

# **Programación Evolutiva**

## **Práctica 1**

*Roberto Plaza Hermira*  
*Gorka Silva Ramón*

## Función 1.

Encontrar el máximo de la función:

$$f(x_1, x_2) = 21.5 + x_1 \cdot \sin(4\pi x_1) + x_2 \cdot \sin(20\pi x_2) :$$

que presenta un máximo de **38.809** en 11.625 y 5.726  $x_1 \in [-3.0, 12.1]$   $x_2 \in [4.1, 5.8]$

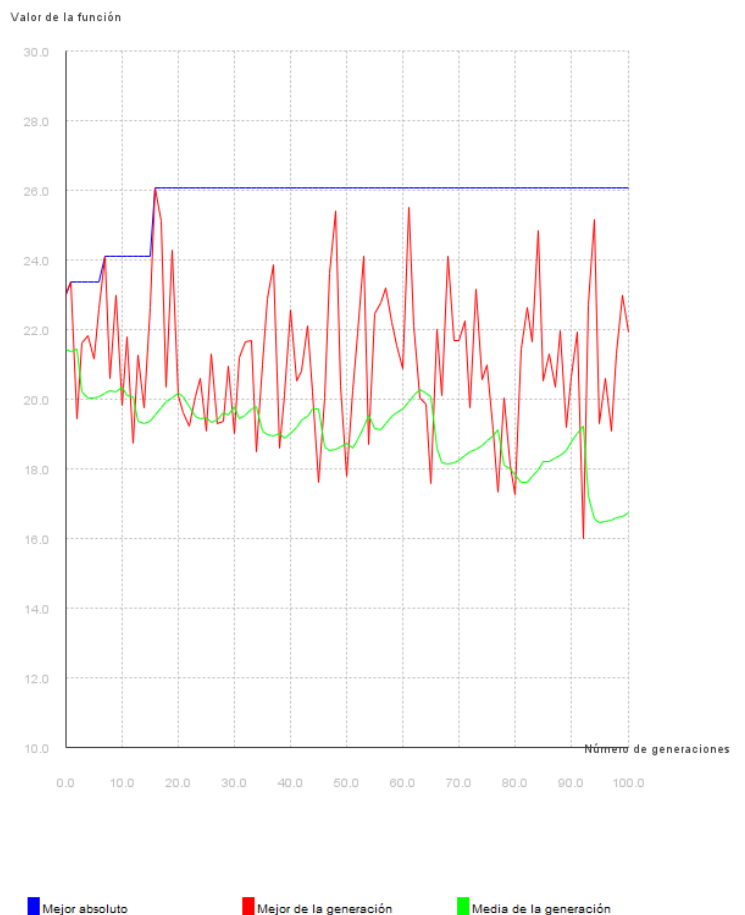
Todos los resultados obtenidos con diferentes selecciones y cruces dan resultados parecidos en un rango aproximado de [26.0, 27.9]. El mejor resultado (27.83) se ha conseguido con:

- Selección Torneo Determinístico
- Cruce Monopunto

En general, en pocas generaciones se obtiene un máximo global y a partir de la generación 50 aproximadamente ya no se ven signos de mejoría. Aunque se hagan variaciones en las probabilidades de cruce y de mutación no se observan cambios significativos.

Si que se puede ver una pequeña mejora cuando se aumenta el porcentaje de élite, pero no ayuda a la mejora a largo plazo de las generaciones.

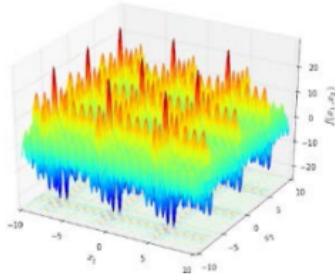
Imagen de ejemplo de cómo funcionan en general los algoritmos independientemente del tipo de selección y de cruce.



## Función 2.

Encontrar el mínimo de la función:

$$f(x_i, i = 1..2) = \left( \sum_{i=1}^5 i \cdot \cos((i+1)x_1 + i) \right) \left( \sum_{i=1}^5 i \cdot \cos((i+1)x_2 + i) \right)$$



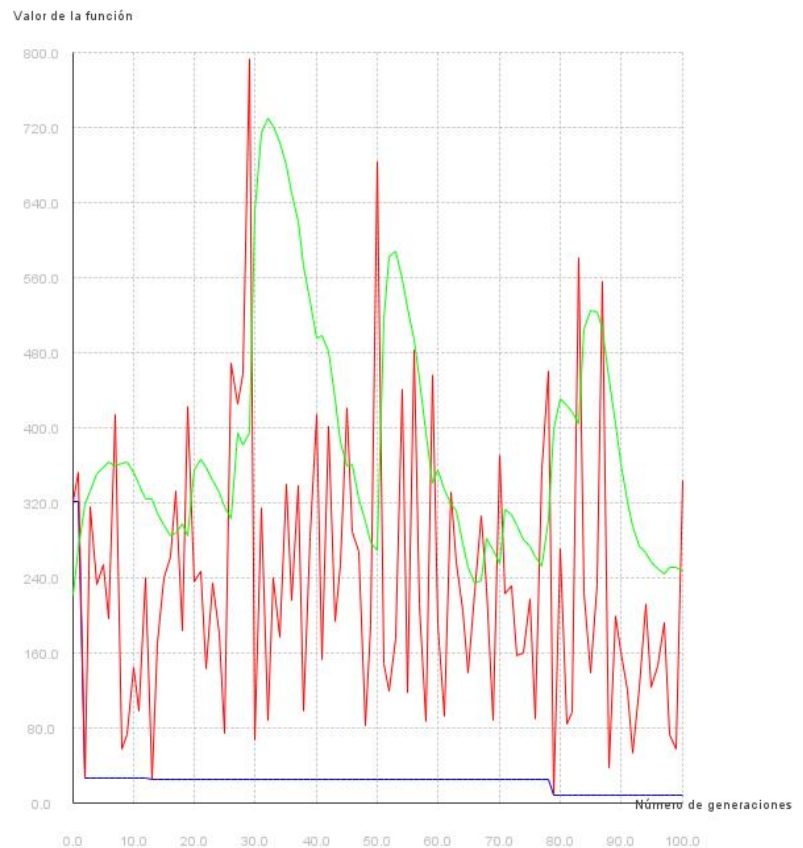
$x_i \in [-10, 10]$  que presenta 18 mínimos de **-186.7309**

Los resultados conseguidos son bastantes pobres. Están comprendidos en un rango de [7.1, 40.0] aproximadamente. El mejor resultado (7.19) se ha conseguido con:

- Selección Torneo Probabilístico
- Cruce Uniforme

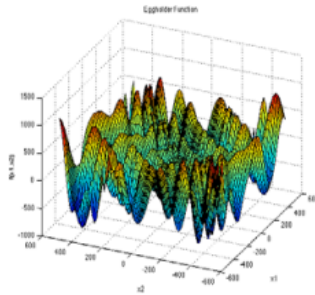
Al igual que en la función 1 es difícil ver mejorías a lo largo de las generaciones.

Esta es la imagen del mejor resultado obtenido. En general la media de la generación no parece mejorar aunque se aumente el tamaño de la población o se varíen las probabilidades de cruce o de mutación.



### Función 3.

Encontrar el mínimo de la función:



$$f(\mathbf{x}) = -(x_2 + 47) \sin \left( \sqrt{\left| x_2 + \frac{x_1}{2} + 47 \right|} \right) - x_1 \sin \left( \sqrt{\left| x_1 - (x_2 + 47) \right|} \right)$$

que presenta un mínimo de -959.6407 en (512, 404.2319)  $x_1, x_2 \in [-512, 512]$

Los resultados están comprendidos en un rango [-771.0 , -490.0] aproximadamente. Al igual que en otras funciones suelen encontrar los máximos globales en las primeras 50 generaciones y la media de generaciones no suele mejorar con el tiempo.

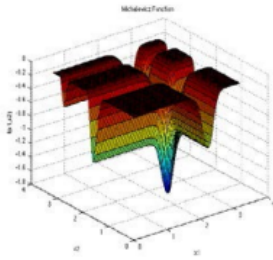
En el mejor resultado obtenido se ha conseguido mejorar a partir de las 50 generaciones cosa que era más difícil de ver en las funciones anteriores.

En este caso también se ve una pequeña mejora cuando se aumenta el porcentaje de élite, pero no se observan mejores resultados con un mayor tamaño de población



## Función 4 con representación binaria.

Encontrar el mínimo de la función:



$$f(x_i | i = 1..n) = - \sum_{i=1}^n \sin(x_i) \sin^{20} \left( \frac{(i+1)x_i^2}{\pi} \right) : x_i \in [0, \pi]$$

$x_i \in [0, \pi]$  que presenta los siguientes mínimos en función de  $n$ :

$n$	1	2	3	4	5	6	7
Mínimo	-1	-1.959091	-2.897553	-3.886358	-4.886358	-5.879585	-6.862457

No hay mucha diferencia entre los tipos de selección y cruces pero se podría decir que en general los mejores son:

- Selección Estocástico Universal
- Cruce Monopunto Binario

A diferencia de los problemas anteriores este si que mejora si se aumenta el tamaño de la población.

$n$	1	2	3	4	5	6	7
Mínimos	-0.489	-1.394	-1.921	-2.586	-2.714	-3.116	-3.408

*Mínimos son los mejores valores obtenidos después de ejecutar varias veces y con varios tipos de selección y cruce los algoritmos con diferente  $n$ .*

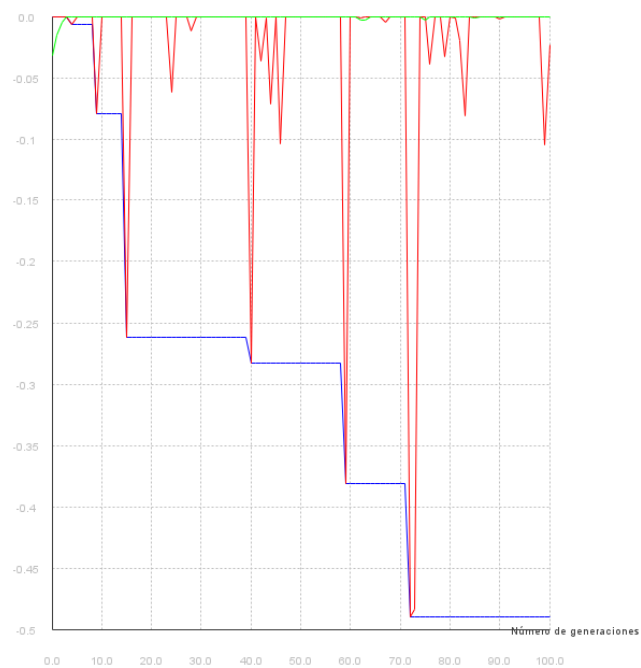
Número de variables 1 Problema 4-Bin

Tamaño población 100  
Número de Generaciones 100  
Valor de error 0.001  
Tipo de selección `algoritmoGenetico.seleccion.SeleccionEstocasticoUniversal@791ef39c`  
Tipo de cruce `algoritmoGenetico.cruce.CruceUniforme@68bd33a7`  
Probabilidad de cruce 60.0  
Tipo de mutación `algoritmoGenetico.mutacion.MutacionBasica@1e590cff`  
Probabilidad de mutación 5.0  
Élite 2.0

Grafica



Valor de la función



■ Mejor absoluto

■ Mejor de la generación

■ Media de la generación

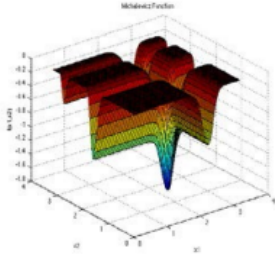
Reset

Solución:  $X_0 = 2.0420000000000003$ , Valor de la función: -0.48950665446584857

Ejecutar

## Función 4 con representación real.

Encontrar el mínimo de la función:



$$f(x_i|i = 1..n) = - \sum_{i=1}^n \text{sen}(x_i) \text{sen}^{20} \left( \frac{(i+1)x_i^2}{\pi} \right) : x_i \in [0, \pi]$$

$x_i \in [0, \pi]$  que presenta los siguientes mínimos en función de  $n$ :

$n$	1	2	3	4	5	6	7
Mínimo	-1	-1.959091	-2.897553	-3.886358	-4.886358	-5.879585	-6.862457

Esta representación hace que se obtengan resultados mucho mejores con  $n = \{1, 2, 3\}$ , pero peores resultados en las demás. Esto se puede comprobar en la imagen del mejor resultado con  $n = 7$ . En general obtiene resultados medios, pero para conseguir un resultado bueno depende en gran medida del azar.

En el caso de  $n > 3$  los resultados son casi totalmente independientes del tipo de cruce o selección.

$n$	1	2	3	4	5	6	7
Mínimos	-0.800	-1.738	-2.299	-2.733	-2.660	-3.393	-3.564

Número de variables

7

Problema

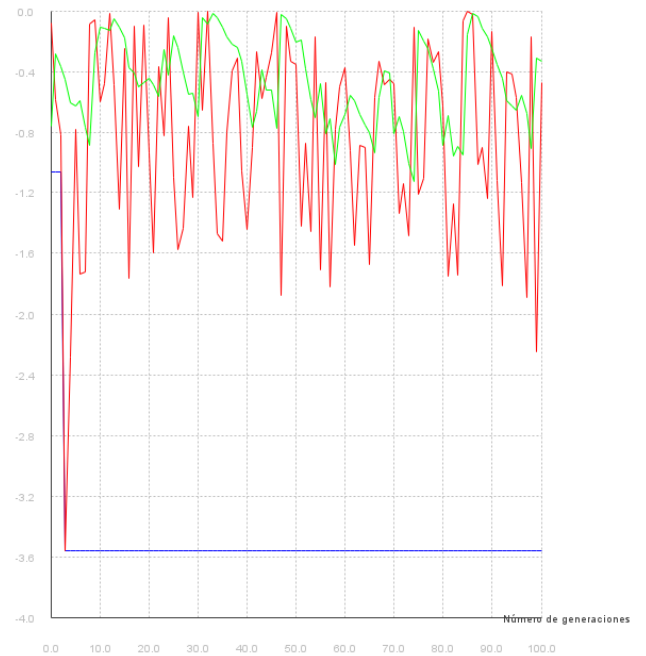
4-Real

Tamaño población	100
Número de Generaciones	100
Valor de error	0.001
Tipo de selección	algoritmoGenetico.seleccion.SeleccionEstocasticoUniversal@791ef39c
Tipo de cruce	algoritmoGenetico.cruce.CruceAritmetico@1f6bb2a1
Probabilidad de cruce	60.0
Tipo de mutación	algoritmoGenetico.mutacion.MutacionUniforme@377ffd9f
Probabilidad de mutación	5.0
Élite	2.0

Gráfica



Valor de la función



■ Mejor absoluto

■ Mejor de la generación

■ Media de la generación

Reset

Solución: Valor de la función: -3.5641291821736836

Ejecutar



## **Reparto de tareas.**

Roberto se ha encargado de crear:

- Los diferentes Individuos.
- Las funciones de Selección, Cruce y Mutación.
- Algunas funciones de la clase AlgoritmoGenetico.

Gorka se ha centrado en:

- Hacer la GUI.
- Crear la estructura general del programa.
- Hacer la parte general de la clase AlgoritmoGenetico.