óptimo) para resolver el problema.
2. Realizar la gráfica generación vs fitness del mejor individuo y generación vs promedio del fitness.

Lineamientos para la entrega

La práctica se puede realizar de manera individual o en equipo de máximo dos personas.

Entregar un empaquetado (.tgz, .zip, etc.) con la siguiente estructura:

```
Apellido1-Apellido2
|----- PracticaN
|               |----- fuentes
|               |----- readme.txt
|               |----- solución a ejercicios.pdf
```

Fuentes: código del programa, ejecutables, etc.

Readme.txt: explicar brevemente como se ejecuta el programa, y la forma en que podemos cambiar parámetros.

SolucionEjercicios.pdf: En este archivo incluir todas las respuestas de la práctica junto con las gráficas y comentarios extras (si los hay).

Enviar la práctica al mail gcarreon@unam.mx y vanchristoph3r@gmail.com

Fecha de entrega: viernes 29 de enero de 2021 antes de las 24:00 hrs.

Contacto

Gustavo Carreón gcarreon@unam.mx

Christopher Chávez vanchristoph3r@gmail.com

Lista del curso evolutivo@listas.iiec.unam.mx