

Reporte práctica siete: Búsqueda Local

José Anastacio Hernández Saldaña

Posgrado de Ingeniería de Sistemas

1186622

jose.hernandezsld@uanl.edu.mx

26 de septiembre de 2017

Resumen

Este es un reporte sobre la práctica siete con respecto al tema de búsqueda local que se realizó en la clase de Simulación de sistemas, cómputo paralelo en R.

1. Tarea y Extra Uno: Encontrar el valor máximo de la función $g(x, y)$

Se conoce como búsqueda local a un procedimiento al que dada una función g , llamada función objetivo, con parámetros x_1, x_2, \dots, x_n busca los valores de x_1, x_2, \dots, x_n para los cuales g es un cota superior o inferior, según sea el caso, dentro del dominio de los parámetros. Este tipo de búsqueda es un problema común en optimización, donde regularmente se busca maximizar un beneficio o minimizar un costo.

Para esta práctica se estudió la función $g(x, y) = (x + 0.5)^4 - 30x^2 - 20x + (y + 0.5)^4 - 30y^2 - 20y$ dentro del dominio $x \in [-6, 5]$ y $y \in [-6, 5]$ y se busca encontrar los valores de x y y que maximizan el valor de g , en la figura 1 podemos ver una representación en tres dimensiones de esta función..

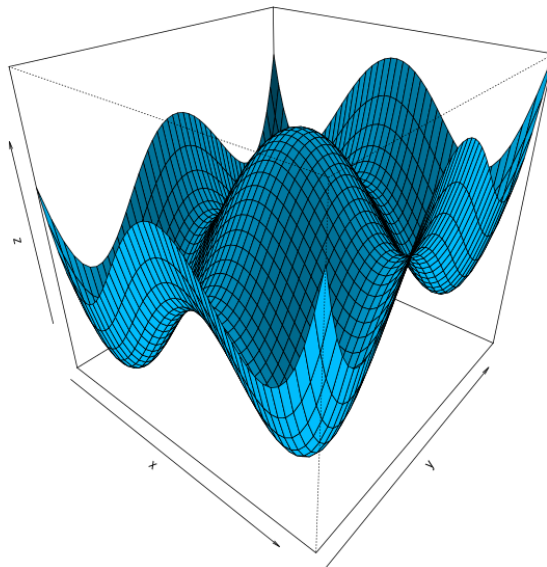


Figura 1: Gráfica en tres dimensiones de $g(x, y)$

1.1. Diseño del Experimento

Para el experimento se tomó como base el código de la página del curso, donde se encuentra la búsqueda local para maximizar la función ya programada. esta búsqueda local es una búsqueda voraz o *greedy*, es decir, dado un valor de x y y busca un mejor valor dentro de un conjunto valores x y y vecinos, si alguno de los vecinos al ser evaluado con la función g nos da un valor mayor lo toma como solución candidata y busca dentro de los vecinos de ese nuevo valor, esto lo repite hasta no encontrar mejores valores hacia los cuales desplazarse.

Uno de los principales problemas de este tipo de búsqueda local es que fácilmente puede converger a un óptimo local, es decir, una cota superior para un intervalo contenido en el dominio. Dado que toda cota es un óptimo local son necesarios métodos que nos ayuden a guiar la búsqueda para evitar óptimos locales. Por lo que se decidió utilizar esta búsqueda desde varios puntos generados al azar dentro del dominio, esto para tener una diversidad de valores iniciales y con la ayuda de la búsqueda local explorar el espacio de soluciones. Este método es conocido como un método multiarranque; como cada búsqueda es independiente a las demás, se decidió paralelizar este procedimiento.

Al estar trabajando con una función tridimensional, es necesario utilizar representaciones de dos dimensiones que faciliten la visualización del proceso de búsqueda, por lo que se utilizó una gráfica de contorno de superficie, para la representación y poder trazar algunos caminos de la búsqueda señalando los inicios y fines, podemos observar un ejemplo de este tipo de gráfica en la figura 2.

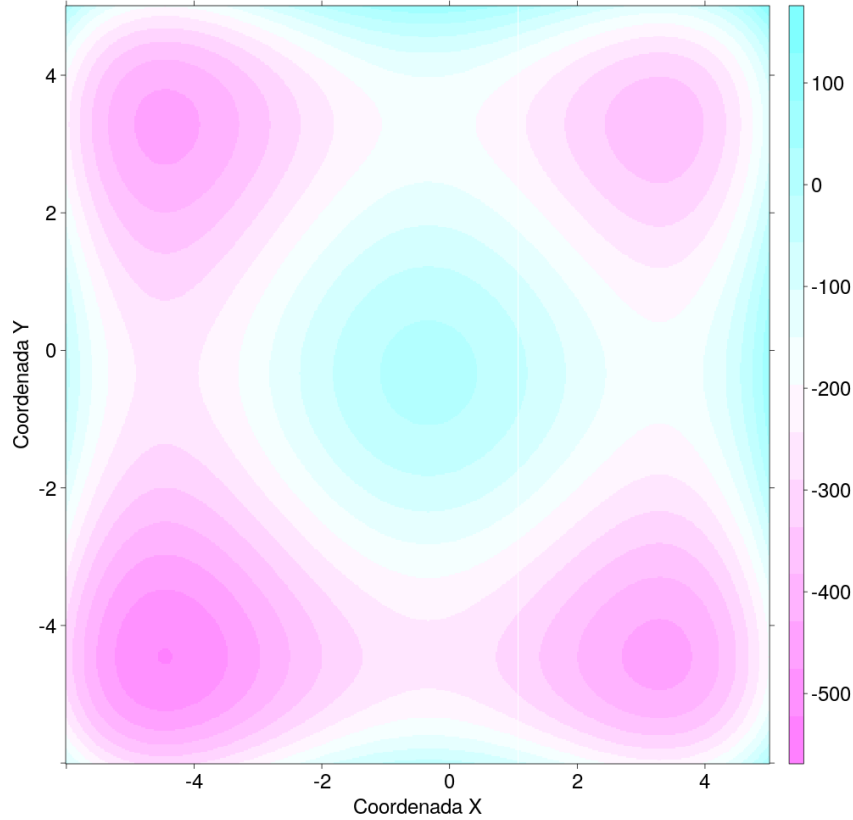


Figura 2: Gráfica de contorno de superficie de $g(x, y)$

Como podemos observar, el valor de $g(x, y)$ se aprecia a partir de la diferencia de color dentro de la gráfica, donde las regiones rosas con las zonas con menor valor de g y las azules con un mayor valor.

Para comenzar la experimentación, se utilizó una computadora con las siguientes especificaciones, Procesador Intel Core i7-4790 CPU @ 3.6GHz \times 8 y memoria RAM de 24 GB utilizando solamente los cuatro núcleos físicos disponibles. Se tomó el valor máximo obtenido de la página de Wolframalpha dentro el intervalo del dominio, el valor encontrado fue $g(x = 5, y = 5) \approx 130.12$, con esta información, se puede calcular la separación entre el valor encontrado con respecto al máximo a partir de la fórmula $sep = \frac{vr - vc}{vr} \times 100$.

Se utilizaron los siguientes parámetros para la experimentación, el multiarranque fue de 100 replicas, con una distancia para la elección de vecinos de 0.3 unidades, el tiempo máximo de cada búsqueda fue de 100 pasos, el código esta disponible en el repositorio git del curso.

1.2. Resultados

Al revisar los resultados de la experimentación, que están graficados en la figura 3, podemos apreciar algunas trayectorias en rojo, que representan caminatas que no alcanzaron el optimo global y nos permite ver la trayectoria que tomó la búsqueda, los puntos verdes son las coordenadas donde terminaron cada uno de las 100 caminatas y como podemos apreciar, convergieron en cada uno de los ocho óptimos locales disponibles en el dominio, en azul podemos ver la trayectoria que siguió búsqueda que alcanza el optimo global.

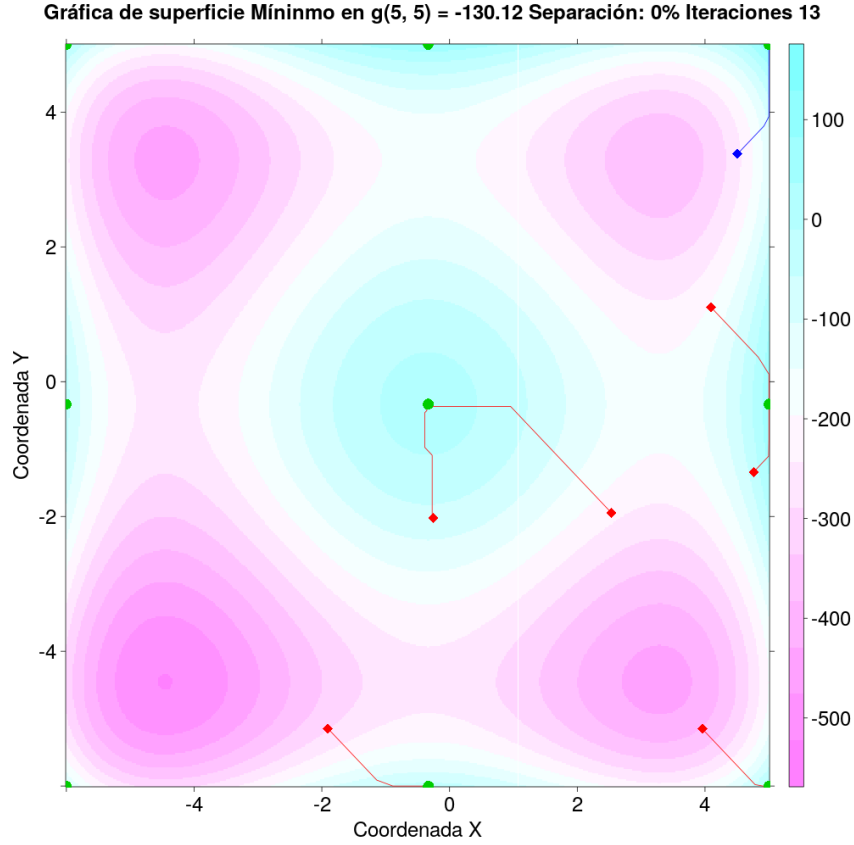


Figura 3: Resultado de trayectorias del método multiarranque

El valor de la separación fue cero, por lo que se encontró el optimo global, en el punto encontrado en wolframalpha, es importante apreciar que tardó solo 13 iteraciones para llegar al valor óptimo y que el lugar del inicio influye hacia que optimo local converge la búsqueda.

2. Extra Dos: Recocido Simulado

Para el extra dos de esta tarea se cambiara la estrategia para escapar de óptimos locales utilizando el método de recocido simulado. El cual esta basado en en el proceso de cristalización de metales calientes al enfriarse, donde las moléculas pasan de un estado de mucha energía a uno de menor energía.

Este método funciona con un valor de temperatura T que tiene un valor inicial $g(x, y)$, al realizar la búsqueda de un vecino, esta se hace de la siguiente manera, se elige una solución vecina al azar $g(x', y')$ y se revisa si esta solución ofrece una mejora comparada con el mejor valor obtenido hasta el momento, de ser así, se toma ese valor para continuar la búsqueda en sus vecinos, si el vecino elegido no es mejor, se calculara un delta $\delta = g(x, y) - g(x', y')$ tal que como este valor es negativo al no ofrecer mejora, se permitiría aceptar buscar dentro de los vecinos de ese punto si al obtener un valor aleatorio de una distribución uniforme de $[0, 1]$ este valor es mayor al calculado por la probabilidad obtenida por la distribución $e^{-\frac{\delta}{T}}$, al aceptar este valor que no mejora el valor de la búsqueda, hace que el valor de temperatura disminuya en proporción a un valor ε tal que $0 \leq \varepsilon < 1$ por lo que esta flexibilidad de aceptar valores no mejores es lo que nos permite escapar de óptimos locales dentro del dominio.

2.1. Diseño del Experimento

Los parámetros utilizados fueron los mismos que los del experimento anterior, la diferencia es en los parámetros del recocido simulado, se tomaron los valores de temperatura $T = 90$ y $\varepsilon = 0.9$.

2.2. Resultados

El resultado obtenido de la experimentación se puede ver en la figura 4 donde se puede apreciar como las trayectorias ya no van directamente hacia el óptimo local y explorar el espacio de soluciones, esto es notorio en el valor óptimo encontrado marcado con azul, ya que por la posición donde inicio, usando una búsqueda local como la realizada en el punto anterior, hubiera convergido hacia el optimo local cercano al punto (5, 0) pero debido a que pudo moverse por otras regiones es que pudo converger hacia el óptimo global.

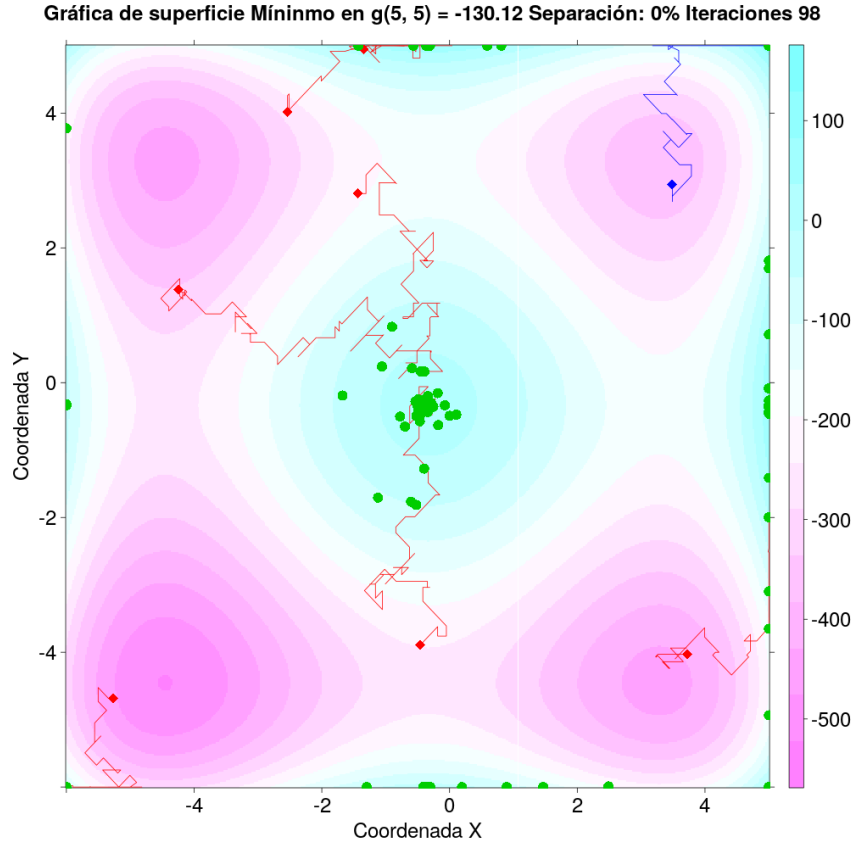


Figura 4: Resultado de trayectorias del método recocido simulado

Por otra parte es importante señalar que a diferencia del método multiarranque, las terminaciones de las búsquedas no convergían todas en alguno de los óptimos locales del dominio, sino que terminaban cerca de alguno de ellos, esto debido a que el valor de temperatura aún era alto al momento de terminar las iteraciones máximas permitidas, de tener una temperatura más baja, las búsquedas comenzarían a converger hacia los óptimos locales ya que comienzan a comportarse como búsquedas voraces, pero dejan de explorar el espacio de soluciones.

Además, la cantidad de iteraciones necesarias para encontrar el valor óptimo es superior a la requerida por el multiarranque, esto debido al valor de temperatura que al permitir valores no mejores, hace que la búsqueda demore más en converger hacia un óptimo local.

3. Conclusiones

Se pudo observar las diferencias entre dos métodos de búsqueda, además de paralelizar en ambas el múltiple inicio de las valores iniciales favoreciendo a la rapidez de los resultados, se pudieron observar los efectos del recocido simulado y conocer opciones de visualización para gráficas en tres dimensiones.