Herramientas Computacionales

Taller 10 - Python: Método de Monte Carlo

Abril de 2015

La solución de este taller debe ser presentada en un solo archivo (Notebook de iPython) con nombre `NombreApellido_HW11.ipynb`. Puede trabajarse en grupos de dos estudiantes.

1. $\boxed{100\text{ pt}}$ En este taller vamos a simular el comportamiento del tráfico de carros en una calle circular de un solo carril y con una velocidad máxima $v_{max} = 5$, con un modelo llamado "Modelo de Nagel-Schreckenberg para el tráfico". En este modelo, la vía está dividida en 1000 espacios. En cada uno de estos espacios cabe exactamente un carro. Vamos a tener k carros en la vía, los cuales, inicialmente, van a estar distribuidos aleatoreamente sobre la misma. Cada carro comienza con una velocidad $v = 0$ y en cada paso de la simulación, los carros avanzan v espacios sobre la vía. Cada etapa de la simulación consiste en cuatro pasos que deben hacerse para todos los carros en paralelo. Esto quiere decir que no se debe mover ningún carro hasta que se halla hecho la simulación sobre todos. Los cuatro pasos son los siguientes: Primero, si la velocidad es menor a la veocidad máxima, se aumenta la velocidad en una unidad. Segundo, se mira la distancia con el carro del frente. Si dicha distancia es d y la velocidad del carro es mayor o igual que d, entonces se cambia la velocidad por $d-1$ para evitar colisiones. Tercero, si la velocidad es positiva, entonces, con una probabilidad $p = 1/3$, se reduce la velocidad en una unidad. En el cuarto paso todos los carros se mueven una distancia v. Así finaliza cada etapa de la simulación.

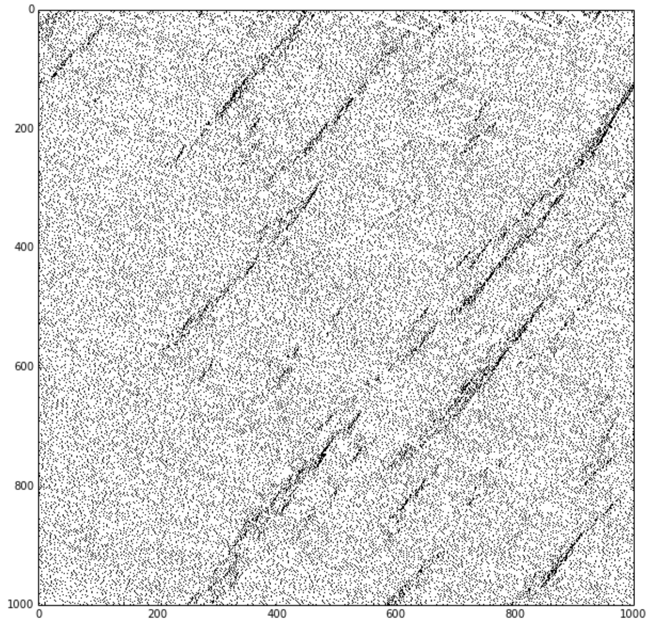
   Resumiendo:

   - $v \leftarrow min(v+1, v_{max})$
   - $v \leftarrow min(v, d-1)$
   - $v \leftarrow max(0, v-1)$ con probabilidad p.
   - $x \leftarrow x + v$

   Tenga en cuenta en el último paso que si $x + v \geq 1000$, entonces, $x \leftarrow x + v - 1000$. Igualmente, para el carro con la posición más grande x, el valor de d es $d = M + x_{min} - x_{max}$, donde x_{min} es la posición del carro con menor posición y x_{max} es la posición del carro con mayor posición.

   (a) $\boxed{70\text{ pt}}$ Haga una función que reciba como parámetro k el número de carros en la vía y devuelva una matriz de tamaño 1000×1000, en donde cada fila de la matriz sea un paso de la simulación. Es decir, la primera fila de la matriz debe ser el estado inicial de la via con k carros distribuidos aleatoreamente, y la segunda fila deben ser los carros movidos a nuevas posiciones de acuerdo con los cuatro pasos del modelo. Cada fila de la matriz debe estar compuesta por ceros y unos. Un 0 representa un espacio vacío y un 1 representa un espacio ocupado por un carro. Además la función debe retornar la distancia total recorrida por todos los carros al final de los 1000 pasos (observe que la distancia total recorrida por los carros en cada paso es la suma de las velocidades). Pista: además de los arreglos que representan a la vía en cada paso, haga arreglos para la velocidad de los carros y las posiciones sobre la misma.

   Una vez tenga la función, ejecútela con $k = 150$, la matriz obtenida visualícela con `imshow` y finalmente comente el resultado. Debe obtener algo así:



(b) $\boxed{\text{30 pt}}$ Ahora, repita la simulación para valores de $k = 50, 60, 70, 80, \ldots, 500$ y para cada una de estas simulaciones, guarde la distancia total recorrida por todos los carros. Haga una gráfica de la distancia total recorrida por los carros contra k. Aproximadamente para qué valor de k la vía es más eficiente. Comente los resultados.

Debe obtener una gráfica como la siguiente: