Tarea 7
Simulación de Sistemas

Beatriz Alejandra García Ramos

A 26 de Septiembre de 2017

1. Búsqueda local

En ésta práctica se trabaja con una optimización heurística para que, al tener una función bidimensional $\mathtt{g}(\mathtt{x}, \mathtt{y})$, se puedan encontrar máximos locales. En lugar de minimizar la función unidimensional que se proporciona en el código de la práctica se deben hacer modificaciones para obtener resultados al maximizar la función bidimensional.

2. Tareas

- Trabajar con la técnica que utiliza la función unidimensional modificando la manera en que se mueven aleatoriamente las variables `x` y `y`, seleccionar una de las cuatro combinaciones de los movimientos y encontrar el mayor valor para `g`.
- Lograr que de manera visual se comprenda lo que ocurre en la tarea base teniendo la visualización del proceso de búsqueda en una réplica sobre la gráfica de proyección plana.
- Modificar la manera en que se mueven aleatoriamente las variables de tal manera que se tenga un *recocido simulado* como se menciona en la práctica.

3. Solución

Dado que el código de la práctica maneja una función unidimensional y busca minimizar dicha función se deben hacer los cambios necesarios para que se tomen valores de la función bidimensional y lograr que se maximice ésta función. Al tomar un movimiento bidimensional sobre las variables de la función se deben tener al menos tres posiciones vecinas para poder compararlas y aprovechar la que sea mayor, en éste caso se tomaron cuatro puntos distintos vecinos al actual para calcular sus valores, localizar aquél cuyo valor fuera el más grande y fijarlo como el nuevo punto el cual ahora debe buscar dentro de sus vecinos el valor más grande y tomar a éste como el actual, y así sucesivamente hasta lograr terminar el proceso con una buena aproximación del valor máximo deseado.
Las modificaciones realizadas del código se pueden encontrar en los códigos de `R` en `GitHub`.

3.1. Tarea base

Para lograr hacer la tarea base se requirió modificar la función réplica, en la cual se tomarían dos mejores valores, uno para la variable `x` y otro para la variable `y`, para ello se tomaron distintos `delta` los cuales fueron de apoyo para hacer las sumas de $\mathtt{curr}[1] \pm \mathtt{delta.x}$ y de $\mathtt{curr}[2] \pm \mathtt{delta.y}$, éstos se modifican si están fuera de la frontera que se ha implementado en un principio. Una vez establecidos éstos valores se hacen las comparaciones al evaluarlos en la función $\mathtt{g}(\mathtt{x}, \mathtt{y})$ y dependiendo de qué valor es más grande es el que se toma como el mejor. Ya que se tienen el mejor `x` y el mejor `y` se evalúan éstos en la función de nuevo y se comparan, el que sea más grande es el que se toma como nuestra variable `curr`, además si se tiene que al evaluar la función como $\mathtt{g}(\mathtt{curr}[1], \mathtt{curr}[2])$ éste es mayor que

tener $\mathtt{g}(\mathtt{best}[1], \mathtt{best}[2])$ ahora el mejor valor será el que tome la variable `curr`.
Se tiene como en la práctica que se realicen las comparaciones para valores de cien, mil y diez mil pasos. Una vez teniendo los resultados de la función se toman como respuesta aquellos valores obtenidos de la evaluación de éstos resultados y se crea una matriz en donde se puedan adjuntar los tres valores: el valor de la variable `x`, el valor de la variable `y` y el valor de la función $\mathtt{g}(\mathtt{x}, \mathtt{y})$ evaluada con dichos valores de las variables.
Como respuesta a nuestra búsqueda se imprime el máximo valor de la función evaluada y es éste el que se toma como el mejor. En el cuadro 3.1 se encuentran algunos de los valores que arroja el proceso y todos tienen un valor similar, solamente variando en algunas décimas. Comparando con Wolfram Alpha, el cual nos da el valor de $\mathtt{g}(\mathtt{x}, \mathtt{y}) = 0{,}0666822$, podemos decir que tenemos un cálculo preciso.

Maximizando la función $g(x, y)$		
$g(x, y)$	x	y
0.06668213	-0.33275106	-0.33307142
0.06668215	-0.33288654	-0.33291561
0.06668215	-0.33314239	-0.33305094
0.06668215	-0.33292638	-0.33305072
0.06668216	-0.33296361	-0.33300552

Cuadro 1: Aproximaciones al valor máximo de la función $g(x, y)$

3.2. Reto 1

Para el primer reto se crearon imágenes que permiten comprender de manera visual lo que está sucediendo ya con las modificaciones hechas en la tarea anterior. Para ello se necesita la librería `lattice` que permite ver en dos dimensiones lo que sucede cuando se tiene un espacio de tres dimensiones, se presenta una imagen en donde se visualiza el espacio desde arriba y con colores se identifica dónde se encuentran las concavidades de la gráfica, así se tiene que el color lila representa los valores menores y el celeste los mayores. Una vez que se tiene éste espacio se van graficando aquellos puntos que resultan en las respuestas al evaluar las variables en la función, como se había mencionado anteriormente. Dentro del gráfico también se tiene aquél valor de Wolfram Alpha que indica el resultado del valor máximo de nuestra función marcado en color azul, ejemplo en la figura 3.2.
Cuando el proceso va avanzando la respuesta se va acercando al valor de Wolfram Alpha hasta llegar a un punto en el que se empalman una con otra. De ésta manera se puede comprobar que el proceso tiene una buena aproximación y se pueden observar los valores que toman las variables verificando los ejes del gráfico.
En `GitHub` hay tres animaciones que permiten ver el proceso que se lleva a cabo teniendo los diferentes tiempos máximos de la variable `tmax`.

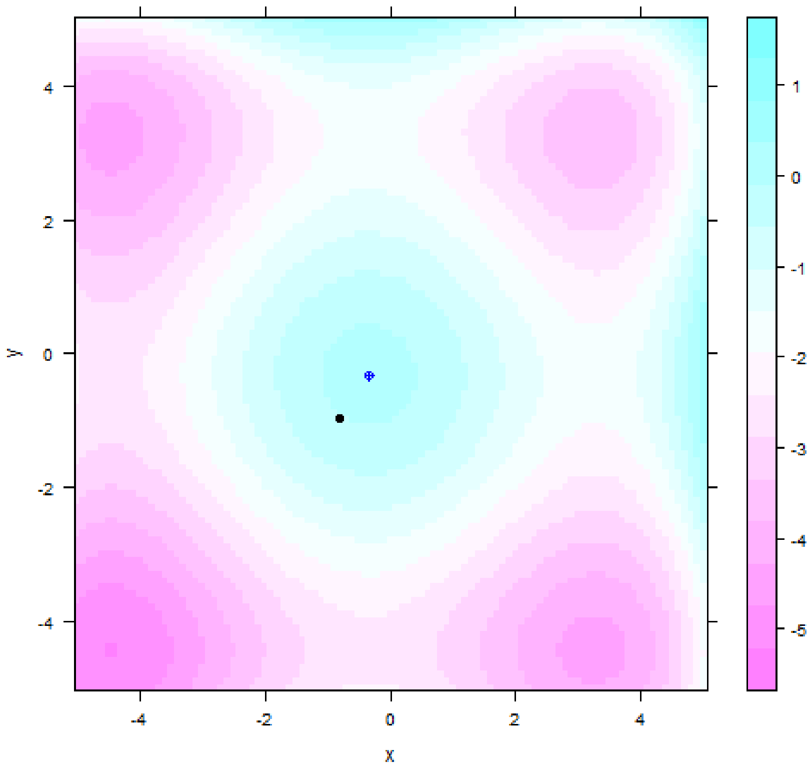

Figura 1: Verificando valores de la función $g(x, y)$ de forma gráfica

4. Conclusiones

Teniendo las comparaciones de los puntos con el proceso de verificación modificado para una función bidimensional podemos lograr obtener el valor máximo de nuestra función y de manera visual observar cómo avanza paso a paso el proceso conforme aumentan las veces en que se realiza éste. El proceso arroja un valor máximo muy preciso por lo que se puede concluir que es bueno.