



UNIVERSIDAD DE GRANADA

SIMULACIÓN DE SISTEMAS
GRADO EN INGENIERÍA INFORMÁTICA

PRÁCTICA 1

DIFERENTES MODELOS DE SIMULACIÓN

Autor

Vladislav Nikolov Vasilev

Rama

Computación y Sistemas Inteligentes



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS INFORMÁTICA Y DE
TELECOMUNICACIÓN

CURSO 2018-2019

Índice general

1. Mi primer modelo de Montecarlo	2
1.1. Pruebas iniciales con el modelo	2
1.2. Experimentación con los parámetros	4
1.2.1. Modificación de la posición destino	4
1.2.2. Modificación de la probabilidad de ocupación	5
1.2.3. Modificación del nivel de visión	7
2. Mi primer modelo de simulación discreto	9

Capítulo 1

Mi primer modelo de Montecarlo

1.1. Pruebas iniciales con el modelo

Inicialmente, se ha probado el modelo para ver cómo funcionaba. Se ha ejecutado el modelo una serie de veces (10 para ser más precisos), y se han almacenado los resultados. Estos resultados pueden verse a continuación:

Mejor posición inicial (c)	Mejor distancia
95	6.525780
94	6.472190
94	6.513340
94	6.531020
94	6.498630
93	6.512340
94	6.469870
94	6.487920
94	6.486590
94	6.478660

Cuadro 1.1: Resultados de mejor posición inicial y mejor distancia para 10 ejecuciones.

Como se puede apreciar, parece que hay cierta semejanza entre los resultados obtenidos en cada ejecución. Se puede ver que el valor de c obtenido es muy similar en casi todos los casos, siendo la moda $c = 94$, y la media un valor muy próximo a éste. Los valores de la mejor distancia también son muy próximos entre sí, ya que todos ellos rondan, aproximadamente, las 6.5 plazas. Por tanto, el modelo es capaz de producir unos resultados muy similares a pesar de que utiliza cierta

“aleatoriedad”.

Si estudiamos cómo evoluciona el valor de la distancia al destino a medida que cambia el valor de c , nos encontramos con la siguiente gráfica:

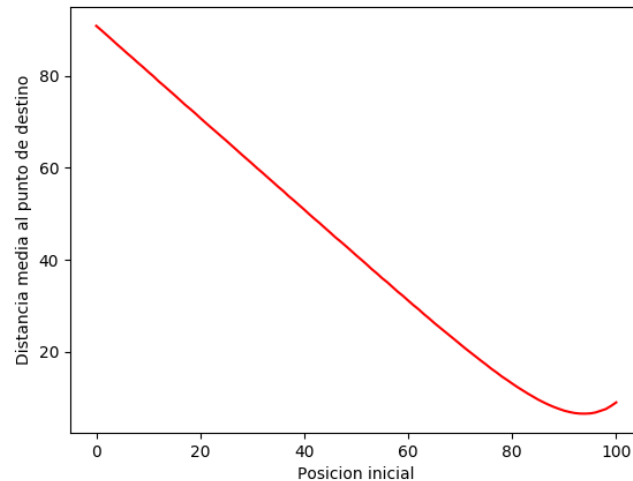


Figura 1.1: Evolución del valor de la distancia al destino en función del valor de la posición inicial.

Como se puede ver, existe una tendencia a que, cuanto más cerca de la posición a la que se quiera llegar se comienza a buscar sitio (es decir, cuanto más alto sea el valor de c), menor será la distancia hasta el destino. Esto es completamente lógico, ya que al comenzar a buscar sitio a partir de una posición muy lejana al destino, mayor será la distancia hasta éste en caso de que se encuentre una plaza libre. Teniendo en cuenta que se escoge la primera plaza libre, ésta puede quedar muy lejos del destino, que es lo que se puede apreciar en la gráfica.

El valor ideal de c , con las condiciones en las que estamos, parece ser $c = 94$. A partir de ahí, se puede ver que existe un ligero incremento en la distancia al destino, posiblemente porque se aparque más lejos debido a que no se encuentre una plaza libre en posiciones más cercanas al destino.

Por tanto, a vista de los resultados que hemos obtenido, podemos afirmar con bastante certeza que la plaza ideal a partir de la que empezar sitio para aparcar es aproximadamente la 94.

1.2. Experimentación con los parámetros

Se han realizado una serie de experimentaciones con los parámetros con los que se puede llamar al programa. A continuación, se muestran algunos de los resultados obtenidos.

1.2.1. Modificación de la posición destino

Se ha probado a modificar la posición destino, estableciendo que $x = 150$, y se han realizado 10 ejecuciones. A continuación, se pueden ver los resultados obtenidos:

Mejor posición inicial (c)	Mejor distancia
144	6.438900
144	6.438900
144	6.438900
143	6.472200
143	6.472200
143	6.472200
143	6.472200
143	6.472200
143	6.472200
143	6.472200
143	6.358800

Cuadro 1.2: Resultados de mejor posición inicial y mejor distancia para 10 ejecuciones con con posición destino $x = 150$.

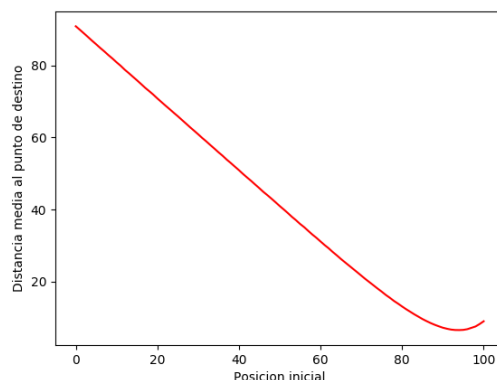


Figura 1.2: Evolución del valor de la distancia al destino en función del valor de la posición inicial para $x = 150$.

Los resultados obtenidos en la tabla 1.2 son muy parecidos a los que obtuvimos en 1.1, solo que los valores de c han cambiado, aunque siguen unos patrones parecidos. En este caso, la moda parece ser 143, y la media tiende a ese valor. Las mejores distancias están también muy próximas, y no se ve mucha disparidad.

Además, tal y como se hizo en el caso anterior, se ha obtenido una gráfica que muestra la evolución de la distancia media en función del valor de c , la cuál se puede ver en la figura 1.2. Como se puede observar, sigue un patrón muy parecido al que se puede ver en la figura 1.1, así que parece que no es un parámetro que influya demasiado por sí solo.

1.2.2. Modificación de la probabilidad de ocupación

Se ha probado a variar la probabilidad de ocupación para ver cómo es afectada la salida. A continuación, se pueden ver los resultados que se han obtenido:

Probabilidad de ocupación	Mejor posición inicial (c)	Mejor distancia
0.01	99	0.008400
0.1	99	0.100000
0.2	99	0.213100
0.3	99	0.344800
0.4	99	0.495100
0.5	98	0.744800
0.6	98	1.074900
0.7	98	1.659600
0.8	97	2.901000
0.9	94	6.477300
0.99	36	67.957802

Cuadro 1.3: Valores de la mejor posición inicial y distancia en función de la probabilidad de ocupación.

Es interesante ver cómo, a medida que va incrementando la probabilidad de ocupación, los valores de c y de distancia van cambiando, y se van haciendo cada vez peores. Se puede ver como el valor de c va disminuyendo a medida que va aumentando la probabilidad, lo cuál afecta directamente a la distancia, aumentándola proporcionalmente cada vez que disminuye el valor de c . Todo esto se debe a que, cuanto menor sea la probabilidad de ocupación, más cerca del destino se podrá encontrar un sitio. Todo esto se puede ver gráficamente en la siguiente figura:

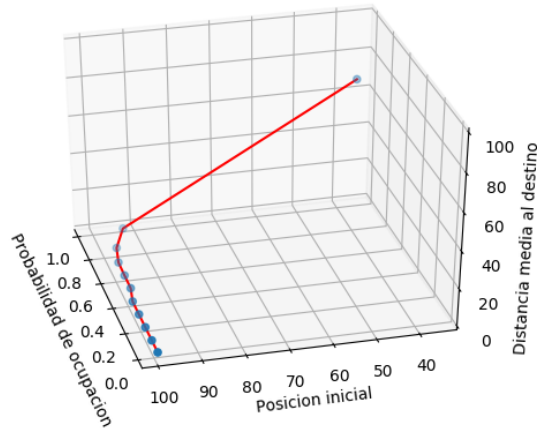


Figura 1.3: Representación 3D de la distancia media al destino en función de la probabilidad de ocupación y la posición inicial.

Si se prueba con valores muy extremos, como por ejemplo con probabilidad de ocupación de 0.999, se obtiene el siguiente error:

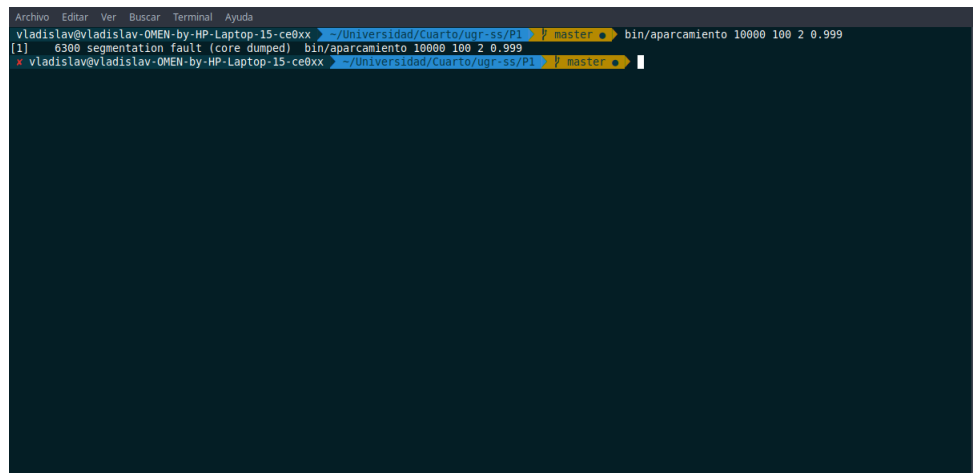


Figura 1.4: Error al ejecutar el programa *aparcamiento* con 0.999 de probabilidad de ocupación.

Esto se debe a que no se consigue encontrar sitio libre y se supera el tamaño del vector que representa las posiciones, lo cuál genera un fallo de segmentación al intentar acceder a posiciones no válidas de memoria.

1.2.3. Modificación del nivel de visión

Vamos a intentar variar ahora el nivel de visión para ver cómo son afectados los resultados. Estos son los resultados obtenidos:

Alcance de visión	Mejor posición inicial (c)	Mejor distancia
2	94	6.480800
4	94	6.339700
6	94	6.069100
8	93	5.706100
10	92	5.505300
15	90	5.244000
20	86	4.954400
25	83	4.890900
30	82	4.778700
50	72	4.687100
100	80	4.624000

Cuadro 1.4: Valores de la mejor posición inicial y distancia en función del alcance de visión.

Como se puede ver, aquí también al aumentar el valor del alcance de visión, el valor de c disminuye. Sin embargo, a diferencia del caso anterior, la mejor distancia media va disminuyendo, posiblemente debido a que se tenga más conocimiento de las posiciones venideras. En el caso extremo en el que se ven todas las plazas de aparcamiento, se ve que la distancia media es la mínima. Todo esto se puede ver en la siguiente figura:

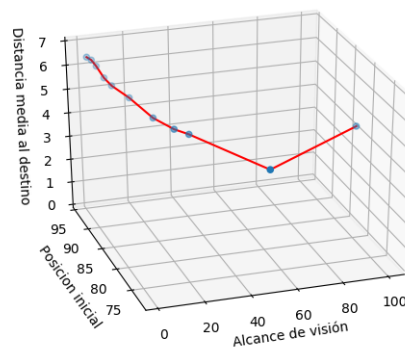


Figura 1.5: Representación 3D de la distancia media al destino en función del alcance de visión y la posición inicial.

Capítulo 2

Mi primer modelo de simulación discreto