

GRUPO 06



Christian González García-Muñoz
Alejandro Huertas Herrero

Descripción representación

La representación elegida ha sido un árbol. El árbol tiene distintos tipos de nodos dependiendo del operador que contenga el nodo.

- Si se trata de un nodo “NOT” tiene sólo el hijo izquierdo.
- Si se trata de un nodo “AND” o “OR” tiene los nodos izquierdo y derecho.
- Si se trata de un nodo “IF” tiene tres nodos, izquierdo, central y derecho.

Para identificar cada uno de los hijos utilizamos un atributo de la clase (`_position`) que puede tomar tres valores:

- 0 es el hijo izquierdo.
- 1 es el hijo central.
- 2 es el hijo derecho.

Operadores utilizados

Los operadores utilizados son los cuatro que se menciona en el enunciado (“IF”, “NOT”, “OR” y “AND”) En cuanto a operadores de mutación, cruce y selección, están implementados todos los disponibles y explicados en clase.

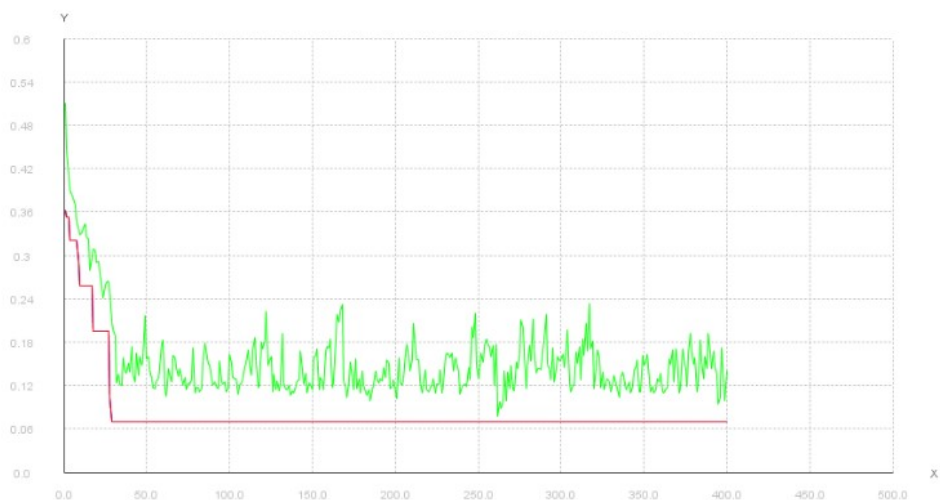
Función de aptitud

La función de aptitud comprueba cuantos aciertos tiene el cromosoma respecto a las 64 combinaciones posibles. Una vez calculados, lo divide entre 64, para calcular el tanto por uno de aciertos. Al resultado, se le aplican las operaciones que se mencionan en el apartado siguiente de la memoria.

Mejoras utilizadas

Para controlar el bloating, se ha aplicado la siguiente medida. Al número de aciertos del cromosoma se le suma la multiplicación de un porcentaje (introducido por la GUI) por el número de nodos que tiene el árbol del cromosoma. Con el porcentaje se indica cuanta importancia se le quiere dar al control del bloating.

Mejores ejecuciones



■ Absolute best ■ Generation best ■ Generation mean

Results

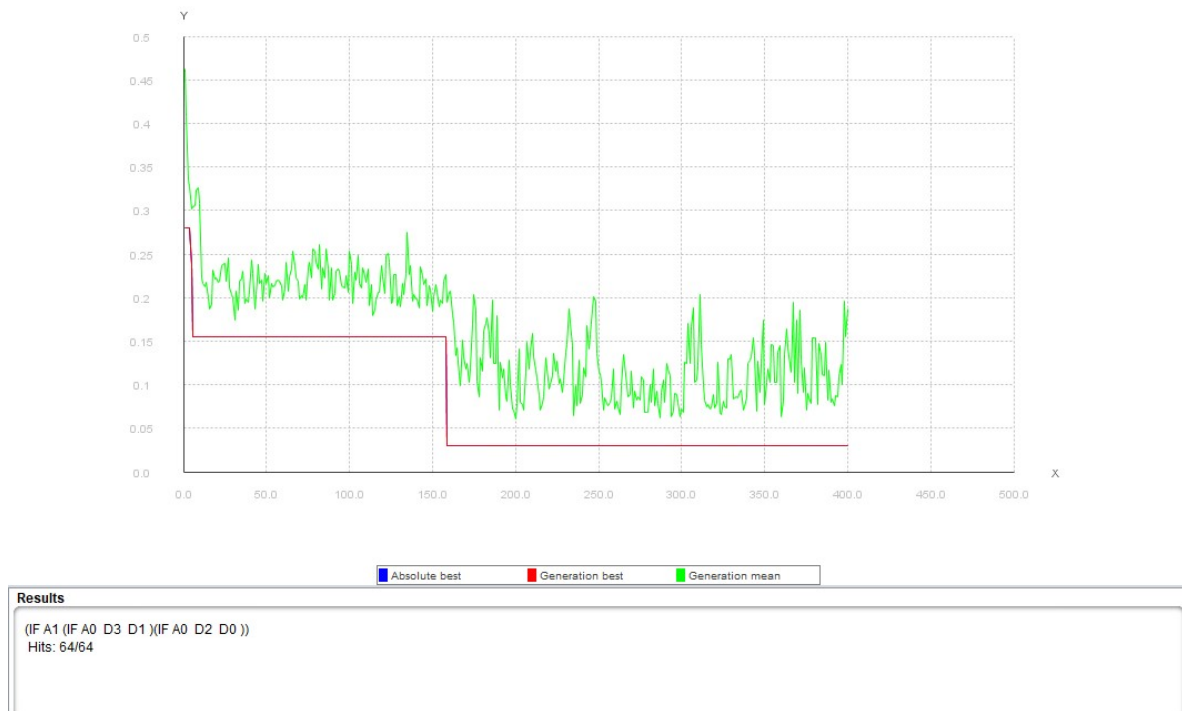
(IF (IF D1 A0 A0) (IF A1 D3 D2) (IF A1 D1 D0)))
Hits: 64/64



■ Absolute best ■ Generation best ■ Generation mean

Results

(NOT (NOT (IF A0 (IF A1 D3 D2) (IF A1 (IF A1 D1 A0) D0)))))
Hits: 64/64



Ejecución realizada utilizando sólo ifs.

Como conclusión, hemos observado que la probabilidad de conseguir una buena solución aumenta si se utilizan sólo ifs y full como método de inicialización, aunque las soluciones son más grandes. Y si utilizamos sólo ifs y grow, a veces saca soluciones peores, pero cuando saca una óptima suele ser corta.