



# UNIVERSIDAD DE GRANADA

SIMULACIÓN DE SISTEMAS  
GRADO EN INGENIERÍA INFORMÁTICA

---

## PRÁCTICA 2

MODELOS DE MONTE CARLO. GENERADORES DE DATOS

---

### Autor

Vladislav Nikolov Vasilev

### Rama

Computación y Sistemas Inteligentes



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍAS INFORMÁTICA Y DE  
TELECOMUNICACIÓN

CURSO 2019-2020

# Índice general

<b>1. MI SEGUNDO MODELO DE SIMULACIÓN DE MONTE CARLO</b>	<b>2</b>
1.1. Modelización por Monte Carlo . . . . .	2
1.2. Modificaciones del Modelo . . . . .	6
<b>2. GENERADORES DE DATOS</b>	<b>7</b>
2.1. Mejorando los Generadores . . . . .	7
2.1.1. Reordenación de los valores . . . . .	7
2.1.2. Mejorando la búsqueda: búsqueda binaria . . . . .	7
2.1.3. Tiempo de acceso constante a la tabla . . . . .	7
2.2. Generadores congruenciales lineales . . . . .	7
<b>Bibliografía</b>	<b>8</b>

# Capítulo 1

## MI SEGUNDO MODELO DE SIMULACIÓN DE MONTE CARLO

### 1.1. Modelización por Monte Carlo

Una vez que hemos creado nuestro modelo de Monte Carlo, vamos a ver qué resultados obtenemos, y si éstos son buenos o pueden mejorar. Para poder contrastar, solo podremos utilizar una de las expresiones que encontramos en el guión proporcionado. La correctitud del resto será confirmada viendo si los resultados obtenidos son parecidos o no a medida que se van aumentando el número de simulaciones.

Para la experimentación, vamos a probar con distintos valores de  $x$ ,  $y$  y número de simulaciones. Vamos a probar con las siguientes combinaciones de ganancias y pérdidas:

- Con  $x = 10$  e  $y = 1$ .
- Con  $x = 10$  e  $y = 5$ .
- Con  $x = 10$  e  $y = 10$ .
- Con  $x = 15$  e  $y = 10$ .

Cada combinación se va a simular 100, 1000, 10000 y 100000 veces para ver cómo van evolucionando los resultados, si éstos se van estabilizando o si van variando mucho. También se va a probar para cada generador, para ver los resultados que ofrecen.

Ganancia por unidad vendida (x)	Pérdida por unidad no vendida (y)	Número de repeticiones	Mejor número de unidades pedidas (s)	Mejor ganancia media	Tiempo (seg)
10	1	100	85	488.76	0.007848
10	1	1000	96	460.116	0.041666
10	1	10000	89	453.7774	0.161990
10	1	100000	86	451.0508	1.363344
10	5	100	79	391.15	0.004215
10	5	1000	67	345.19	0.016622
10	5	10000	71	331.6865	0.170900
10	5	100000	67	329.49985	1.358017
10	10	100	48	309	0.002749
10	10	1000	57	270.66	0.016464
10	10	10000	50	253.072	0.136419
10	10	100000	48	246.02	1.289375
15	10	100	72	528	0.004562
15	10	1000	59	461.9	0.016942
15	10	10000	58	456.05	0.137328
15	10	100000	60	444.4585	1.320523

Cuadro 1.1: Resultados obtenidos por el modelo utilizando el generador de distribución uniforme.

Para contrastar los datos podemos utilizar, tal y como hemos dicho antes, la expresión analítica que aparece en el guión:

- Para el caso de  $x = 10$  e  $y = 1$ , obtenemos que  $s^* = 90$ .
- Para el caso de  $x = 10$  e  $y = 5$ , obtenemos que  $s^* = 66$ .
- Para el caso de  $x = 10$  e  $y = 10$ , obtenemos que  $s^* = 49$ .
- Para el caso de  $x = 15$  e  $y = 10$ , obtenemos que  $s^* = 59$ .

De los resultados obtenidos, podemos observar que los valores obtenidos, a medida que se van incrementando el número de repeticiones, se van acercando más y más a los valores óptimos reales. Esto es normal, ya que, aunque haya números aleatorios envueltos en el proceso, al hacer muchas repeticiones, de media, el resultado se aproximará al valor óptimo, siempre con un cierto margen de error. Con pocas repeticiones esto es difícil que pase, ya que no se da mucho margen para obtener un promedio decente, ya que la aleatoriedad puede hacer que salgan muchos valores en los extremos. Por tanto, intentar extraer conclusiones con un número tan pequeño de repeticiones sería contraproducente y no reflejaría muy bien la realidad.

En general, podemos afirmar que los resultados obtenidos son bastante precisos, siempre y cuando hagamos un número razonable de repeticiones. Por tanto, el modelo parece estar funcionando bien.

A la vista de lo que hemos obtenido, podemos decir que, si mantenemos el valor de  $x$  constante y subimos el de  $y$ , el número de unidades pedidas media va disminuyendo. Esto puede deberse a que no es viable tener un mayor número si las pérdidas son más grandes.

Una vez vistos los resultados para el modelo que usa la distribución uniforme, vamos a ver qué obtenemos para el resto. Vamos a seguir con la distribución proporcional. Los resultados obtenidos se pueden ver en la siguiente tabla:

Ganancia por unidad vendida (x)	Pérdida por unidad no vendida (y)	Número de repeticiones	Mejor número de unidades pedidas (s)	Mejor ganancia media	Tiempo (seg)
10	1	100	63	337.51	0.001631
10	1	1000	76	298.187	0.034052
10	1	10000	71	287.7859	0.129329
10	1	100000	73	284.1546	1.019929
10	5	100	56	218	0.005015
10	5	1000	42	197.925	0.014599
10	5	10000	39	190.266	0.105564
10	5	100000	40	188.95885	1.065668
10	10	100	40	169.6	0.005048
10	10	1000	31	140.26	0.025562
10	10	10000	29	137.962	0.130958
10	10	100000	30	134.8548	1.031648
15	10	100	44	329.75	0.004820
15	10	1000	31	255.775	0.035382
15	10	10000	35	252.32	0.104106
15	10	100000	37	250.91575	1.028880

Cuadro 1.2: Resultados obtenidos por el modelo utilizando el generador de distribución proporcional.

Como no tenemos una expresión analítica con la que comparar, vamos a fijarnos en cómo van evolucionando los valores de  $s$ .

Tal y como pasaba en el ejemplo anterior, a medida que vamos aumentando el número de repeticiones, más se aproximan los valores obtenidos a los óptimos. Podemos ver que con unas pocas repeticiones (unas 100) los resultados se quedan bastante lejos de los que se obtienen con un mayor número de repeticiones, tal y como pasaba antes. Por tanto, si quisiéramos sacar unas conclusiones sólidas, tendríamos que fijarnos en los resultados con un número grande de repeticiones, como por ejemplo 10000 o 100000. Podemos decir que el número óptimo estará próximo a los valores reflejados en la tabla para ese número de repeticiones.

Los resultados son obviamente distintos a los que podemos observar en la tabla 1.1. Esto es así porque la distribución utilizada es diferente. El efecto que causa usar una distribución diferente a la anterior es que los valores medios de  $s$  se ven reducidos. Esto se debe principalmente a la forma que tiene la distribución propor-

cional, ya que es decreciente, y por tanto, las demandas más pequeñas tienen una mayor probabilidad que las grandes. Sin embargo, las dos tablas tienen dos cosas en común. La primera son los tiempos, ya que no hay mucha diferencia notable. Este resultado no debe sorprender a nadie, ya que las tablas se han generado antes, y lo único que se está haciendo es recuperar los valores. Y la segunda es la relación entre el valor de  $x$  y el de  $y$ , la cuál ya se comentó anteriormente.

Visto este generador, vamos a ver qué resultados nos permite obtener el último, el de la distribución “triangular”. A continuación se pueden ver los resultados:

Ganancia por unidad vendida ( $x$ )	Pérdida por unidad no vendida ( $y$ )	Número de repeticiones	Mejor número de unidades pedidas ( $s$ )	Mejor ganancia media	Tiempo (seg)
10	1	100	77	505.45	0.004564
10	1	1000	84	472.743	0.041040
10	1	10000	82	466.5304	0.141798
10	1	100000	82	464.88623	1.392658
10	5	100	64	420.25	0.006475
10	5	1000	55	393.64	0.017583
10	5	10000	65	387.3845	0.167988
10	5	100000	61	387.02035	1.394215
10	10	100	49	366.6	0.002019
10	10	1000	51	351.26	0.043020
10	10	10000	52	336.416	0.142922
10	10	100000	51	334.3226	1.428285
15	10	100	61	627	0.002402
15	10	1000	56	565.075	0.042256
15	10	10000	55	550.065	0.149202
15	10	100000	55	549.21975	1.365606

Cuadro 1.3: Resultados obtenidos por el modelo utilizando el generador de distribución “triangular”.

De nuevo, para extraer unas conclusiones sólidas, vamos a fijarnos en los valores medios obtenidos con un número de repeticiones más alto, por los motivos comentados anteriormente.

En general, podemos ver que los valores de  $s$  obtenidos son más “céntricos”, debido a la forma de la distribución, ya que esta tiene más forma de triángulo, y por tanto, los valores más probables estarán en el centro. De nuevo, tal y como pudimos observar en las tablas 1.1 y 1.2, podemos ver claramente la relación entre los valores de  $x$  y de  $y$ . Y de nuevo, tal y como pasaba antes, no hay mucha diferencia en los tiempos de ejecución.

Observando los valores de los resultados, podemos ver que en este caso son mucho más próximos que en los anteriores. Incluso con muy pocas repeticiones, los valores de  $s$  obtenidos no distan tanto de aquellos obtenidos con un mayor número de repeticiones, cosa que sí que sucedía, sobre todo en la tabla 1.1.

Por tanto, para concluir este apartado, podemos decir que es importante ejecutar un modelo de Monte Carlo repitiendo la simulación muchas veces, para así obtener unos resultados más fiables. Los valores obtenidos para cada generador han sido diferentes, como era de esperar, ya que las distribuciones han sido diferentes. Por tanto, esto nos indica que, a la hora de diseñar un buen modelo, debemos tener cierta información de como son las distribuciones reales, para así poder obtener unos resultados más representativos. Sin embargo, el hecho de poder probar distintas distribuciones para ver como varían los resultados ofrece a los modelos de Monte Carlo mucha flexibilidad y potencia, ya que pueden ser adaptados a las necesidades específicas del usuario.

## 1.2. Modificaciones del Modelo

## Capítulo 2

# GENERADORES DE DATOS

### 2.1. Mejorando los Generadores

#### 2.1.1. Reordenación de los valores

#### 2.1.2. Mejorando la búsqueda: búsqueda binaria

#### 2.1.3. Tiempo de acceso constante a la tabla

### 2.2. Generadores congruenciales lineales



# Bibliografía

- [1] Texto referencia  
<https://url.referencia.com>