TP 3 - Simulación de un modelo MM1 e Inventario

Berruti, Octavio
UTN, FRRO
Zeballos 1341
octavioberruti8@gmail.com

Villegas, Franco
UTN, FRRO
Zeballos 1341
villegasfranco1098@gmail.com

Guerrini, Lucas
UTN, FRRO
Zeballos 1341
lucasguerrinii@gmail.com

Romaniuk, Federico
UTN, FRRO
Zeballos 1341
federico.roma98@gmail.com

June 5, 2021

UTN: Ingeniería en sistemas de Información
Simulación | Comisión 401

Abstract

Desarrollar un algoritmo con el software Phyton y AnyLogic que simule tanto un sistema de colas del tipo M/M/1, como un modelo de inventario . Realizando un posterior analisis y comparacion

1 Introducción

En este trabajo practico se estudiará, por un lado la teoria de colas (MM1). En muchas ocasiones de la vida real, un fenómeno muy común es la formación de colas o líneas de espera. Esto suele ocurrir cuando la demanda real de un servicio es superior a la capacidad que existe para dar dicho servicio. El estudio de las colas nos sirve para proporcionar tanto una base teórica del tipo de servicio que podemos esperar de un determinado recurso, como la forma en la cual dicho recurso puede ser diseñado para proporcionar un determinado grado de servicio a sus clientes.

Por otro lado se hará estudio tambien del modelo de inventario. Las empresas mantienen inventarios de materias primas y de productos terminados. Puesto que estos representan frecuentemente una considerable inversión, las decisiones con respecto a las cantidades de inventario son importantes. Los modelos y la descripción matemática de los sistemas de inventario constituyen una base para estas decisiones

2 Enunciado

El trabajo consiste en realizar un estudio de simulación de un modelo MM1 y de Inventario.

En donde para MM1 se analice:

- Promedio de clientes en el sistema
- Promedio de clientes en cola.
- Tiempo promedio en sistema.
- Tiempo promedio en cola.
- Utilización del servidor.
- Probabilidad de n clientes en cola.
- Probabilidad de denegación de servicio.

En cuanto al modelo de inventario:

- Costo de orden
- Coste de mantenimiento
- Costo de faltante
- Costo total (suma de los tres anteriores)

3 Teoria de colas

3.1 Caracteristicas

La teoría de colas es el estudio matemático de las colas o líneas de espera dentro de un sistema, se pueden describir las siguientes caracteristicas , las cuales forman parte de la misma como conjunto:

3.1.1 Población de cliente

La poblacion de clientes puede ser infinita o finita.El analisis para una poblacion finita es mas complicado que el analisis en donde la base de población se considera infinita.

3.1.2 El proceso de llegada

El proceso de llegada es la forma en que los clientes llegan a solicitar un servicio. La característica más importante del proceso de llegada es el tiempo entre llegadas, que es la cantidad de tiempo entre dos llegadas sucesivas.

Proceso de colas Parte del proceso de colas tienen que ver con la forma en que los clientes esperan para ser atendidos. Los clientes puede esperar en una sola fila o en un sistema de colas de lineas multiples. Otra caracteristica es el número de espacios de espera en cada fila

3.1.3 Disciplina de colas

Las disiplina de colas son : FIFO (First-In-First-Out), LIFO (Last-In-First-Out), SIRO (Service-In-Random-Order), PS(Processsor haring), PR (Priority)

3.1.4 Proceso de servicios

El proceso de servicios define como son atendidos los clientes. En algunos casos, puede existir mas de una estacion en el sistema en la cual se proporcione el servicio requerido. A tales estructuras se les conoce como sistema de colas de canal multiple. En estos sistemas los servicios pueden ser identicos o pueden ser no identicos.

Otras caracteristicas del proceso de servicios es el número de clientes atendidos al mismo tiempo es una estación. y si más de un proceso de servicio es si se permite o no la prioridad.

Cualquiera que sea el proceso de servicio, es necesario tener una idea de cuánto tiempo se requiere para llevar a cabo el servicio. Esta cantidad es importante debido a que cuanto mas dure el servicio, mas tendrán que esperar los clientes que llegan.

Este proceso puede ser deterministico o probabilistico. Con un tiempo de servicio probabilístico, cada cliente requiere una cantidad distinta e incierta de tiempo de servicio.

Los tiempos de servicio probabilisticos se describen matematicamente mediante una distribución exponencial.

3.2 Descripción de un sistema de colas

Un sistema de colas se puede describir como sigue. Un conjunto de “clientes” llega a un sistema buscando un servicio, esperan si este no es inmediato, y abandonan el sistema una vez han sido atendidos. En algunos casos se puede admitir que los clientes abandonan el sistema si se cansan de esperar. El término “cliente” se usa con un sentido general y no implica que sea un ser humano, puede significar piezas esperando su turno para ser procesadas o una lista de trabajo esperando para imprimir en una impresora en red.

Aunque la mayor parte de los sistemas se puedan representar como en la figura 1, debe quedar claro que una representación detallada exige definir un número elevado de parámetros y funciones. La teoría de colas fue originariamente un trabajo práctico. La primera aplicación de la que se tiene noticia es del matemático danés Erlang sobre conversaciones telefónicas en 1909, para el cálculo de tamaño de centralitas. Después se convirtió en un concepto teórico que consiguió un gran desarrollo, y desde hace unos años se vuelve a hablar de un concepto aplicado aunque exige un importante trabajo de análisis para convertir las fórmulas en realidades, o viceversa

3.3 Número de canales del servicio

En esta fase es importante conocer o identificar cuántos servidores están disponibles para atender los clientes que llegan al sistema. De esta manera se pueden presentar diferentes estructuras de sistemas de colas. Se habla generalmente de una cola que alimenta a varios servidores mientras que el caso de colas independientes se asemeja a múltiples sistemas con sólo un servidor.

Es evidente que es preferible utilizar sistemas multiservidos con una única línea de espera para todos que con una cola por servidor. Por tanto, cuando se habla de canales de servicio paralelos, se habla generalmente de una cola que alimenta a varios servidores mientras que el caso de colas independientes se asemeja a múltiples sistemas con sólo un servidor.

4 Modelo de un solo canal (M / M / 1)

A continuación se presenta el enfoque de análisis que se debe dar al sistema de línea de espera típico con llegadas de tipo Poisson, tiempos de servicio de tipo Exponencial con un sólo servidor.Se supone que en este sistema, la entidad está dispuesta a esperar el tiempo que sea para ser atendido, es decir no hay rechazo.

Donde:

- λ : Número promedio de llegadas al sistema - unidad de tiempo (velocidad de llegadas)
- μ: Número promedio de entidades que se atienden en el sistema - unidad de tiempo. (velocidad de atención del servidor)

Y las ecuaciones serán de la siguiente manera:

- L_s : Número promedio de unidades en el sistema

$$L_s = \frac{\lambda}{\mu - \lambda} \tag{1}$$

- W_s: Tiempo promedio en que una unidad esta dentro del sistema

$$W_s = \frac{1}{\mu - \lambda} \tag{2}$$

- L_q : Número promedio de unidades en la fila de espera

$$L_q = \frac{\lambda^2}{\mu(\mu - \lambda)} \tag{3}$$

- W_q : Tiempo promedio en que una unidad pasa por la fila de espera

$$W_q = \frac{\lambda}{\mu(\mu - \lambda)} \tag{4}$$

- ρ : Factor de uso del sistema o del servidor

$$\rho = \frac{\lambda}{\mu} \tag{5}$$

- P_o : Probabilidad de que ninguna unidad se encuentre en el sistema

$$P_o = 1 - \rho \tag{6}$$

$$P_o = 1 - \frac{\lambda}{\mu} \tag{7}$$

- P_n : Probabilidad de que el sistema tenga exactamente "n" unidades

$$P_n = (1 - \frac{\lambda}{\mu})(\frac{\lambda}{\mu})^n \tag{8}$$

5 Desarrollo

Para realizar este experimento, como mencionamos al principio de este trabajo, vamos a simular una cola MM1. Calcularemos los valores teóricos esperados y los compararemos con los valores observados de la simulación en Python y en Anylogic. Para ello tenemos 5 casos de estudio con diferentes tasas de arribo respecto de la tasa de servicio.

- Caso de estudio 1: cuando la tasa de arribo es un 25% de la tasa de servicio.
- Caso de estudio 2: cuando la tasa de arribo es un 50% de la tasa de servicio.
- Caso de estudio 3: cuando la tasa de arribo es un 75% de la tasa de servicio.
- Caso de estudio 4: cuando la tasa de arribo es un 100% de la tasa de servicio.
- Caso de estudio 5: cuando la tasa de arribo es un 125% de la tasa de servicio.

5.1 Componentes de la simulación

En esta sección se detallaran los componentes que se encuentran en un nuestra simulación de eventos discretos sin explicitar en explicaciones de código o algoritmos. Siguiendo los lineamientos del libro SIMULATION MODELING ANALYSIS By $Kelton$ and Law estos son los diferentes componentes que programamos para llevar a cabo la simulación en Python

- Estado del sistema: Es la colección de las variables de estado necesarias para describir el sistema en un tiempo especifico
- Reloj de simulación: variable que nos proporciona el tiempo actual de la simulación
- Contadores estáticos: variables usadas para almacenar información estática sobre el rendimiento del sistema.
- Rutina de inicialización: función cuyo objetivo es inicializar el modelo de simulación en el tiempo cero.
- Rutina de tiempo: función que determina cual es el siguiente evento de la lista de eventos y avanza el reloj al tiempo en el cual ocurre el evento.
- Rutina de evento: función que actualiza el estado del sistema cuando un evento particular ocurre. Se utiliza una rutina de evento diferente para cada tipo de evento.
- Rutina de librerías: un conjunto de funciones utilizadas para generar valores aleatorios de distintas distribuciones de probabilidad.
- Generador de reportes: función que computa estimadores de las medidas de desempeño esperadas y produce un reporte cuando finaliza una simulación
- Programa principal: subprograma que invoca las rutinas mencionadas anteriormente.

5.2 Caso de estudio número 1

Los supuestos que tomaremos en este caso son los siguientes:

- μ =1
- λ =0.25
- Disciplina de cola FIFO

Los valores esperados de las medidas de rendimiento son:

- Utilización del servidor $\rho = \frac{0.25}{1} = 0.25$
- Promedio de clientes en cola $L_q = \frac{0.25^2}{1-0.25} = 0.0833$
- Tiempo promedio en cola $W_q = \frac{0.25}{1(1-0.25)} = 0.\widehat{3}$

5.2.1 Python

Primero realizamos las 10 corridas de nuestro programa, obtenemos:

Medidas de rendimiento	**SIM 1**	**SIM 2**	**SIM 3**	**SIM 4**	**SIM 5**	**SIM 6**	**SIM 7**	**SIM 8**	**SIM 9**	**SIM 10**	**Promedio**	**Diferencia Absoluta Teorico**
Utilizacion del servidor (ρ)	0.315	0.235	0.267	0.286	0.215	0.304	0.241	0.352	0.203	0.228	0.265	0.015
Promedio de clientes en cola (L_q)	0.095	0.103	0.086	0.084	0.088	0.087	0.097	0.093	0.089	0.088	0.091	0.007
Tiempo promedio en cola (W_q)	0.279	0.354	0.298	0.315	0.376	0.342	0.329	0.279	0.356	0.343	0.327	0.006

5.3 Caso de estudio número 2

Los supuestos que tomaremos en este caso son los siguientes:

- μ =1
- λ =0.5
- Disciplina de cola FIFO

Los valores esperados de las medidas de rendimiento son:

- Utilización del servidor $\rho = \frac{0.5}{1} = 0.5$
- Promedio de clientes en cola $L_q = \frac{0.5^2}{1-0.5} = 0.5$
- Tiempo promedio en cola $W_q = \frac{0.5}{1(1-0.5)} = 1$

5.3.1 Python

Primero realizamos las 10 corridas de nuestro programa, obtenemos:

Medidas de rendimiento	**SIM 1**	**SIM 2**	**SIM 3**	**SIM 4**	**SIM 5**	**SIM 6**	**SIM 7**	**SIM 8**	**SIM 9**	**SIM 10**	**Promedio**	**Diferencia Absoluta Teorico**
Utilizacion del servidor (p)	0.472	0.513	0.506	0.548	0.573	0.526	0.592	0.605	0.546	0.507	0.5388	0.038
Promedio de clientes en cola (Lq)	0.49	0.48	0.683	0.705	0.667	0.591	0.474	0.707	0.504	0.659	0.596	0.096
Tiempo promedio en cola (Wq)	1.097	1.184	1.2	0.813	1.07	1.15	1.112	1.079	0.915	0.888	1.0508	0.050

5.4 Caso de estudio número 3

Los supuestos que tomaremos en este caso son los siguientes:

- μ =1
- λ =0.5
- Disciplina de cola FIFO

Los valores esperados de las medidas de rendimiento son:

- Utilización del servidor $\rho = \frac{0.75}{1} = 0.75$
- Promedio de clientes en cola $L_q = \frac{0.75^2}{1-0.75} = 2.25$
- Tiempo promedio en cola $W_q = \frac{0.75}{1(1-0.75)} = 3$

5.4.1 Python

Primero realizamos las 10 corridas de nuestro programa, obtenemos:

Medidas de rendimiento	**SIM 1**	**SIM 2**	**SIM 3**	**SIM 4**	**SIM 5**	**SIM 6**	**SIM 7**	**SIM 8**	**SIM 9**	**SIM 10**	**Promedio**	**Diferencia Absoluta Teorico**
Utilizacion del servidor (p)	0.942	0.755	0.939	0.795	0.964	0.79	0.946	0.932	0.971	0.79	0.882	0.132
Promedio de clientes en cola (Lq)	2.413	2.497	2.302	2.071	2.437	2.062	2.483	2.164	2.214	2.293	2.294	0.044
Tiempo promedio en cola (Wq)	2.835	3.285	3.117	3.395	3.351	3.341	3.289	3.262	3.044	3.213	3.213	0.213

5.5 Caso de estudio número 4 y 5

Los supuestos que tomaremos en este caso son los siguientes:

- μ =1
- λ =1
- Disciplina de cola FIFO

Estamos frente a un caso donde la tasa de arribo es el 100% de la tasa de servicio Los valores esperados de las medidas de rendimiento son:

- Utilización del servidor $\rho = \frac{1}{1} = 1$
- Promedio de clientes en sistema $L = \frac{1}{1-1} = \infty$
- Promedio de clientes en cola $L_q = \frac{1^2}{1-1} = \infty$
- Tiempo promedio en sistema $W = \frac{1}{1(1-1)} = \infty$
- Tiempo promedio en cola $W_q = \frac{1}{1(1-1)} = \infty$

Para estos dos casos de estudio podemos llegar a la mismas observaciones en Python. Al comenzar la simulación se empieza a formar una extensa cola que continua incrementándose a medida que pasa el tiempo. Esto se debe a que el servidor puede atender a los clientes a un ritmo mucho menor de lo que un cliente llega al sistema. La única diferencia entre el caso 4 y 5 es que en el caso numero 5, al ser mayor la tasa de arribo, el fenómeno se ve intensificado y notamos que el sistema se satura de forma mas rápida, pero el resultado terminara siendo el mismo.

5.5.1 Modelo MM1 en Anylogic

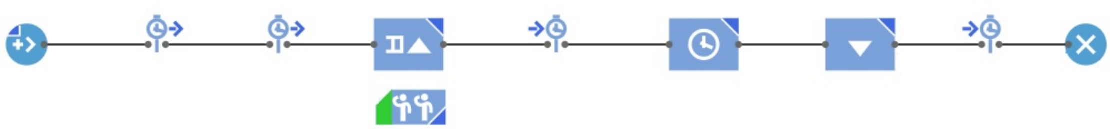

Figure 1: Modelo MM1 en software Anylogic

6 Modelo de inventarios

Un modelo es una simplificación que imita los fenómenos del mundo real, de modo que se puedan comprender las situaciones complejas y podamos hacer predicciones.

En cuanto al inventario es un puente de unión entre la producción y las ventas.

Un problema de inventario existe cuando es necesario guardar bienes físicos o mercancías con el propósito de satisfacer la demanda sobre un horizonte de tiempo especificado (finito o infinito). Casi cada empresa debe almacenar bienes para asegurar un trabajo uniforme y eficiente en sus operaciones. Las decisiones considerando cuándo hacer pedidos y en qué cantidad, son típicas de cada problema de inventario. La demanda requerida puede satisfacerse almacenando una vez según todo el horizonte de tiempo o almacenando separadamente cada unidad de tiempo durante el horizonte. Los dos casos que pueden considerarse son sobre-almacenamiento (con respecto a una unidad de tiempo) o sub-almacenamiento (con respecto al horizonte completo).

6.1 Caracteristica de un sistema de inventario

La mayoría de los Sistemas de Inventarios son modelados estocásticamente. Los modelos de simulación estocásticos producen una salida que es aleatoria, y por lo tanto, debe ser tratado solo como una estimación de las verdaderas características del modelo. Esta es una de las mayores desventajas de esta simulación.

6.2 Caso de estudio

Para el caso de estudio consideremos una compañía que vende un único producto y le gustaría saber cuánto stock solicitar para cada "n" meses próximos.

Los tiempos interdemanda son variables exponenciales aleatorias con un promedio de 0.1 meses. Los tamaños de las demandas, D, son independientes de cuando ocurre la demanda, con

$$D = \begin{cases} 1 w.p. \frac{1}{6} \\ 2 w.p. 13 \\ 3 w.p. 13 \\ 4 w.p. 16 \end{cases} \tag{9}$$

Donde w.p implica "con probabilidad".

Al comienzo de cada mes, se hace una revisión del inventario y se deciden cuanto se comprará. La formula utilizada para determinar el costo es

$$K + iZ \tag{10}$$

Siendo "K" el costo base, "i" el costo incremental y "Z" la cantidad de inventario a solicitar.

Para este ejemplo consideramos K=$32, i=$3, y el tiempo de que tarda la entrega es una variable aleatoria distribuida uniformemente entre 0.5 y 1 mes.

En este experimento, consideramos que la compañía utiliza una política estacionaria (s,S) para elegir cuanto va va a solicitar al proveedor.

A modo de comparación entre los resultados, vamos a comparar las siguientes políticas de inventario:

(s,S)									
s	20	20	20	20	40	40	40	60	60
S	40	60	80	100	60	80	100	80	100

Para determinar la cantidad de inventario a comprar, consideramos la condición si I(t)<s, se realiza una orden por S - I(t).

6.3 Caso de estudio

Los supuestos que tomaremos en este caso son los siguientes:

- Nivel de inventario inicial = 60
- Tamaño de demanda = 4
- Tasa entre demanda = 0.1 mes
- Retraso de entrega = 0.5 a 1 mes
- Tamaño de simulación = 120 meses
- K = 32
- i = 3.0
- h = 1.0
- pi = 5.0
- Número de políticas = 9

6.3.1 Python

Primero realizamos la corrida de nuestro programa, obtenemos:

Política	Promedio costo total	Promedio costo de orden	Promedio costo de mantenimiento	Promedio costo de faltante
(20, 40)	130.2674	101.7	8.3395	20.22
(20, 60)	121.1839	89.2916	17.4862	14.4060
(20, 80)	120.7108	86.2083	25.1487	9.3538
(20, 100)	128.6862	82.7166	37.0046	8.9649
(40, 60)	125.2195	97.2583	26.1465	1.8146
(40, 80)	126.6655	90.4083	35.5934	0.6637
(40, 100)	133.8434	87.3833	45.3221	1.1380
(60, 80)	145.9596	101.1833	44.7687	0.0075
(60, 100)	141.4238	87.6750	53.7229	0.0258

6.3.2 Anylogic

Tras correr la simulación para el caso de estudio en cuestión, obtenermos los siguientes valores:

Política	Promedio costo total	Promedio costo de orden	Promedio costo de mantenimiento	Promedio costo de faltante
(20, 40)	122.46	97.17	9.01	16.28
(20, 60)	123.01	95.44	18.36	11.77
(20, 80)	125.14	87.39	25.63	12.12
(20, 100)	126.82	82.04	36.37	8.41
(40, 60)	123.01	95.44	25.8	1.77
(40, 80)	121.47	85.34	35.06	1.06
(40, 100)	128.56	82.9	44.91	0.75
(60, 80)	141.6	97.71	43.89	0.001
(60, 100)	140.57	85.76	54.82	0.0005

6.3.3 Gráficos para los respectivos valores de Políticas (s,S)

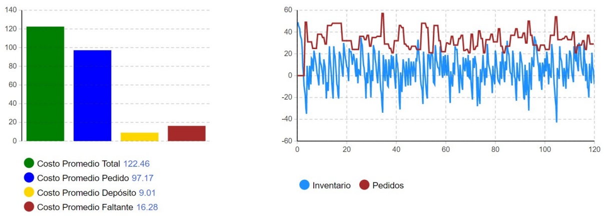

Figure 2: Política (s,S) = (20, 40)

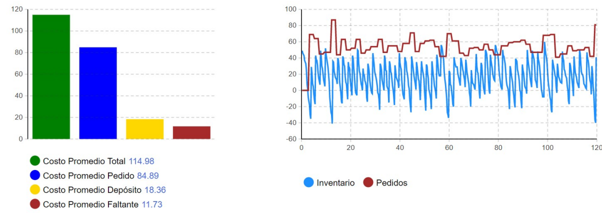

Figure 3: Política (s,S) = (20, 60)

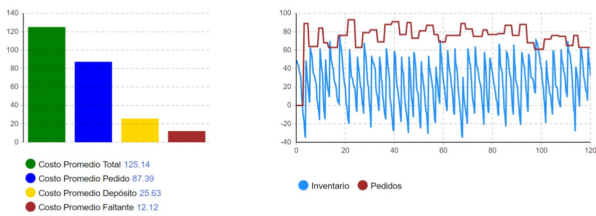

Figure 4: Política (s,S) = (20, 80)

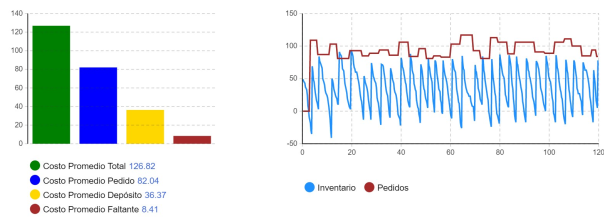

Figure 5: Política (s,S) = (20, 100)

Podemos observar que, al aumentar S, de 40 a 100, aumenta el costo promedio de mantenimiento de $9.01 a $36.37, aunque, al mismo tiempo, disminuye el costo promedio de faltante, de $16.25 a $8.41, dado que utilizar valores mayores de S implica que, estos pedidos mayores se realizarán con menor frecuencia.
Esto ocurre con todas las políticas evaluadas.

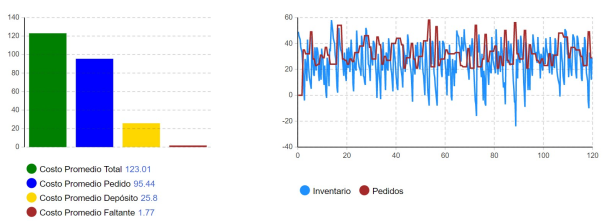

Figure 6: Política (s,S) = (40, 60)

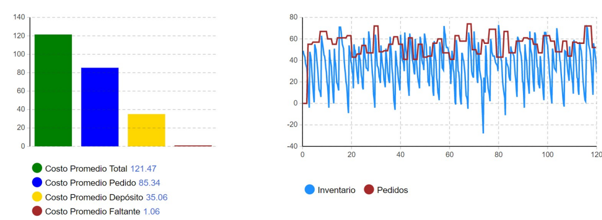

Figure 7: Política (s,S) = (40, 80)

Si incrementamos "s", y dejamos "S" en un valor fijo, como podemos observar, por ejemplo con (20,60), (40,60), vemos que el costo promedio de faltante disminuye, pero aumenta el costo de mantenimiento

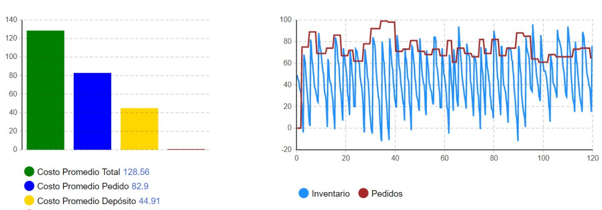

Figure 8: Política (s,S) = (40, 100)

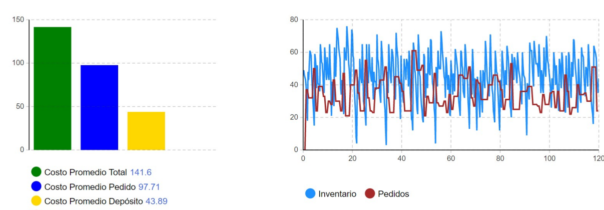

Figure 9: Política (s,S) = (60, 80)

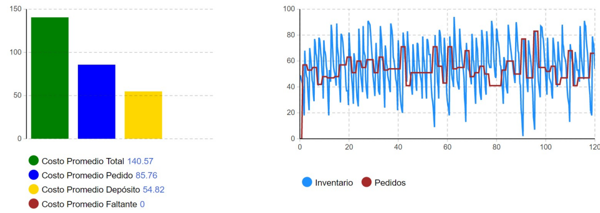

Figure 10: Política (s,S) = (60, 100)

7 Conclusión

En base al modelo de inventario, podemos decir que sería posible predecir la dirección del movimiento de los costos de orden, mantenimiento y faltante sin necesidad de ejecutar una simulación, pero no así el del costo total, ya que el criterio general de éste es la suma de tres componentes que, varían de diferentes maneras en reacción a cambios en "s" y "S", no podemos predecir la direccion del costo sin ejecutar una simulación.

Lo que logramos con las simulaciones y las comparativas realizadas es demostrar que el modelo analítico dado por los valores teóricos esperados coindicen con los valores obtenidos en las distintas simulaciones.
La finalidad es similar a la del trabajo de distribuciones de probabilidad la cual es tener una base solida para realizar experimentos posteriores, la conclusión del trabajo mencionado nos llevo a poder realizar este trabajo, y a su vez, las conclusiones de este trabajo nos permitirán tener un mayor abanico de posibilidades a la hora de realizar simulaciones más complejas.
En resumen, tras obtener los resultados de las simulaciones, podemos decir que existe una similitud entre el modelo analítico y el modelo simulado, por ende afirmamos que nuestros modelos simulados tienden a reflejar la realidad.

References

[1] http://iocontadoresuaeh.blogspot.com/2014/06/estructura-de-la-teoria-de-colas-o.html

[2] http://personales.upv.es/jpgarcia/linkeddocuments/teoriadecolasdoc.pdf

[3] Simulationmodeling Averill M. Law, W David Kelton. Capítulo 1. *Simulation modeling & analysis*, pags. 0-72.

[4] The Art of Process-Centric Modeling *The Art of Process-Centric Modeling with Anylogic*, pags. 69-99.

[5] M/M/1 Queue
`https://en.wikipedia.org/wiki/M/M/1queue`

[6] Repositorio de código Python:
`https://github.com/oberruti/Simulacion/tree/master/TP3`

[7] Repositorio de Anylogic:
`https://github.com/federomaniuk/Simulacion/tree/main/Anylogic`