

Maestría en Cómputo Estadístico - CIMAT

Proyecto final

Programación y Análisis de Algoritmos

Victor Manuel Gómez Espinosa
10-12-2019

Contenido

- 1. Resumen2
- 2. Introducción.....2
- 3. Comparación.....2
- 4. Simulación de ejemplos de libro.....3
- 5 Conclusiones.....5
- 6 Referencias6

1. Resumen

Se realizó un programa en C++ y en R que simula un sistema de cola de tamaño finito y un servidor, repitiendo el experimento una determinada cantidad de veces, comparando los tiempos de ambos y simulando ejemplos de libro.

2. Introducción

El tipo de sistema simulado se considera de ventas perdidas, esto sucede cuando ya no ocurren entradas al sistema debido a que el sistema se encuentra saturado ya que es de tamaño finito, en este caso particular solo se cuenta con 1 servidor y un tamaño de fila de espera determinado. Este tipo de sistema lo que busca es encontrar el tamaño de fila optimo a fin de minimizar las perdidas debido a que el sistema se encuentre saturado. Ejemplo de este tipo de sistema puede ser un restaurante donde el numero de mesas es el tamaño del sistema, cuando ya no hay mesas disponibles y llega un nuevo cliente, al estar lleno, no espera y se va lo cual representa una perdida para el restaurante.

3. Comparación

Para la simulación de este tipo de sistema es necesario hacerlo por un determinado tiempo (días, semanas, etc.), lo cual representa un ciclo, y a su vez se necesita repetir varias veces el mismo experimento (otro ciclo), esto es ineficiente en R (`cola_finitaR`), por lo cual se escribió el mismo código en C++ (`cola_finitaC`) y se conectó con R, para comparar la velocidad de ejecución contra el mismo algoritmo en R. los resultados fueron los siguientes:

```
> #compara el rendimiento del codigo en C++ contra R
> microbenchmark(
+   cola_finitaR(12,15,2,1,40,1,1e3),
+   cola_finitaC(12,15,2,1,40,1,1e3)
+ )
unit: milliseconds
```

	expr	min	lq	mean	median	uq	max	neval
cola_finitaR(12, 15, 2, 1, 40, 1, 1000)		1170.6514	1191.21215	1218.02236	1207.68750	1241.87670	1327.5841	100
cola_finitaC(12, 15, 2, 1, 40, 1, 1000)		26.2503	26.50145	26.98933	26.71625	27.06895	30.0002	100

Ilustración 1 R vs C++

Se puede observar claramente que el código en C++ es mucho más rápido.

4. Simulación de ejemplos de libro

Para probar el código se simularon 2 ejemplos de libro (Thomopoulos 2012), y se compararon resultados.

4.1 Ejemplo 1

Se simula una barbería con un barbero (1 servidor) y 2 sillas de espera disponibles (tamaño de fila 2), por lo tanto, tamaño de sistema 3, con una tasa promedio de llegadas de 15 min, tasa promedio de servicio de 12 min (tiempos exponenciales), en el cual la barbería está abierta 8hrs al día, por 5 días a la semana, dando un tiempo de simulación total de 40 horas, y repitiendo el experimento anterior 10,000 veces, para obtener la probabilidad de perder clientes (ploss o P_3), los clientes perdidos esperados por semana, clientes esperados por semana, ganancias esperadas por semana y pérdidas esperadas por semana. Los resultados se muestran en las dos imágenes siguientes:

Expected customers per week ($40\lambda_e$)	132
Expected customers lost per week [$40(\lambda-\lambda_e)$]	28
Expected weekly fees (132×12)	\$1584
Expected weekly fees lost (28×12)	\$336
Note: $P_0 = 0.339$	
$P_1 = 0.271$	
$P_2 = 0.217$	
$P_3 = 0.173$	

Ilustración 2 Resultados de libro ejemplo 1

```
> cat("Probabilidad de perder clientes :",round(p_cperdidos,2),"\\n" )
Probabilidad de perder clientes : 0.16
> cat("Clientes perdidos esperados por semana :",round(c_perdidos,0),"\\n" )
Clientes perdidos esperados por semana : 25
> cat("Clientes esperados por semana :",round(c_esperados,0),"\\n" )
Clientes esperados por semana : 129
> cat("Ganancias esperadas por semana :",round(ganancias,0),"\\n" )
Ganancias esperadas por semana : 1544
> cat("Pérdidas esperadas por semana :",round(perdidas,0),"\\n" )
Pérdidas esperadas por semana : 296
```

Ilustración 3 Resultados de simulación ejemplo 1

4.1 Ejemplo 2

Este ejemplo simula la probabilidad de perdida (Ploss) con los mismos datos del ejemplo anterior, variando el tamaño del sistema (N) de 1 a 10, y variando la tasa promedio de servicio de 1.5 a 13.5 y manteniendo fija la tasa promedio de llegadas (ρ). En las siguientes 2 ilustraciones se muestran los resultados del libro y los de la simulación y adicionalmente se realiza un gráfico de con los resultados de la tabla para visualizarlos mejor:

The table below lists values of Ploss, for $\rho = 0.1-0.9$ and $N = 1-10$, when one service facility and all times are exponential. Blanks are the same as 0.00.

Ploss

ρ/N	1	2	3	4	5	6	7	8	9	10
0.1	0.09	0.01	0.00							
0.2	0.17	0.03	0.01	0.00						
0.3	0.23	0.06	0.02	0.01	0.00					
0.4	0.29	0.10	0.04	0.02	0.01	0.00				
0.5	0.33	0.14	0.07	0.03	0.02	0.01	0.00			
0.6	0.38	0.18	0.10	0.06	0.03	0.02	0.01	0.01	0.00	
0.7	0.41	0.22	0.14	0.09	0.06	0.04	0.03	0.02	0.01	0.01
0.8	0.44	0.26	0.17	0.12	0.09	0.07	0.05	0.04	0.03	0.02
0.9	0.47	0.30	0.21	0.16	0.13	0.10	0.08	0.07	0.06	0.05

Note when $\rho = 0.8$ and $N = 3$. Ploss = 0.17.

Ilustración 4 Resultados de libro ejemplo 2

ρ/N	0	1	2	3	4	5	6	7	8	9
0.1	0.09	0	0	0	0	0	0	0	0	0
0.2	0.18	0.03	0	0	0	0	0	0	0	0
0.3	0.23	0.05	0.01	0	0	0	0	0	0	0
0.4	0.29	0.1	0.03	0.01	0	0	0	0	0	0
0.5	0.33	0.12	0.05	0.02	0.01	0	0	0	0	0
0.6	0.38	0.18	0.09	0.05	0.03	0.01	0.01	0	0	0
0.7	0.4	0.2	0.11	0.07	0.04	0.03	0.02	0.01	0.01	0
0.8	0.44	0.25	0.16	0.11	0.08	0.05	0.04	0.03	0.02	0.02
0.9	0.46	0.27	0.18	0.13	0.1	0.08	0.06	0.05	0.04	0.03

Ilustración 5 Resultados simulación ejemplo 2

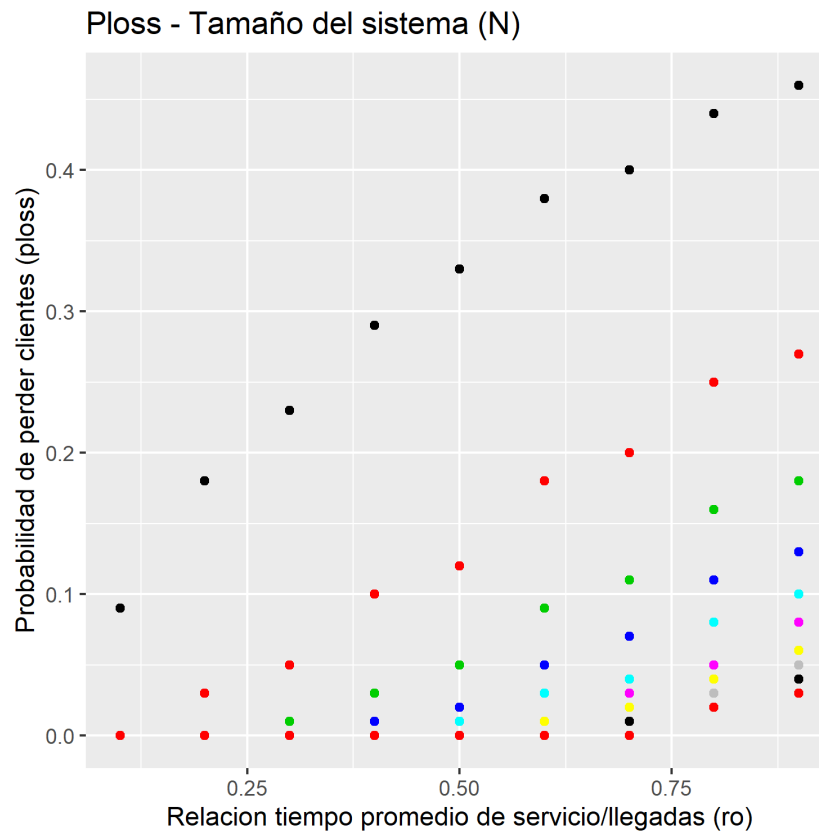


Ilustración 6 Resultados simulación

En la ilustración se observa que mientras mayor sea la tasa promedio de servicio respecto a la de llegadas, la probabilidad de que este lleno y por lo tanto de perder clientes aumenta, así como también mientras mas pequeño sea el sistema o en este caso asientos disponibles en la barbería (color negro).

5 Conclusiones

- 1) Escribir un código en C++ y conectarlo con R, es muy útil y eficiente a la hora de simular muchas veces un experimento o algo que involucre ciclos muy grandes, lo hace mucho más rápido.
- 2) Se observó que los resultados de las simulaciones contra los resultados de los ejemplos de libro (ilustraciones 2-5), no son idénticos, pero se aproximan muy bien, esto debido a que en el libro los resultados son calculados de forma analítica y los aquí obtenidos fueron simulados.

6 Referencias

Thomopoulos, N. T. (2012), *Fundamentals of queuing systems: statistical methods for analyzing queuing models*, Springer.