Operadores de codificación para números reales

Para usar al máximo los algoritmos genéticos, se debe comprender que representar números reales usando listas binarias o enteras está lejos de ser ideal y, en cambio, las listas (o matrices) de números con valores reales se consideran un enfoque más real y de mayor aplicación.

Algunos de los operadores de codificación-real comúnmente utilizados son:

- Blend Crossover (también conocido como **BLX**), donde cada descendencia se selecciona aleatoriamente del siguiente intervalo creado por sus padres:

[padre1- α(padre2 - padre1), padre2 + α(padre2 - padre1)]

El valor α se establece comúnmente en 0.5, lo que resulta en un intervalo de selección dos veces más amplio que el intervalo entre los padres.

- Cruce binario simulado (**SBX**), donde los dos descendientes se crean a partir de los dos padres utilizando la siguiente fórmula, garantizando que el promedio de los valores de la descendencia sea igual al de los valores de los padres:

descendiente1 = 1/2[(1 + β)padre1 + (1 - β) padre2]

descendiente2 = 1/2[(1 - β)padre1 + (1 + β) padre2]

El valor de β, también conocido como factor de propagación, se calcula utilizando una combinación de un valor elegido aleatoriamente y un parámetro predeterminado conocido como η (eta), índice de distribución o factor de hacinamiento. Con valores más grandes de η, la descendencia tenderá a ser más similar a sus padres. Los valores comunes de η están entre 10 y 20.

- **Mutación** normalmente distribuida (o **gaussiana**), donde el valor original se reemplaza por un número aleatorio que se genera utilizando una distribución normal, con valores predeterminados para la media y la desviación estándar.

Uso de DEAP con funciones continuas

Para la codificación cromosómica, podemos usar una lista (o matriz) de números de punto flotante. Una cosa a tener en cuenta, sin embargo, es que los operadores genéticos existentes de DEAP no funcionarán bien con objetos individuales. El uso de individuos requerirá redefinir los operadores genéticos en consecuencia. Esto se trata con más detalle en la documentación de DEAP, en Heredar de NumPy. Por esta razón, así como por razones de rendimiento, generalmente se prefieren listas ordinarias de Python o matrices de números de punto flotante cuando se usa numpy.ndarray.

Optimización de la función de Himmelblau

Una función muy utilizada para medir el rendimiento (benchmarking) de los algoritmos de optimización es la función de Himmelblau, representada en el siguiente diagrama:

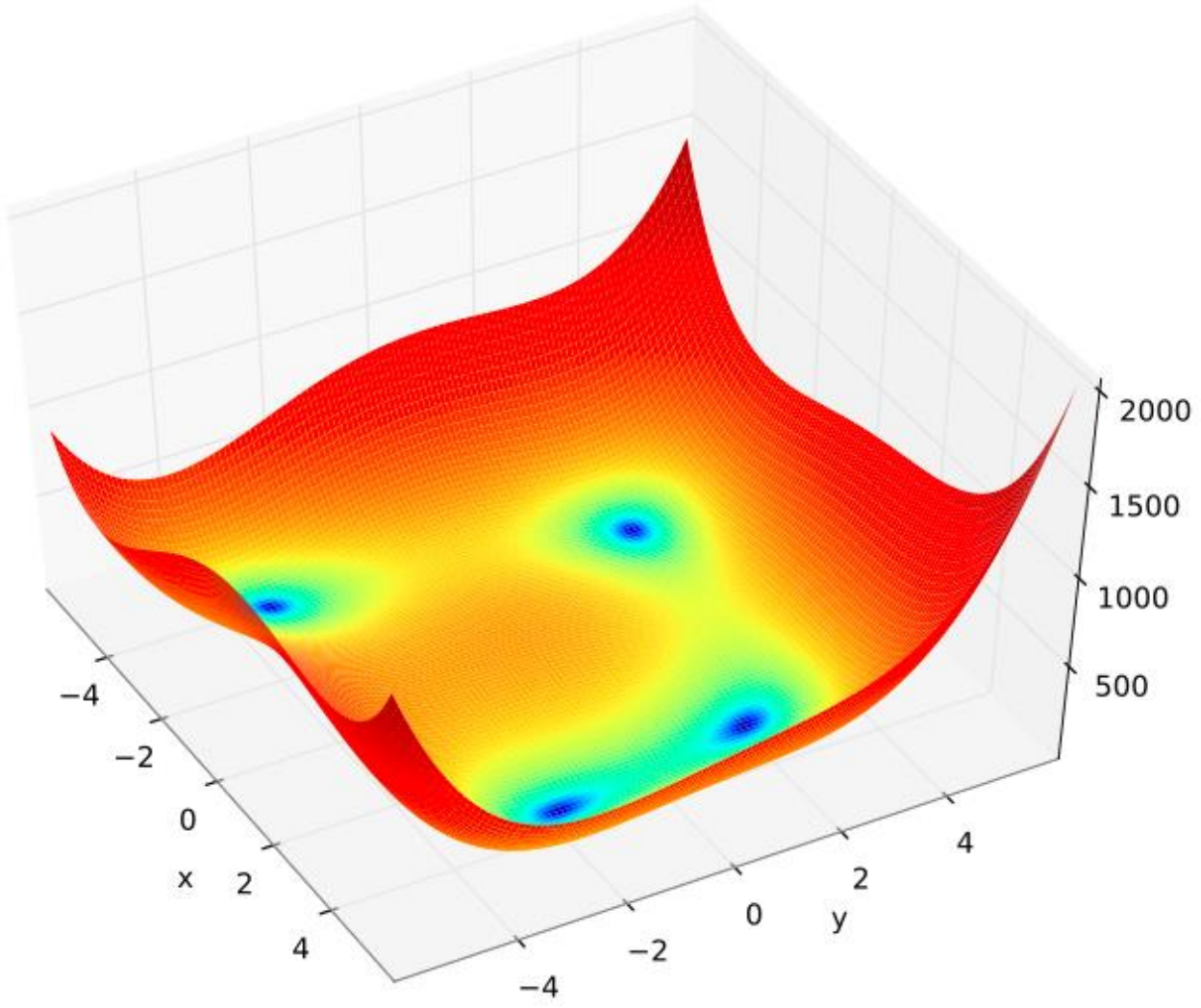

Figura 5. Función de Himmelblau. File:Himmelblau function.svg - Wikimedia Commons

Usar niching y compartir para encontrar múltiples soluciones

En la comunidad de GA es bien sabido que el uso compartido de algoritmos genéticos imita la forma en que un entorno natural se divide en múltiples subentornos o nichos. Estos nichos están poblados por diferentes especies, o subpoblaciones, aprovechando los recursos únicos disponibles en cada nicho, mientras que los ejemplares que coexisten en el mismo nicho tienen que competir por los mismos recursos. La implementación de un mecanismo de intercambio dentro del algoritmo genético alentará a la población a explorar nuevos nichos y se puede utilizar para encontrar varias soluciones óptimas. Una forma común de lograr **optimizar es dividir el valor bruto de aptitud de cada individuo** con alguna función que comparta las distancias combinadas de todos los demás individuos, penalizando efectivamente a una población abarrotada al compartir la recompensa local entre sus individuos.

https://www.macti-meia.unam.mx/lms/pluginfile.php/5035/mod_page/content/14/PID_multiObjetivo.pdf

PID con Raspberry Pi