



**BENEMÉRITA UNIVERSIDAD
AUTÓNOMA DE PUEBLA**

FACULTAD DE CIENCIAS FÍSICO MATEMÁTICAS

**ANÁLISIS DE UNA LÍNEA DE ESPERA
USANDO PROCESOS DE DECISIÓN
SEMI-MARKOVIANOS**

T E S I S

que para obtener el título de:

**LICENCIADO EN MATEMÁTICAS
APLICADAS**

presenta:

CARLOS CAMILO GARAY

director de tesis:

DR. HUGO ADÁN CRUZ SUÁREZ

PUEBLA, PUE.

FECHA

Índice

1. Introducción	1
2. Modelos de Colas	5
2.1. Estructura de los Modelos de Colas	5
2.2. Medidas de Rendimiento	10
2.2.1. Fórmulas de Little	11
2.3. Proceso de Nacimiento y Muerte	13
2.4. Sistema de Colas M/M/1	18
2.5. Sistema de Colas con Servidores Paralelos M/M/s	24
3. Simulación	29
3.1. Simulación Mediante Eventos Discretos	30
3.2. Simulación del Sistema $M/M/1$	30
3.3. Simulación de Sistemas no Homogéneos	32
3.3.1. Sistema de Línea de Espera con un Servidor	33
3.3.2. Sistema de Línea de Espera con Dos Servidores en Paralelo	37
4. Procesos de Control Semi-Markovianos	43
4.1. Modelo de Control Semi-Markoviano	44
4.2. Políticas de Control	45
4.3. Índice de Funcionamiento y PCO con Costo Desc.	48
4.3.1. Condiciones Sobre el Modelo de Control	49
4.4. Criterio de Optimalidad con Costo Descontado	52

4.4.1. Ecuación de Optimalidad	54
4.5. Optimalidad de una Política Umbral en $M/M/1$ con Re- cesos Repetidos	58
4.5.1. Descripción del Modelo	60
4.5.2. Ejemplo Numérico	68
Conclusiones	71
A. Otros resultados	73
A.1. Funciones	73
A.2. Teorema de Ionescu-Tulcea	74
B. Códigos	75
B.1. Código Para $M/M/1$ Homogéneo	75
B.2. Subrutina Para Generar T_s	78
B.3. Código Para $M/M/1$ no Homogéneo	78
B.4. Código Para $M/M/2$ no Homogéneo	79
B.5. Código Para el Costo Total Esperado α -Descontado	81
Bibliografía	84

Capítulo 1

Introducción

El trabajo realizado en la presente tesis está relacionado con la Teoría de Colas y la Teoría de Procesos de Decisión Semi-Markovianos. Un Proceso de Decisión es una sucesión de decisiones realizadas en un tiempo determinado siguiendo una estrategia y pagando un costo por cada decisión realizada. Se considerarán sistemas que se desarrollan en el tiempo, tales que su comportamiento tienen aspectos aleatorios. El desarrollo de estos sistemas se puede observar en tiempo discreto o continuo. En el caso discreto, el eje del tiempo es particionado en segmentos de tamaño fijo, dichos segmentos se llaman etapas o periodos. En un sistema de tiempo continuo los eventos pueden ocurrir en cualquier instante de tiempo y son registrados cuando estos ocurren.

Los Procesos de Decisión Semi-Markovianos (PDSMs) (ver, por ejemplo, [7]) son una clase importante de los Procesos de Control en tiempo continuo. Su desarrollo se puede describir de la siguiente manera: si en el tiempo de la n -ésima época de decisión el sistema se encuentra en el estado $x_n = x$, entonces el controlador elige una acción $a_n = a$ y sucede lo siguiente: el sistema permanece en el estado x durante un tiempo aleatorio no-negativo δ_{n+1} con distribución H conocida, lo que genera una recompensa inmediata que depende del estado, la acción y el tiempo de permanencia, el sistema se mueve a un nuevo estado $x_{n+1} = y$ de

acuerdo a una ley de transición, una vez ocurrido lo anterior, el proceso se repite.

Una política es una regla mediante la cual se elige una acción en cada punto de observación del proceso. Para evaluar la eficiencia de cada política se cuenta con un *criterio de rendimiento*, este trabajo usará como índice de funcionamiento el costo total esperado α -descontado.

El problema de control óptimo consiste en encontrar una política que optimice el criterio de rendimiento y ésta es llamada política óptima y, al criterio de rendimiento evaluado en tal política, función de valor óptimo. Una manera de resolver un PDSM está basado en el principio de Bellman conocido como Programación Dinámica (ver [27]), dicho principio permite resolver problemas en los que es necesario tomar decisiones en etapas sucesivas. Las decisiones tomadas en una etapa condicionan la evolución futura del sistema, afectando a las situaciones en las que el sistema se encontrará en el futuro (estados), y a las decisiones (acciones) que se plantearán en el futuro. En el presente trabajo, el principio de Programación Dinámica proporciona una técnica para determinar de manera eficiente las estrategias que optimizan un problema.

Los Procesos de Control Semi-Markovianos fueron introducidos en [19]. Estos modelos han sido estudiados y aplicados, especialmente en líneas de espera con control. Hoy en día los PCSMs son útiles para el estudio de una amplia gama de problemas de optimización, en una variedad de áreas, incluyendo la robótica, economía y en la industria, entre otras.

Con respecto a la Teoría de Colas (ver [1], [2] y [5]), una cola se presenta con frecuencia cuando se solicita un servicio por parte de una serie de clientes y tanto el servicio como los clientes son de tipo probabilístico. La teoría de colas es un modelo del comportamiento de tráfico que se ve todos los días, como lo puede ser un semáforo, la espera en un banco, etc. Los modelos de líneas de espera son muy útiles para determinar cómo operar un sistema de colas de la manera más eficaz. Proporcionar demasiada capacidad de servicio para operar el sistema implica costos excesivos; pero si no se cuenta con suficiente capacidad de servicio surgen

esperas excesivas con todas sus desafortunadas consecuencias. Los modelos permiten encontrar un balance adecuado entre el costo de servicio y la cantidad de espera.

El trabajo de tesis está estructurado de la siguiente manera.

En el Capítulo 2, se presentan los conceptos generales de la Teoría de Colas. Se hace un estudio analítico de los modelos $M/M/1$ y $M/M/s$.

En el Capítulo 3, se hace la simulación del modelo de colas $M/M/1$, que se verá en el Capítulo 2, así como para el caso *no-homogéneo* y se presentan los códigos en el Apéndice B.

Finalmente en el Capítulo 4, se dan los conceptos generales de la Teoría de Procesos de Control Semi-Markovianos; se considera un sistema de colas $M/M/1$ con periodos de receso permitidos en el servidor, donde el servidor es apagado tan pronto la cola se vacía. Se supone también que las duraciones de estos periodos forman una secuencia de variables aleatorias independientes e idénticamente distribuidas con distribución exponencial. Al final de un período de receso, el servidor puede o bien ser activado si la cola no está vacía o tomar nuevamente otro período de receso. En dicho sistema se incurren los siguientes costos: un costo de almacenamiento h por unidad de tiempo y por cliente en el sistema y un costo fijo γ cada vez que el servidor se vuelve a activar. En este capítulo se mostrará que existe una política umbral que minimiza el criterio de costo total esperado α -descontado.

Capítulo 2

Modelos de Colas

En este capítulo se desarrollarán los elementos básicos de un sistema de línea de espera así como dos modelos de teoría de colas, a saber, $M/M/1$ y $M/M/s$.

Una situación de línea de espera (cola o fila) se genera cuando un cliente llega a una instalación y se forma en una línea de espera, el servidor elige a un cliente de dicha línea para comenzar a prestar el servicio, al culminar dicho servicio se repite el proceso de elegir a un nuevo cliente en espera.

2.1. Estructura de los Modelos de Colas

Proceso básico de colas

El proceso básico supuesto por la mayoría de los modelos de colas es el siguiente. Los *clientes* que requieren un servicio se generan en el tiempo en una *fente de entrada*. Luego, entran al *sistema* y se unen a una *cola*. En determinado momento se selecciona un miembro de la cola para proporcionarle el servicio mediante alguna regla conocida como *disciplina de la cola*. Se lleva a cabo el servicio que el cliente requiere mediante un *mecanismo de servicio*, y después el cliente sale del sistema de colas. La

Figura 2.1 describe este proceso.

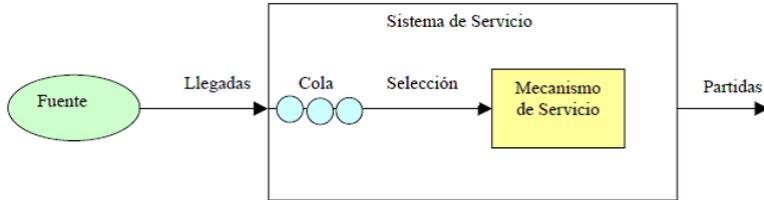


Figura 2.1: Proceso básico de colas

Fuente de entrada

Una característica de la fuente de entrada es su tamaño. El *tamaño* es el número total de clientes que pueden requerir servicio en determinado momento. Esta población a partir de la cual surgen las unidades que llegan se conoce como **población de entrada**. Puede suponerse que el tamaño es *infinito* o *finito*. Debido a que los cálculos son mucho más sencillos en el caso del tamaño infinito, este supuesto se hace a menudo aun cuando el tamaño real sea un número fijo relativamente grande, y debe tomarse como un supuesto implícito en cualquier modelo en el que no se establezca otra cosa (ver [5]).

En los modelos también se debe especificar el patrón estadístico mediante el cual se generan los clientes en el tiempo. El supuesto normal es que se generan de acuerdo con un *proceso Poisson*; es decir, el número de clientes que llegan hasta un momento específico tiene una distribución de Poisson.

Cola

La cola es donde los clientes esperan antes de recibir el servicio. Una cola se caracteriza por el número máximo permisible de clientes que puede admitir. El supuesto de una cola infinita es el estándar de la mayoría de

los modelos, incluso en situaciones en las que en realidad existe una cota superior sobre el número permitido de clientes, puesto que manejar una cota así puede ser un factor que complique el análisis.

Disciplina de la cola

La disciplina de la cola se refiere al orden en el que sus miembros se seleccionan para recibir el servicio. Por ejemplo, puede ser: primero en entrar, primero en salir (“First In First Out”, FIFO); en orden aleatorio; último en llegar, primero en ser atendido (“Last Come First Served”, LCFS); de acuerdo con algún procedimiento de prioridad o con algún otro orden. En los modelos de colas se supone como normal a la disciplina de primero en entrar, primero en salir, a menos que se establezca de otra manera.

Mecanismo de servicio

El mecanismo de servicio consiste en una o más *estaciones de servicio*, cada una de ellas con uno o más *canales de servicios paralelos*, llamados **servidores**. Si existe más de una estación de servicio, el cliente puede recibirlo de una secuencia de ellas (*canales de servicio en serie*). En una estación dada, el cliente entra en uno de estos canales y el servidor le presta el servicio completo. Los modelos de colas deben especificar el arreglo de las estaciones y el número de servidores (canales paralelos) en cada una de ellas. Los modelos más elementales suponen una estación, ya sea con un servidor o con un número finito de servidores.

El tiempo que transcurre desde el inicio del servicio para un cliente hasta su terminación en una estación se llama **tiempo de servicio**. Un modelo de un sistema de colas determinado debe especificar la distribución de probabilidad de los tiempos de servicio de cada servidor, aunque es común suponer la misma distribución para todos los servidores. La distribución del tiempo de servicio que más se usa en la práctica es la distribución *exponencial*.

Terminología

Usaremos la siguiente terminología estándar:

Estado del sistema	= número de clientes en el sistema.
Longitud de la cola	= número de clientes que esperan servicio. = estado del sistema menos número de clientes a quienes se les da el servicio.
$N(t)$	= número de clientes en el sistema de colas en el tiempo t ($t \geq 0$).
$p_n(t)$	= probabilidad de que exactamente n clientes estén en el sistema en el tiempo t .
s	= número de servidores en el sistema.
λ_n	= tasa media de llegadas de nuevos clientes cuando hay n clientes en el sistema.
μ_n	= tasa media de servicio en todo el sistema cuando hay n clientes en el sistema.

Notación

La notación para resumir las características principales de una línea de espera fue ideada originalmente por D. G. Kendall (1953) de la siguiente forma, $A/B/X$ que se le conoce como *notación Kendall*. Después A. M. Lee (1966) agregó los símbolos Y/Z , dando así el nombre de *notación Kendall-Lee*, donde:

- A es la distribución de tiempos entre llegadas:
 1. $M \equiv$ tiempos entre llegadas exponenciales
 2. $D \equiv$ tiempos entre llegadas deterministas
 3. $E_k \equiv$ tiempos entre llegadas Erlang con parámetro k
 4. $G \equiv$ tiempos entre llegadas generales

- B es la distribución de tiempos de servicios, puede tomar los mismos valores que A .
- X es el número de servidores en paralelo.
- Y es la capacidad del sistema, se puede omitir si es infinita.
- Z es la disciplina del sistema, se puede omitir si es FIFO.

Cuando λ_n es constante para toda n , esta constante se denota por λ . Cuando la tasa media de servicio por servidor ocupado es constante para toda $n \geq 1$, esta constante se denota por μ . Por consiguiente, $\frac{1}{\lambda}$ y $\frac{1}{\mu}$ es el tiempo esperado entre llegadas y el tiempo esperado de servicio, respectivamente. Asimismo, $\rho = \frac{\lambda}{s\mu}$ es el **factor de utilización** de la instalación de servicio, es decir, la fracción esperada de tiempo que los servidores individuales están ocupados, puesto que $\frac{\lambda}{s\mu}$ representa la fracción de la capacidad de servicio del sistema ($s\mu$) que utilizan en promedio los clientes que llegan (λ). Cuando $\rho > 1$ ($\lambda > \mu$), el número promedio de llegadas al sistema excede a la tasa de promedio máximo de servicio del sistema, y se esperarí que, conforme pasa el tiempo, la cola se hiciera más y más grande, hasta que, en algún punto, a los clientes ya no se les dejará entrar.

Requerimos cierta notación para describir los resultados de *estado estable*. Cuando un sistema de colas inicia su operación, el estado del sistema (el número de clientes que esperan en el sistema) se encuentra bastante afectado por el estado inicial y el tiempo que ha pasado desde el inicio. Se dice entonces que el sistema se encuentra en **condición transitoria**. Sin embargo, una vez que ha pasado suficiente tiempo, el estado del sistema se vuelve, en esencia, independiente del estado inicial y del tiempo transcurrido. Así, en este contexto, se puede decir que el sistema ha alcanzado su **condición de estado estable**, en la que la distribución de probabilidad del estado del sistema se conserva a través del tiempo. La teoría de colas tiende a dedicar su análisis a la condición de estado estable, en parte porque el caso transitorio es analíticamente

más difícil. Así, para que el sistema sea estable, el factor de utilización, ρ , debe ser estrictamente menor que 1.

2.2. Medidas de Rendimiento

En los sistemas de líneas de espera, las medidas de rendimiento, también conocidas como medidas de efectividad (ver [2]), son las medidas de mayor interés, generalmente hay tres tipos: (1) la medida sobre el tiempo de espera que un cliente está dispuesto a tolerar; (2) una medida en cómo los clientes se van acumulando; y (3) una medida del tiempo de inactividad de los servidores.

Hay dos tipos de tiempo de espera, el tiempo de espera en la cola y el tiempo de espera en el sistema (tiempo en la cola mas tiempo de servicio). Dependiendo del sistema de colas que se esté estudiando, una medida puede ser de mayor interés que la otra. Por ejemplo, en un parque de diversiones, es el tiempo de espera en la cola el que nos gustaría fuera mínimo. Por otro lado, si se está tratando con máquinas que necesitan ser reparadas, entonces el tiempo total (tiempo de espera en la cola mas tiempo de reparación) es el que desearíamos fuera lo más pequeño posible. Análogamente, hay dos tipos de medidas de acumulamiento, longitud de la cola y la longitud del sistema. La medida de inactividad puede incluir el porcentaje de tiempo que cualquier servidor puede estar inactivo, o el tiempo de todo el sistema cuando el servidor carece de clientes.

Lo que regularmente se desea encontrar en los modelos de colas es la distribución de probabilidad para el número total de clientes en el sistema al tiempo t , $N(t)$. Sea $p_n(t) = Pr(N(t) = n)$, y $p_n = Pr(N = n)$, que es la probabilidad de estado estable de n clientes en el sistema. Si consideramos s -servidores en el sistema, dos medidas de valores-esperado de mayor interés son el número esperado de clientes en el sistema,

$$L = E[N] = \sum_{n=0}^{\infty} np_n, \quad (2.1)$$

y el número esperado en la cola,

$$L_q = E[N_q] = \sum_{n=s}^{\infty} (n-s)p_n. \quad (2.2)$$

donde N_q es el número de clientes esperando en la cola.

2.2.1. Fórmulas de Little

Una de las relaciones más fuertes en la teoría de colas fue desarrollada por John D. C. Little por los años 60's (véase [2]). Desarrollaremos las Fórmulas de Little haciendo que T_q represente el tiempo que un cliente tarda en la cola antes de recibir el servicio y T que represente el tiempo total que un cliente espera en el sistema, ($T = T_q + S$), donde S es el tiempo de servicio, y T , T_q y S son variables aleatorias. A partir de esta nomenclatura $W_q = E[T_q]$ y $W = E[T]$, son el tiempo esperado en la cola y el tiempo esperado en el sistema, respectivamente. Sea

$Pr(n, t)$ = Probabilidad de que n clientes lleguen durante el tiempo t

$$W(t) = Pr(T \leq t \mid n \text{ clientes en el sistema}) \cdot p_n(t).$$

Primero calculemos p_n de la siguiente forma:

$$\begin{aligned} p_n &= \int_0^{\infty} Pr(n, t) dt \\ &= \int_0^{\infty} Pr(n \text{ llegadas durante } T \mid T = t) dw(t) \end{aligned}$$

donde $w(t)$ es la distribución acumulada de la variable T .

Dado que la probabilidad de llegadas tiene una distribución Poisson se tiene que

$$p_n = \frac{1}{n!} \int_0^{\infty} [(\lambda t)^n \exp(-\lambda t)] dw(t),$$

utilizando el resultado anterior y la ecuación (2.1) tenemos

$$\begin{aligned}
 L &= E[N] = \sum_{n=1}^{\infty} np_n \\
 &= \sum_{n=1}^{\infty} \frac{n}{n!} \int_0^{\infty} [(\lambda t)^n \exp(-\lambda t)] dw(t) \\
 &= \int_0^{\infty} dw(t) \exp(-\lambda t) \sum_{n=1}^{\infty} \frac{(\lambda t)^n}{(n-1)!} \\
 &= \int_0^{\infty} (\lambda t) \exp(-\lambda t) dw(t) \sum_{n=1}^{\infty} \frac{(\lambda t)^{n-1}}{(n-1)!} \\
 &= \int_0^{\infty} (\lambda t) \exp(-\lambda t) \exp(\lambda t) dw(t) \\
 &= \int_0^{\infty} (\lambda t) dw(t) = \lambda \int_0^{\infty} t dw(t) \\
 &= \lambda E[T] = \lambda W.
 \end{aligned}$$

Así, existe una relación entre L y W de manera que cualquier medida se determina automáticamente a partir de la otra:

$$L = \lambda W.$$

Debido a que John D. C. Little proporcionó la primera demostración rigurosa, dichas expresiones se les denominan **Fórmulas de Little**. Además, la misma demostración prueba que

$$L_q = \lambda W_q.$$

Puesto que $T = T_q + S$, entonces $E[T] = E[T_q] + E[S]$, o lo que es equivalente

$$W = W_q + \frac{1}{\mu},$$

donde μ representa la tasa de salidas en el sistema, multiplicando ambos miembros de la ecuación anterior por λ se obtiene

$$L = L_q + \frac{\lambda}{\mu}.$$

Así que sólo es necesario encontrar una de las cuatro medidas de valores esperados, para obtener las demás usando las Fórmulas de Little.

2.3. Proceso de Nacimiento y Muerte

La mayor parte de los modelos elementales de colas suponen que las entradas y salidas del sistema ocurren de acuerdo con un *proceso de nacimiento y muerte* (ver [2] y [5]), el término **nacimiento** se refiere a la *llegada* de un nuevo cliente al sistema de colas, mientras que el término **muerte** se refiere a la *salida* del cliente servido. El proceso de nacimiento y muerte describe en *términos probabilísticos* cómo cambia $N(t)$ al aumentar t . En general, sostiene que los nacimientos y muertes individuales ocurren de manera aleatoria, y que sus tasas medias de ocurrencia dependen del estado actual del sistema. De manera más precisa, los supuestos del proceso de nacimiento y muerte son los siguientes:

Supuesto 1. Dado $N(t) = n$, la distribución de probabilidad actual del tiempo que falta para el próximo nacimiento es exponencial con parámetro λ_n ($n = 0, 1, 2, \dots$).

Supuesto 2. Dado $N(t) = n$, la distribución de probabilidad actual del tiempo que falta para la próxima muerte es exponencial con parámetro μ_n ($n = 0, 1, 2, \dots$).

Supuesto 3. La variable aleatoria del supuesto 1 y la variable aleatoria del supuesto 2 son mutuamente independientes. La siguiente transición del estado del proceso es

$$n \rightarrow n + 1 \text{ (un solo nacimiento)}$$

o

$$n \rightarrow n - 1 \text{ (una sola muerte)}$$

lo que depende de cuál de las dos variables es más pequeña.

En el caso de un sistema de colas, λ_n y μ_n representan, respectivamente, la *tasa media de llegada* y la *tasa media de terminaciones*

de servicio, cuando hay n clientes en el sistema. Como se dijo anteriormente en algunos sistemas de colas, los valores de las λ_n serán las mismas para todos los valores de n , y las μ_n también serán las mismas para toda n excepto para aquella n tan pequeña que el servidor esté desocupado (es decir, $n = 0$). Sin embargo, las λ_n y las μ_n también pueden variar en forma considerable con n , para algunos sistemas de colas (ver [1] y [5]).

Una de las formas en las que λ_n puede ser diferente para valores distintos de n es, si los clientes que llegan se pueden *perder* (rechazar la entrada al sistema) con mayor probabilidad a medida que n aumenta. De manera similar, μ_n puede ser diferente ante valores distintos de n debido a que existe una mayor probabilidad de que los clientes *renuncien* (se vayan sin haber sido servidos) a medida que aumenta el tamaño de la cola.

Análisis del proceso de nacimiento y muerte

Como consecuencia de los Supuestos 1 y 2, el proceso de nacimiento y muerte es un tipo especial de *cadena de Markov en tiempo continuo*.

Las flechas del diagrama en la Figura 2.2 muestran las únicas *transiciones* posibles en el estado del sistema (como lo especifica el Supuesto 3) y el elemento junto a cada flecha es la tasa media de esa transición (según los supuestos 1 y 2) cuando el sistema se encuentra en el estado que hay en la base de la flecha.

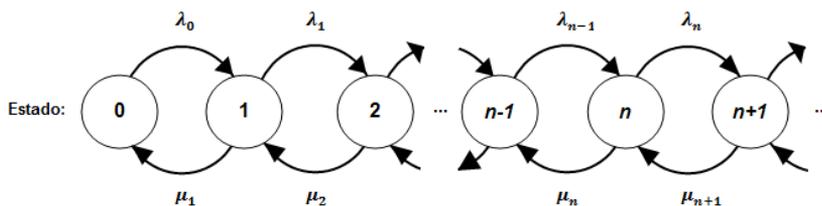


Figura 2.2: Diagrama de tasas del proceso de nacimiento y muerte.

Suponemos que el flujo de entrada es el mismo que el flujo de salida, es decir:

Principio de tasa de entrada = tasa de salida. Para cualquier estado n ($n = 0, 1, 2, \dots$) del sistema, la tasa media de entrada=tasa media de salida.

La ecuación que expresa este principio se llama **ecuación de balance** del estado n . Después de construir las ecuaciones de balance de todos los estados en términos de las probabilidades p_n desconocidas, se puede resolver este sistema de ecuaciones (mas una ecuación que establezca que las probabilidades deben sumar 1) para encontrarlas.

A fin de ilustrar una ecuación de balance, consideremos el estado 0. El proceso entra a este estado *sólo* desde el estado 1. En consecuencia, la probabilidad de estado estable de encontrarse en el estado 1 $\cdot (p_1)$ representa la proporción de tiempo que es *posible* que el proceso entre al estado 0. Dado que el proceso se encuentra en el estado 1, la tasa media de entrada al estado 0 es μ_1 . (En otras palabras, para cada unidad acumulada de tiempo que el proceso pasa en el estado 1, el número esperado de veces que lo dejaría para entrar al estado 0 es μ_1 .) Desde cualquier otro estado, esta tasa media es 0. Por lo tanto, la tasa media global a la que el proceso deja su estado actual para entrar al estado 0 (la tasa media de entrada) es

$$\mu_1 p_1 + 0(1 - p_1) = \mu_1 p_1.$$

Por el mismo razonamiento, la tasa media de salida debe ser $\lambda_0 p_0$, de manera que la ecuación de balance del estado 0 es

$$\mu_1 p_1 = \lambda_0 p_0.$$

En el caso de todos los demás estados, existen dos transiciones posibles, hacia adentro y hacia afuera del estado. Entonces, cada lado de las ecuaciones de balance de estos estados representa la suma de las tasas medias de las dos transiciones incluidas. Por lo demás, el razonamiento

es igual que para el estado 0. Estas ecuaciones de balance se resumen en la Tabla 2.1

Estado	Tasa de entrada = Tasa de salida
0	$\mu_1 p_1 = \lambda_0 p_0$
1	$\lambda_0 p_0 + \mu_2 p_2 = (\lambda_1 + \mu_1) p_1$
2	$\lambda_1 p_1 + \mu_3 p_3 = (\lambda_2 + \mu_2) p_2$
\vdots	\vdots
$n - 1$	$\lambda_{n-2} p_{n-2} + \mu_n p_n = (\lambda_{n-1} + \mu_{n-1}) p_{n-1}$
n	$\lambda_{n-1} p_{n-1} + \mu_{n+1} p_{n+1} = (\lambda_n + \mu_n) p_n$
\vdots	\vdots

Tabla 2.1: Ecuaciones de balance del proceso de nacimiento y muerte

De la Tabla 2.1 se observa que la primera ecuación de balance contiene dos variables (p_0 y p_1), las primeras dos ecuaciones contienen tres variables (p_0 , p_1 y p_2), y así sucesivamente, de manera que siempre se tiene una variable adicional. Por lo tanto, el procedimiento para resolver estas ecuaciones es despejar todas las variables en términos de una de ellas, entre las cuales la más conveniente es p_0 . La primera ecuación se usa para despejar p_1 en términos de p_0 ; después se usa este resultado y la segunda ecuación para obtener p_2 en términos de p_0 , etc. Al final, el requisito de que la suma de todas las probabilidades debe ser igual a 1 se puede usar para evaluar p_0 . Es decir

$$\sum_{n=0}^{\infty} p_n = 1$$

se usará para encontrar p_0 .

Resultados del proceso de nacimiento y muerte

Al aplicar este procedimiento se obtienen los siguientes resultados:

estado			
0 :	p_1	$= \frac{\lambda_0}{\mu_1} p_0$	
1 :	p_2	$= \frac{\lambda_1}{\mu_2} p_1 + \frac{1}{\mu_2} (\mu_1 p_1 - \lambda_0 p_0)$	$= \frac{\lambda_1}{\mu_2} p_1$ $= \frac{\lambda_1 \lambda_0}{\mu_2 \mu_1} p_0$
2 :	p_3	$= \frac{\lambda_2}{\mu_3} p_2 + \frac{1}{\mu_3} (\mu_2 p_2 - \lambda_1 p_1)$	$= \frac{\lambda_2}{\mu_3} p_2$ $= \frac{\lambda_2 \lambda_1 \lambda_0}{\mu_3 \mu_2 \mu_1} p_0$
⋮	⋮		
$n - 1 :$	p_n	$= \frac{\lambda_{n-1}}{\mu_n} p_{n-1} + \frac{1}{\mu_n} (\mu_{n-1} p_{n-1} - \lambda_{n-2} p_{n-2})$	$= \frac{\lambda_{n-1}}{\mu_n} p_{n-1}$ $= \frac{\lambda_{n-1} \cdots \lambda_0}{\mu_n \cdots \mu_1} p_0$
$n :$	p_{n+1}	$= \frac{\lambda_n}{\mu_{n+1}} p_n + \frac{1}{\mu_{n+1}} (\mu_n p_n - \lambda_{n-1} p_{n-1})$	$= \frac{\lambda_n}{\mu_{n+1}} p_n$ $= \frac{\lambda_n \cdots \lambda_0}{\mu_{n+1} \cdots \mu_1} p_0$
⋮	⋮		

Para simplificar la notación, sea

$$C_n = \frac{\lambda_{n-1} \lambda_{n-2} \cdots \lambda_0}{\mu_n \mu_{n-1} \cdots \mu_1}, \quad n = 1, 2, \dots, \quad (2.3)$$

y después se define $C_n = 1$, para $n = 0$ (ver [5]). En este contexto, las probabilidades de estado estable son

$$p_n = C_n p_0, \quad n = 0, 1, 2, \dots \quad (2.4)$$

El requisito

$$\sum_{n=0}^{\infty} p_n = 1, \quad (2.5)$$

implica que (2.5) se transforma en

$$\left(\sum_{n=0}^{\infty} C_n \right) p_0 = 1, \quad (2.6)$$

así,

$$p_0 = \left(\sum_{n=0}^{\infty} C_n \right)^{-1}. \quad (2.7)$$

2.4. Sistema de Colas M/M/1

Éste es el más simple de los modelos de colas usado en la práctica. Los arribo llegan según un proceso Poisson de tasa λ . Cada cliente, al llegar, pasa directamente a ser servido si el servidor está libre y, si no, se une a la cola. Cuando el servidor completa el procesamiento de un cliente, éste abandona el sistema y el siguiente cliente en la cola, si hay alguno, pasa a ser servido.

Recordemos que $N(t)$ es el número de clientes en el sistema de colas en el tiempo t , luego $\{N(t), t \geq 0\}$ tiene una distribución tipo Poisson

$$Pr[N(t) = j] = e^{-\lambda t} \frac{(\lambda t)^j}{j!}, \quad j = 0, 1, 2, \dots$$

Las funciones de densidad para los tiempos de entre llegadas y los tiempos de servicio para el sistema $M/M/1$, están dadas, respectivamente, por

$$a(t) = \lambda e^{-\lambda t},$$

$$b(t) = \mu e^{-\mu t}$$

donde $\frac{1}{\lambda}$ es el tiempo promedio de entre llegadas y $\frac{1}{\mu}$ es el tiempo promedio de servicio. Los tiempos de entre llegadas, así como los tiempos de servicio, se suponen independientes. Entonces, el sistema $M/M/1$ es un proceso de nacimiento y muerte con $\lambda_n = \lambda$ y $\mu_n = \mu$, para toda n , y a este sistema se le llama *sistema homogéneo*. Las llegadas pueden considerarse como *nacimientos*, puesto que si en el sistema hay n clientes y una llegada ocurre, el estado cambia a $n + 1$. Por el otro lado, una salida que ocurra cuando se está en n , manda al sistema a $n - 1$ y puede considerarse como una *muerte*. El diagrama de transición de estados para un sistema de colas $M/M/1$ se puede ver en la Figura 2.3.

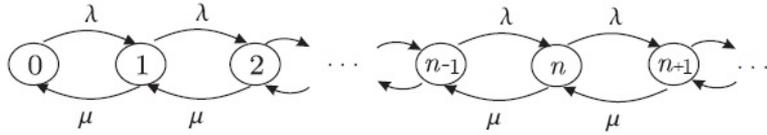


Figura 2.3: Diagrama de transición para el sistema M/M/1.

A partir de la Tabla 2.1, tenemos el siguiente sistema de ecuaciones de equilibrio para el caso M/M/1

$$\begin{aligned}
 p_1 &= \frac{\lambda}{\mu} p_0, \\
 p_2 &= \left(\frac{\lambda}{\mu}\right)^2 p_0, \\
 &\dots \\
 p_n &= \left(\frac{\lambda}{\mu}\right)^n p_0.
 \end{aligned}$$

Para este tipo de sistema, se define la intensidad de tráfico, también conocida como factor de utilización, de la siguiente manera

$$\rho = \frac{\lambda}{\mu},$$

recordemos que, para este caso, $s = 1$ y para que el sistema sea estable

$$\rho < 1,$$

este parámetro mide la relación entre la media de los clientes por unidad de tiempo y la capacidad de atenderlos por unidad de tiempo.

Si se cumple la condición de no saturación, las probabilidades del estado estable existen, de la ecuación (2.3) tenemos

$$C_n = \rho^n, \tag{2.8}$$

a partir de las ecuaciones (2.7) y (2.8) se tiene

$$p_0 = \frac{1}{\sum_{n=0}^{\infty} \rho^n}.$$

Ahora, $\sum_{n=0}^{\infty} \rho^n$ es la serie geométrica $1 + \rho + \rho^2 + \rho^3 \cdots$ y converge si y sólo si $\rho < 1$.

Haciendo uso de el bien conocido resultado para la suma de los términos de una progresión geométrica,

$$\sum_{n=0}^{\infty} \rho^n = \frac{1}{1 - \rho} \quad (\rho < 1),$$

obtenemos

$$p_0 = 1 - \rho \quad (\rho < 1), \quad (2.9)$$

finalmente

$$p_n = (1 - \rho)\rho^n \quad (\rho < 1). \quad (2.10)$$

Medidas de rendimiento

Como se vio en la Sección 2.2, las medidas de rendimiento nos ayudan a calcular la medida de efectividad del sistema. Calcularemos el número esperado de clientes en la cola y en el sistema para el modelo $M/M/1$. Sea L como se definió en la ecuación (2.1), entonces

$$\begin{aligned} L = E[N] &= \sum_{n=0}^{\infty} np_n \\ &= (1 - \rho) \sum_{n=0}^{\infty} n\rho^n, \end{aligned}$$

desarrollando la suma

$$\begin{aligned} \sum_{n=0}^{\infty} n\rho^n &= \rho + 2\rho^2 + 3\rho^3 + \cdots \\ &= \rho(1 + 2\rho + 3\rho^2 + \cdots) \\ &= \rho \sum_{n=1}^{\infty} n\rho^{n-1}. \end{aligned}$$

Observemos que $\sum_{n=1}^{\infty} n\rho^{n-1}$ es simplemente la derivada de $\sum_{n=0}^{\infty} \rho^n$ con respecto a ρ , como la sumatoria y la derivación se pueden inter-

cambiar y puesto que $\rho < 1$,

$$\sum_{n=0}^{\infty} \rho^n = \frac{1}{1-\rho};$$

por lo tanto

$$\begin{aligned} \sum_{n=1}^{\infty} n\rho^{n-1} &= \frac{d[1/(1-\rho)]}{d\rho} \\ &= \frac{1}{(1-\rho)^2}. \end{aligned}$$

Así que el número esperado en el sistema es

$$L = \frac{\rho(1-\rho)}{(1-\rho)^2},$$

o simplemente

$$\boxed{L = \frac{\rho}{1-\rho} = \frac{\lambda}{\mu-\lambda}}. \quad (2.11)$$

Como N_q denota el número de clientes en la cola y L_q su valor esperado, entonces, a partir de la ecuación (2.2), tenemos

$$\begin{aligned} L_q &= \sum_{n=1}^{\infty} (n-1)p_n \\ &= \sum_{n=1}^{\infty} np_n - \sum_{n=1}^{\infty} p_n \\ &= L - (1-p_0) \\ &= \frac{\rho}{1-\rho} - \rho, \end{aligned}$$

así, la media de la longitud de la cola es

$$\boxed{L_q = \frac{\rho^2}{1-\rho} = \frac{\lambda^2}{\mu(\mu-\lambda)}}. \quad (2.12)$$

Para encontrar los valores del tiempo de espera en la cola y en el sistema, W_q y W , respectivamente, hacemos uso de las Fórmulas de Little, vistas en la Sección 2.2 para obtener

$$\boxed{W = \frac{L}{\lambda} = \frac{\rho}{\lambda(1-\rho)} = \frac{1}{\mu-\lambda}}, \quad (2.13)$$

y

$$W_q = \frac{L_q}{\lambda} = \frac{\rho^2}{\lambda(1-\rho)} = \frac{\rho}{\mu-\lambda}. \quad (2.14)$$

Distribución del tiempo de espera

Con todo esto, tenemos bien caracterizado el comportamiento estacionario de la cola. Una característica interesante del sistema $M/M/1$ es que, además, es relativamente sencillo proporcionar la distribución del tiempo que un cliente pasa en la cola y en el sistema (ver [1] y [2]).

Sea T la variable que denote el tiempo que pasa un cliente en el sistema y $W(t)$ su función de distribución. Para obtener $W(t)$ condicionamos en el número de clientes en el sistema a la llegada del cliente.

$$W(t) = Pr(T \leq t) = \sum_{n=0}^{\infty} Pr(T \leq t \mid N = n)Pr(N = n).$$

Donde N es la variable que cuenta el número de clientes en el sistema. Ahora bien,

- Si $n = 0$, el cliente que llega estará en el sistema su tiempo de servicio.
- Si $n \geq 1$, habrá un cliente en servicio y $n - 1$ esperando delante de él. Debido a la pérdida de memoria de la exponencial (para el tiempo de servicio del primer cliente), el cliente deberá esperar un tiempo que es suma de $n + 1$ exponenciales independientes de parámetro μ , que sabemos sigue una distribución Gamma con parámetro $\alpha = n + 1$ y $\beta = \mu$ cuya función de densidad es

$$f(x) = \frac{\mu^{n+1} e^{-\mu x} x^n}{n!}, \quad x \geq 0.$$

Así,

$$\begin{aligned}
 W(t) &= \sum_{n=0}^{\infty} \left(\int_0^t \mu e^{-\mu x} \frac{(\mu x)^n}{n!} dx \right) \left(1 - \frac{\lambda}{\mu} \right) \left(\frac{\lambda}{\mu} \right)^n \\
 &= \int_0^t (\mu - \lambda) e^{-\mu x} \sum_{n=0}^{\infty} \frac{(\lambda x)^n}{n!} dx \\
 &= \int_0^t (\mu - \lambda) e^{-(\mu - \lambda)x} dx \\
 &= 1 - e^{-(\mu - \lambda)t}.
 \end{aligned}$$

Por lo tanto, el tiempo que un cliente pasa en el sistema es una variable aleatoria exponencial de parámetro $\mu - \lambda$, cuya media es

$$E[T] = \frac{1}{\mu - \lambda},$$

como ya se había obtenido en la ecuación (2.13).

Sea ahora T_q el tiempo que pasa un cliente en la cola, para obtener la función de distribución $W_q(t)$ procedemos de manera similar a como se hizo para obtener $W(t)$.

$$W_q(t) = Pr(0 \leq T_q \leq t).$$

Notemos primero que

$$W_q(0) = Pr(T_q = 0) = Pr(0 \text{ clientes en el sistema}) = 1 - \rho,$$

ya que ρ es la probabilidad de que el sistema esté ocupado.

Para $t > 0$, tenemos que

$$\begin{aligned}
 W_q(t) &= Pr(T_q = 0) + Pr(0 < T_q \leq t), \\
 Pr(0 < T_q \leq t) &= \sum_{n=1}^{\infty} Pr(T_q \leq t \mid N = n) Pr(N = n).
 \end{aligned}$$

Si un cliente llega cuando hay n clientes en el sistema ($n - 1$ delante de él en la cola y 1 en servicio), tendrá que esperar en cola un tiempo que es suma de n tiempos exponenciales, es decir, de n variables aleatorias independientes e idénticamente distribuidas (v.a.i.i.d.) según

una exponencial de tasa μ , que es una Gamma de parámetros $\alpha = n$ y $\beta = \mu$, cuya función de densidad es

$$f(x) = \frac{\mu^n e^{-\mu x} x^{n-1}}{(n-1)!}, \quad x \geq 0, n \geq 1.$$

Así,

$$\begin{aligned} Pr(0 < T_q \leq t) &= \sum_{n=1}^{\infty} \left(\int_0^t \mu e^{-\mu x} \frac{(\mu x)^{n-1}}{(n-1)!} dx \right) \left(1 - \frac{\lambda}{\mu} \right) \left(\frac{\lambda}{\mu} \right)^n \\ &= \int_0^t (\mu - \lambda) e^{-\mu x} \sum_{n=1}^{\infty} \frac{(\lambda x)^{n-1}}{(n-1)!} dx \\ &= \frac{\lambda}{\mu} \int_0^t (\mu - \lambda) e^{-(\mu-\lambda)x} dx \\ &= \rho(1 - e^{-(\mu-\lambda)t}), \end{aligned}$$

con lo que, si $t > 0$

$$\begin{aligned} W_q(t) &= Pr(T_q = 0) + Pr(0 < T_q \leq t) \\ &= 1 - \rho + \rho(1 - e^{-(\mu-\lambda)t}) \\ &= 1 - \rho e^{-(\mu-\lambda)t} \\ &= 1 - \rho e^{-\mu(1-\rho)t} \quad (t > 0). \end{aligned}$$

Así, la distribución del tiempo de espera en la cola es

$$\boxed{W_q(t) = 1 - \rho e^{-\mu(1-\rho)t} \quad (t > 0).}$$

Igual que antes se puede comprobar que $E[T_q] = W_q$.

2.5. Sistema de Colas con Servidores Paralelos M/M/s

Ahora para el modelo $M/M/s$, los servidores son independientes e idénticamente distribuidos con distribución de tiempos de servicio exponenciales, y con llegadas nuevamente Poisson, así, tenemos un proceso de

nacimiento y muerte. Por tanto $\lambda_n = \lambda$ para toda n , y falta determinar μ_n , para los s servidores.

Si hay más de s clientes en el sistema, todos los s servidores estarán ocupados, cada uno de ellos con una tasa μ , y la tasa de salidas del sistema será $s\mu$. Cuando hay menos de s clientes en el sistema, digamos $n < s$, solamente n de los s servidores estarán ocupados y la tasa de salida del sistema será $n\mu$ (ver [2]), por lo tanto μ_n podrá escribirse como

$$\mu_n = \begin{cases} n\mu & (1 \leq n < s), \\ s\mu & (n \geq s). \end{cases} \quad (2.15)$$

Utilizando (2.15) en la ecuación (2.4) y el hecho de que $\lambda_n = \lambda$ para toda n , obtenemos

$$p_n = \begin{cases} \frac{\lambda^n}{n!\mu^n} & (1 \leq n < s), \\ \frac{\lambda^n}{s^{n-s}s!\mu^n} & (n \geq s). \end{cases} \quad (2.16)$$

Las ecuaciones de equilibrio (ver Tabla 2.1) para este sistema de colas son

$$\begin{aligned} \lambda p_0 &= \mu p_1, \\ (\lambda + n\mu)p_n &= \lambda p_{n-1} + (n+1)\mu p_{n+1}, \quad (1 \leq n \leq s-1), \\ (\lambda + s\mu)p_n &= \lambda p_{n-1} + s\mu p_{n+1}, \quad (n \geq s), \\ \sum_{n=0}^{\infty} p_n &= 1, \quad p_n \geq 0, \quad \forall n \geq 0, \end{aligned}$$

cuya solución es

$$p_n = \frac{\lambda_0 \cdots \lambda_{n-1}}{\mu_1 \cdots \mu_n} p_0 = \begin{cases} \left(\frac{\lambda}{\mu}\right)^n \frac{1}{n!} p_0, & n = 0, 1, \dots, s-1, \\ \left(\frac{\lambda}{s\mu}\right)^n \frac{s^s}{s!} p_0, & n \geq s. \end{cases} \quad (2.17)$$

Para encontrar p_0 , nuevamente hacemos uso de la ecuación (2.5), lo cual se llega a que

$$\begin{aligned}
 p_0 &= \left(\sum_{n=0}^{s-1} \left(\frac{\lambda}{\mu} \right)^n \frac{1}{n!} + \sum_{n=s}^{\infty} \left(\frac{\lambda}{s\mu} \right)^n \frac{s^n}{s!} \right)^{-1} \\
 &= \left(\sum_{n=0}^{s-1} \left(\frac{\lambda}{\mu} \right)^n \frac{1}{n!} + \left(\frac{\lambda}{\mu} \right)^s \frac{1}{s!} \frac{1}{1 - \frac{\lambda}{s\mu}} \right)^{-1}.
 \end{aligned}$$

Sea $r = \frac{\lambda}{\mu}$, y $\rho = \frac{\lambda}{s\mu}$ entonces

$$p_0 = \left(\sum_{n=0}^{s-1} \frac{r^n}{n!} + \frac{r^s}{s!(1-\rho)} \right)^{-1}, \quad (2.18)$$

y, para la estabilidad, tiene que ser $\rho < 1$. Si $s = 1$, (2.18) se reduce a la ecuación del modelo M/M/1.

Ahora podemos deducir las medidas de efectividad para el modelo M/M/s. Consideremos la longitud de la cola esperada L_q , ya que es más sencillo de calcular que L , debido a que sólo tenemos que trabajar con p_n para $n \geq s$. Puesto que $r = \frac{\lambda}{\mu}$, entonces

$$\begin{aligned}
 L_q &= \sum_{n=s+1}^{\infty} (n-s)p_n = \sum_{n=s+1}^{\infty} (n-s) \frac{r^n}{s^{n-s}s!} p_0 \\
 &= \frac{r^s p_0}{s!} \sum_{m=1}^{\infty} m \rho^m = \frac{r^s \rho p_0}{s!} \sum_{m=1}^{\infty} m \rho^{m-1} \\
 &= \frac{r^s \rho p_0}{s!} \frac{d}{d\rho} \sum_{m=1}^{\infty} \rho^m = \frac{r^s \rho p_0}{s!} \frac{d}{d\rho} \left(\frac{1}{1-\rho} - 1 \right) \\
 &= \frac{r^s \rho p_0}{s!(1-\rho)^2} = \frac{\left(\frac{\lambda}{\mu} \right)^s \rho p_0}{s!(1-\rho)^2}.
 \end{aligned}$$

Así,

$$\boxed{L_q = \left(\frac{\left(\frac{\lambda}{\mu} \right)^s \rho}{s!(1-\rho)^2} \right) p_0.} \quad (2.19)$$

Para encontrar L empleamos las Fórmulas de Little y se determina W_q , después usar W_q para encontrar $W = W_q + \frac{1}{\mu}$, y finalmente calculamos $L = \lambda W$. Entonces tenemos

$$\boxed{W_q = \frac{L_q}{\lambda} = \left(\frac{\left(\frac{\lambda}{\mu} \right)^s}{s!(s\mu)(1-\rho)^2} \right) p_0,} \quad (2.20)$$

$$W = \frac{1}{\mu} + \left(\frac{\left(\frac{\lambda}{\mu}\right)^s}{s!(s\mu)(1-\rho)^2} \right) p_0, \quad (2.21)$$

$$L = \frac{\lambda}{\mu} + \left(\frac{\left(\frac{\lambda}{\mu}\right)^s \rho}{s!(1-\rho)^2} \right) p_0. \quad (2.22)$$

La distribución del tiempo de espera en la cola, se obtiene si hacemos que T_q sea la variable que denote el tiempo de espera de un cliente en la cola, y $W_q(t)$ su función de distribución acumulada, tenemos

$$\begin{aligned} W_q(0) &= Pr(T_q = 0) = Pr(\leq s - 1 \text{ clientes en el sistema}) \\ &= \sum_{n=0}^{s-1} p_n = p_0 \sum_{n=0}^{s-1} \frac{r^n}{n!}. \end{aligned}$$

Utilizando (2.18) tenemos

$$\sum_{n=0}^{s-1} \frac{r^n}{n!} = \frac{1}{p_0} - \frac{r^s}{s!(1-\rho)}.$$

Por lo tanto

$$\begin{aligned} W_q(0) &= p_0 \left(\frac{1}{p_0} - \frac{r^s}{s!(1-\rho)} \right) \\ &= 1 - \frac{r^s p_0}{s!(1-\rho)}. \end{aligned} \quad (2.23)$$

Si $T_q > 0$,

$$\begin{aligned} W_q(t) &= Pr(T_q \leq t) \\ &= W_q(0) + \sum_{n=s}^{\infty} Pr(n - s + 1 \text{ servicios en } \leq t \mid \text{hay } n \text{ clientes}) p_n. \end{aligned}$$

Ahora, cuando $n \geq s$, la tasa de salidas del sistema es $s\mu$, así que el tiempo entre servicios sucesivos es una exponencial con media $\frac{1}{s\mu}$, y la distribución del tiempo para los $n - s + 1$ servicios es del tipo Erlang

$n - c + 1$. Entonces podemos escribir

$$\begin{aligned}
 W_q &= W_q(0) + p_0 \sum_{n=s}^{\infty} \frac{r^n}{s^{n-s} s!} \int_0^t \frac{s\mu(s\mu x)^{n-s}}{(n-s)!} e^{-s\mu x} dx \\
 &= W_q(0) + \frac{r^s p_0}{(s-1)!} \int_0^t \mu e^{-s\mu x} \sum_{n=s}^{\infty} \frac{(\mu r x)^{n-s}}{(n-s)!} dx \\
 &= W_q(0) + \frac{r^s p_0}{(s-1)!} \int_0^t \mu e^{-\mu x(s-r)} dx \\
 &= W_q(0) + \frac{r^s p_0}{s!(1-\rho)} (1 - e^{-(s\mu-\lambda)t}). \tag{2.24}
 \end{aligned}$$

Sustituyendo (2.23) en (2.24), se tiene

$$\boxed{W_q(t) = 1 - \frac{r^s p_0}{s!(1-\rho)} e^{-(s\mu-\lambda)t}},$$

la cual es la distribución del tiempo de espera en la cola.

En el siguiente capítulo se hará la simulación en MATLAB del sistema M/M/1, así como para el caso no-homogéneo; los códigos se incluyen en el Apéndice B.

Capítulo 3

Simulación

La simulación de un modelo probabilístico requiere la generación de los mecanismos estocásticos del modelo para después observar el flujo resultante de éste conforme transcurre el tiempo. Debido a que la evolución del modelo con el tiempo comprende a menudo una compleja estructura lógica de sus elementos, no siempre es fácil hallar la forma de seguirla como para determinar estas cantidades de interés. Así, para estos propósitos se ha elaborado un marco general, formulando en torno de la idea de “eventos discretos”. En tal marco surge el *método de simulación con eventos discretos*.

En este capítulo se darán algoritmos para la simulación del modelo $M/M/1$ (ver Sección 2.4), así como para el casos *no homogéneo* de dos modelos de colas, es decir para los sistemas en los cuales λ_n es distinto para toda n . Se realizó la simulación de estos modelos y fueron programados en MATLAB R2011b, los cuales se muestran en el Apéndice B. También se hará una comparación entre las medidas de rendimiento obtenidas en el Capítulo 2 con la efectividad de los programas simulados y dichas comparaciones se muestran en tablas.

3.1. Simulación Mediante Eventos Discretos

Los elementos fundamentales en una simulación por medio de eventos discretos son las variables y los eventos. Para realizar la simulación, mantenemos un seguimiento continuo de ciertas variables. En general, hay tres clases de variables que se utilizan con frecuencia: la variable de tiempo t , se refiere al tiempo simulado que ha transcurrido; las variables de conteo o contadores, estas variables mantienen un conteo del número de veces que ciertos eventos han ocurrido hasta el instante t ; y la variable de Estado del Sistema (**ES**), éste describe el “estado del sistema” en el instante t .

Siempre que ocurre un “evento”, los valores de estas variables se modifican o actualizan, y como salida reunimos los datos de interés. Para determinar el momento de ocurrencia del siguiente evento, se mantiene una “lista de eventos” que enumera los próximos eventos y el momento en que se espera que ocurra. Siempre que “ocurra” un evento, restablecemos la variable del tiempo, así como todas las variables de estado y contadores, y reunimos los datos importantes. De esta forma, podemos “seguir” el sistema conforme evoluciona con el tiempo.

3.2. Simulación del Sistema $M/M/1$

Esta sección muestra los resultados obtenidos después de varias ejecuciones del programa que simula un sistema de colas $M/M/1$ y éste se muestra en el Apéndice B.1 y se comparan con las medidas de rendimiento desarrolladas en la Sección 2.4.

En la Tabla 3.1 se muestran los resultados de doce simulaciones para distintos valores de λ y μ , por ejemplo, cuando $\lambda = 8/9$ y $\mu = 1$ se realizaron tres simulaciones para estos valores, dicha tabla muestra las medidas de rendimiento simuladas y sus valores reales, se puede observar que los resultados se acercan a sus valores reales conforme n aumenta.

Tabla 3.1					
Resultados de simulaciones <i>vs</i> valores reales					
valores λ, μ	simulación	medidas de rendimiento			
		L	L_q	W	W_q
$\lambda = 5/6$ $\mu = 1$	1ra. (n=100)	3.65	7.81	4.24	9.23
	2da. (n=1000)	5.24	6.40	5.79	8.23
	3ra. (n=10000)	5.22	3.94	5.82	5.71
	real	5	4.1667	6	5
$\lambda = 1$ $\mu = 2$	1ra. (n=100)	1.47	0.72	1.02	1.13
	2da. (n=1000)	1.45	0.49	0.97	0.99
	3ra. (n=10000)	1.40	0.51	0.99	0.88
	real	1	1/2	1	1/2
$\lambda = 8/9$ $\mu = 1$	1ra. (n=100)	4.39	11.85	5.00	12.16
	2da. (n=1000)	7.49	7.97	8.05	9.66
	3ra. (n=10000)	7.90	7.86	8.52	8.55
	real	8	7.11	9	8
$\lambda = 8$ $\mu = 11$	1ra. (n=100)	2.69	2.50	0.29	0.40
	2da. (n=1000)	2.67	2.05	0.30	0.34
	3ra. (n=10000)	2.66	1.98	0.33	0.33
	real	2.66	1.94	0.33	0.24

En el Apéndice B.1, se muestran dos códigos; el primero calcula el promedio de la longitud de la cola y el promedio del tiempo de espera en la cola, el segundo calcula el valor esperado de la longitud en todo el sistema y el promedio que esperaría un cliente en el sistema. Ambos programas arrojan un gráfico el cual muestra la evolución de la cola y del sistema a través del tiempo, respectivamente.

La Figura 3.1 muestra el resultado que genera la simulación con valores $\lambda = 8$ y $\mu = 11$ (con $n = 100$) para la longitud del sistema. La Figura 3.2 muestra la evolución del sistema a través del tiempo con $\lambda = 1$ y $\mu = 2$ (para $n = 100$).

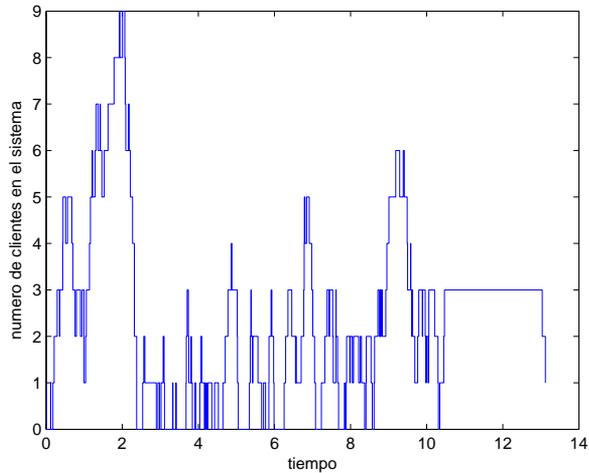


Figura 3.1: Número de clientes en el sistema

Como se puede observar en la Tabla 3.1 el valor real de L es 2.66 para $\lambda = 8$ y $\mu = 11$, esta evolución se muestra en la Figura 3.1 con ($n = 100$). Similarmente para valores de $\lambda = 1$ y $\mu = 2$, $L = 1$ (ver Tabla 3.1), se observa en la Figura 3.2 que la evolución del número de clientes en el sistema se concentra en 1 ($n = 100$).

3.3. Simulación de Sistemas no Homogéneos

En los siguientes modelos de colas, se supondrá que los clientes llegan de acuerdo a un proceso Poisson no homogéneo con una función de intensidad $\lambda(t)$ acotada, con $t > 0$. Al simular estos modelos, aplicamos la siguiente subrutina para generar el valor de una variable aleatoria T_s , definida como el tiempo de la primera llegada después del instante s .

Sea λ tal que $\lambda(t) \leq \lambda$ para toda t . Suponiendo dados $\lambda(t)$, $t > 0$ y λ , la siguiente subrutina genera el valor de T_s , (ver [3]).

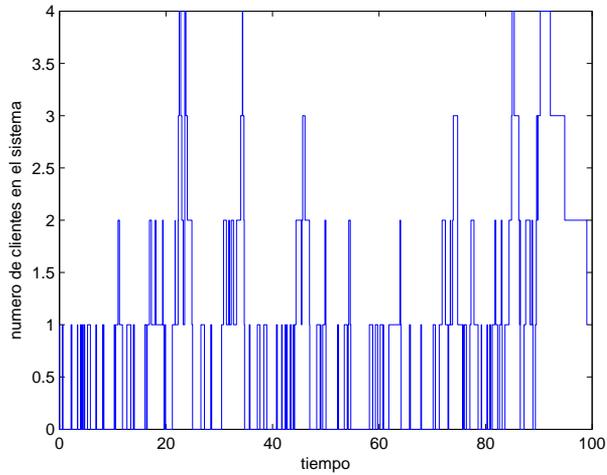


Figura 3.2: Número de clientes en el sistema

Una subrutina para generar T_s

PASO 1: Sea $t = s$.

PASO 2: Generar U .

PASO 3: Sea $t = t - \frac{1}{\lambda} \log U$.

PASO 4: Generar U .

PASO 5: Si $U \leq \lambda(t)/\lambda$, hacer $T_s = t$ y terminar.

PASO 6: Ir al paso 2.

En el Apéndice B.2 se muestra el código para generar dicha subrutina.

3.3.1. Sistema de Línea de Espera con un Servidor

Consideremos una estación de servicio a la cual los clientes llegan de acuerdo con un proceso Poisson no homogéneo con función de intensidad $\lambda(t)$, $t \geq 0$. Hay un único servidor, y al llegar un cliente pasa a servicio si el servidor está libre en ese momento, o bien se une a la fila de espera si está ocupado. Cuando el servidor termina de dar servicio a un cliente,

se ocupa del cliente que ha estado esperando más tiempo (la disciplina “primero en llegar, primero en atender”) si hay clientes esperando, o si no los hay, permanece libre hasta la llegada del siguiente cliente. El tiempo que tarda la atención a un cliente es una variable aleatoria (independiente de los demás tiempos de servicio y del proceso de llegada) con distribución de probabilidad tipo exponencial. Además, hay un tiempo fijo T después del cual no se permite que otras llegadas entren al sistema, aunque el servidor atiende a todos los que ya estén dentro del sistema en el instante T .

En esta sección estamos interesados en simular el sistema descrito anteriormente para determinar el tiempo promedio que pasa un cliente dentro del sistema *no-homogéneo*. Para simularlo utilizamos las siguientes variables:

Variable de tiempo	t
Variables de conteo	N_A : el número de llegadas hasta t N_D : el número de salidas hasta t
Variable de ES	n : el número de clientes en sistema

Puesto que el instante para modificar las cantidades anteriores es cuando ocurre una llegada o una salida, las consideraremos “eventos”; es decir, hay dos tipos de eventos: llegadas y salidas. La lista de eventos contiene el instante de la siguiente llegada y el instante de la salida del cliente que se encuentra en servicio. Es decir, la lista de eventos (**LE**) es

$$\mathbf{LE} = (t_A, t_D)$$

donde t_A es la hora de la siguiente llegada y t_D es la hora a la que concluye el servicio del cliente que se está atendiendo actualmente. Si no hay clientes en servicio, t_D es igual a ∞ .

Las variables de salida por registrar son $A(i)$, la hora de llegada del cliente i ; $D(i)$, la hora de salida del cliente i , y T_p , el tiempo de salida del último cliente, posterior a T .

Para comenzar la simulación, inicializamos las variables y los tiempos

de los eventos como sigue:

Inicialización

Sean $t = N_A = N_D = 0$.

ES = 0.

Generar T_0 , y hacer $t_A = T_0$, $t_D = \infty$.

Para actualizar el sistema, nos movemos a lo largo del eje del tiempo hasta encontrar el siguiente evento. Para ello, debemos considerar varios casos, dependiendo de cuál sea el menor elemento de la lista de eventos. En adelante, Y se referirá a una variable aleatoria, con distribución exponencial, para el tiempo de servicio.

t = variable de tiempo, **ES** = n , **LE** = t_A , t_D

Caso 1 $t_A \leq t_D$, $t_A \leq T$

Restablecer: $t = t_A$ (nos movemos hasta el tiempo t_A).

Restablecer: $N_A = N_A + 1$ (pues hay una llegada adicional en el instante t_A).

Restablecer: $n = n + 1$ (pues ahora se tiene un cliente más).

Generar T_t , y hacer $t_A = T_t$ (ésta es la hora de la siguiente llegada).

Si $n = 1$, generar Y y hacer $t_D = t + Y$ (pues el sistema ha quedado vacío, por lo cual necesitamos generar el tiempo de servicio del nuevo cliente).

Reunir los datos de salida $A(N_A) = t$ (pues el cliente N_A llega en el instante t).

Caso 2 $t_D \leq t_A$, $t_D \leq T$

Restablecer: $t = t_D$.

Restablecer: $n = n - 1$.

Restablecer: $N_D = N_D + 1$ (pues ha ocurrido una salida en el instante t).

Si $n = 0$, hacer $t_D = \infty$; en caso contrario, generar Y y hacer $t_D = t + Y$.

Reunir los datos de salida $D(N_D) = t$ (pues el cliente N_D acaba de salir).

Caso 3 $\min(t_A, t_D) > T, n > 0$

Restablecer: $t = t_D$.

Restablecer: $n = n - 1$.

Restablecer: $N_D = N_D + 1$.

Si $n > 0$, generar Y y hacer $t_D = t + Y$.

Reunir los datos de salida $D(N_D) = t$.

Caso 4 $\min(t_A, t_D) > T, n = 0$

Reunir los datos de salida $T_p = \max(t - T, 0)$.

Este proceso se ilustra en la Figura 3.3. Cada vez que llegamos a una caja “terminar” habremos obtenido los datos N_A , el número total de llegadas, que será igual a N_D , el número total de salidas. Para cada i , $i = 1, \dots, N_A$, tenemos A_i y D_i , las horas de llegada y salida del cliente i , respectivamente (y así $D_i - A_i$ representa la cantidad de tiempo que el cliente i ha estado dentro del sistema). Por último, tenemos T_p , el tiempo posterior a T en que sale el último cliente. Cada vez que se reúnen los datos anteriores, se dice que se ha concluido una ejecución de la simulación. Después de cada ejecución, se reinicializa y se genera otra ejecución hasta decidir que se cuenta con los datos suficientes. El promedio de todos los valores de T_p generados de esta forma será nuestra estimación del tiempo promedio posterior a T en que saldrá el último cliente; de manera análoga, el promedio de todos los valores observados de $D - A$ (es decir, el tiempo promedio que un cliente pasa en el sistema, sobre todos los clientes observados en las ejecuciones de simulación) será la estimación del tiempo promedio que un cliente pasa en el sistema.

En el Apéndice B.3 se muestra el código que simula un sistema de línea de espera no homogéneo. En este apéndice se dan los valores de $\lambda = 1$ y $\mu = 2$, mas estos parámetros pueden tomar valores arbitrarios. Una vez ejecutado el programa en MATLAB R2011b, éste imprime las llegadas de cada cliente, así como las salidas de cada uno de ellos y por

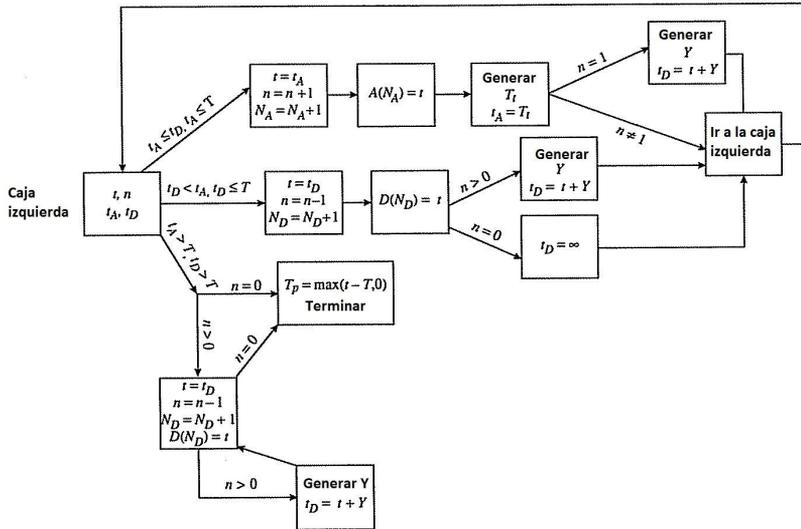


Figura 3.3: Diagrama de operaciones

último imprime el tiempo promedio que un cliente pasa en el sistema.

3.3.2. Sistema de Línea de Espera con Dos Servidores en Paralelo

Para el caso de un modelo de colas $M/M/s$ (ver Sección 2.5) vamos a tomar $s = 2$, es decir tomar un sistema de colas $M/M/2$ y simular este modelo. Así, se considera un modelo en el que los clientes llegan a un sistema con dos servidores. Al llegar, un cliente se forma en una fila si ambos servidores están ocupados, entra a servicio con el servidor 1 si ese servidor está desocupado, o bien con el servidor 2 en caso contrario. Cuando el cliente concluye el servicio con un servidor (sin importar cuál sea), sale del sistema y el cliente que ha estado formado más tiempo (si hay clientes en la cola) entra a servicio. La distribución de servicio se supone de tipo exponencial (ver Figura 3.4).

Lo que se quiere es simular el modelo anterior llevando un registro de las cantidades de tiempo que pasa cada cliente dentro del sistema y

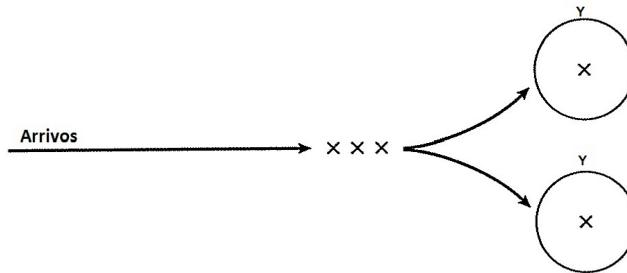


Figura 3.4:

el número de servicios realizados por cada servidor. En este caso, como en número de servidores es dos, los clientes no saldrán en el orden de llegada. Por lo tanto, para saber qué cliente deja el sistema al concluir su servicio, necesitamos llevar un registro para ver qué clientes están en el sistema. Así, se numerarán conforme vayan llegando, de modo que el primero en llegar es el cliente número 1, el siguiente es el número 2, y así sucesivamente.

Para analizar el sistema se utilizarán las siguientes variables:

Variable de tiempo t

Variable de Estado del Sistema (ES)

(n, i_1, i_2) si hay n clientes en el sistema, i_1 está con el servidor 1 y i_2 con el servidor 2. Note que **ES** = (0) cuando el sistema se encuentra vacío, y **ES** = $(1, j, 0)$ o $(1, 0, j)$ cuando el único cliente es j y está siendo atendido por el servidor 1 o el servidor 2, respectivamente.

Variables de Conteo

N_A : número de arriros hasta el tiempo t .

C_j : número de clientes atendidos por j , $j = 1, 2$ hasta el tiempo t .

Variables de Salida

$A(n)$: tiempo en que llega el cliente n , $n \geq 1$.

$D(n)$: tiempo de salida del cliente n , $n \geq 1$.

Lista de eventos t_A, t_1, t_2

donde t_A es el tiempo de la próxima llegada, y t_i es el tiempo de culminación de servicio del cliente presente que fue atendido por el servidor i , $i = 1, 2$. Si no hay cliente presente con el servidor i , entonces asignamos $t_i = \infty$, $i = 1, 2$.

Para empezar la simulación, inicializamos las variables y la lista de eventos como sigue:

Inicializar

Hacer $t = N_A = C_1 = C_2 = 0$.

Hacer $\mathbf{ES} = (0)$.

Generar T_0 , y hacer $t_A = T_0$, $t_1 = t_2 = \infty$.

Para actualizar el sistema, nos movemos a través del tiempo hasta encontrar el siguiente evento. En adelante, Y siempre se referirá a una variable aleatoria con distribución exponencial.

Caso 1: $\mathbf{ES} = (n, i_1, i_2)$ y $t_A = \min(t_A, t_1, t_2)$

Restablecer: $t = t_A$.

Restablecer: $N_A = N_A + 1$.

Generar T_t y restablecer $t_A = T_t$.

Reunir los datos $A(N_A) = t$.

Si $\mathbf{ES} = (0)$:

Restablecer: $\mathbf{ES} = (1, N_A, 0)$.

Generar Y y restablecer $t_1 = t + Y$.

Si $\mathbf{ES} = (1, j, 0)$:

Restablecer: $\mathbf{ES} = (2, j, N_A)$.

Generar Y y restablecer $t_2 = t + Y$.

Si $\mathbf{ES} = (1, 0, j)$:

Restablecer: $\mathbf{ES} = (2, N_A, j)$.

Generar Y y restablecer $t_1 = t + Y$.

Si $n > 1$:

Restablecer: $\mathbf{ES} = (n + 1, i_1, i_2)$.

Caso 2: $\mathbf{ES} = (n, i_1, i_2)$ y $t_1 < t_A$, $t_1 \leq t_2$

Restablecer: $t = t_1$.

Restablecer: $C_1 = C_1 + 1$.

Reunir los datos $D(i_1) = t$.

Si $n = 1$:

Restablecer: $\mathbf{ES} = (0)$.

Restablecer: $t_1 = \infty$.

Si $n = 2$:

Restablecer: $\mathbf{ES} = (1, 0, i_2)$.

Restablecer: $t_1 = \infty$.

Si $n > 2$: Sea $m = \max(i_1, i_2)$ y

Restablecer: $\mathbf{ES} = (n - 1, m + 1, i_2)$.

Generar Y y restablecer $t_1 = t + Y$.

Caso 3: $\mathbf{ES} = (n, i_1, i_2)$ y $t_2 < t_A$, $t_2 < t_1$

Restablecer: $t = t_2$.

Restablecer: $C_2 = C_2 + 1$.

Reunir los datos $D(i_2) = t$.

Si $n = 1$:

Restablecer: $\mathbf{ES} = (0)$.

Restablecer: $t_2 = \infty$.

Si $n = 2$:

Restablecer: $\mathbf{ES} = (1, i_1, 0)$.

Restablecer: $t_2 = \infty$.

Si $n > 2$: Sea $m = \max(i_1, i_2)$ y

Restablecer: $\mathbf{ES} = (n - 1, i_1, m + 1)$.

Generar Y y restablecer $t_2 = t + Y$.

Cuando se simula el sistema de acuerdo a lo anterior, y teniendo un tiempo de paro, entonces usando las variables de salida así como los valores finales de las variables C_1 y C_2 , se obtienen los datos de los tiempos de llegadas y salidas de los clientes así como el número de servicios que realizó cada servidor.

En el Apéndice B.4 se muestra el código que simula un sistema de línea de espera *no-homogéneo* con dos servidores en paralelos. Los

parámetros pueden tomar valores arbitrarios. Una vez ejecutado el programa en MATLAB R2011b, éste imprime las llegadas de cada cliente, así como las salidas de cada uno de ellos y por último muestra el número de servicios de cada servidor.

Capítulo 4

Procesos de Control Semi-Markovianos

Consideremos un sistema de colas $M/M/1$ con periodos de recesos permitidos en el servidor, donde el servidor es apagado tan pronto la cola se vacía. Suponemos que las duraciones de estos periodos forman una secuencia de variables aleatorias independientes e idénticamente distribuidas (v.a.i.i.d.) con distribución exponencial. Al final de un período de receso, el servidor puede o bien ser activado si la cola no está vacía, o tomar nuevamente otro período de receso. En el sistema se incurrirán los siguientes costos: un costo de almacenamiento h por unidad de tiempo y por cliente en el sistema y un costo fijo γ cada vez que el servidor se vuelve a activar. Se mostrará que existe una política umbral que minimiza el criterio de costo total esperado α -descontado.

Los objetivos principales de este capítulo son: presentar formalmente el Modelo de Control Semi-Markoviano (MCSM) y, demostrar que existe una política umbral que minimiza el criterio de costo total esperado α -descontado en el sistema $M/M/1$, usando Programación Dinámica.

4.1. Modelo de Control Semi-Markoviano

Definición 4.1 *Un Modelo de Control Semi-Markoviano (MCSM)*

$$\left(\mathbb{S}, \mathbb{A}, \{\mathbb{A}(s) : s \in \mathbb{S}\}, Q, H, D, d \right) \quad (4.1)$$

consiste en lo siguiente:

- \mathbb{S} es un espacio de Borel (ver A.1) llamado espacio de estados.
- \mathbb{A} es un espacio de Borel llamado espacio de controles o acciones.
- Para cada $s \in \mathbb{S}$, $\mathbb{A}(s) \subset \mathbb{A}$ es un conjunto medible y no vacío, cuyos elementos representan las acciones admisibles cuando el sistema se encuentre en el estado s .

Observación: Se supone que el conjunto $\mathbb{K} := \{(s, a) : s \in \mathbb{S}, a \in \mathbb{A}(s)\}$ de pares estado-acción admisible, es un subconjunto de Borel del espacio $\mathbb{S} \times \mathbb{A}$. También se supone que \mathbb{K} contiene a la gráfica de una función medible $f : \mathbb{S} \rightarrow \mathbb{A}$, de manera que $f(s) \in \mathbb{A}$, para todo $s \in \mathbb{S}$.

- La ley de transición $Q(\cdot | \cdot)$ es un kernel estocástico (véase A.1) sobre \mathbb{S} dado \mathbb{K} .
- $H(\cdot | s, a)$ es la función de distribución del tiempo de permanencia sobre \mathbb{R} , para cada $(s, a) \in \mathbb{K}$.
- D y d son funciones de costo, las cuales son medibles sobre \mathbb{K} .

Un MCSM representa un sistema dinámico que evoluciona de la siguiente manera. En el tiempo de la n -ésima época de decisión, t_n , el sistema se encuentra en el estado $s_n = s$ y el controlador elige una acción $a_n = a \in \mathbb{A}(s)$, generándose con ello lo que se describe a continuación:

- Se incurre un costo inmediato $D(s, a)$.
- El sistema permanece en dicho estado $s_n = s$ durante un tiempo aleatorio no-negativo δ_{n+1} con distribución $H(\cdot | s, a)$.

- En el tiempo $t_{n+1} := t_n + \delta_{n+1}$ ($n \in \mathbb{N}$, $t_0 := 0$), el sistema transita a un nuevo estado $s_{n+1} = s'$ de acuerdo a la distribución $Q(\cdot | s, a)$.
- Se produce un costo debido al tiempo de permanencia en el estado s , cuya razón de costo es $d(s, a)$.
- Finalmente, una vez en el estado s' el proceso se repite.

Observación 4.1 *Se identificará al conjunto de los enteros no-negativos como \mathbb{N} y a \mathbb{N}^* como el conjunto de todos los enteros estrictamente positivos.*

4.2. Políticas de Control

Sea un MCSM fijo, se define el espacio de historias admisibles hasta la n -ésima época de decisión mediante

$$\mathbb{H}_0 := \mathbb{S}$$

y

$$\mathbb{H}_n := (\mathbb{K} \times \mathbb{R}_+)^n \times \mathbb{S} \text{ para } n \in \mathbb{N}^*.$$

De lo anterior se desprende que un elemento $h_n \in \mathbb{H}_n$, llamado n -historia, es un vector de la forma,

$$h_n := (s_0, a_0, \delta_1, s_1, a_1, \delta_2, \dots, s_{n-1}, a_{n-1}, \delta_n, s_n),$$

donde $(s_k, a_k, \delta_{k+1}) \in \mathbb{K} \times \mathbb{R}_+$ para $k = 0, 1, 2, \dots, n-1$ y $s_n \in \mathbb{S}$.

Definición 4.2 *Una política de control admisible o simplemente una política es una sucesión $\pi := \{\pi_n\}$, donde cada π_n es un kernel estocástico sobre \mathbb{A} dado \mathbb{H}_n , tales que satisfacen la restricción $\pi_n(\mathbb{A}(x) | h_n) = 1$ para todo $h_n \in \mathbb{H}_n$ y $n \in \mathbb{N}^*$.*

Denotaremos por Π al conjunto de todas las políticas.

Sea \mathbb{F} el conjunto de todas las funciones $f : \mathbb{S} \rightarrow \mathbb{A}$, tal que $f(s) \in \mathbb{A}(s)$ para todo $s \in \mathbb{S}$.

Se denotará a la familia de k erneos estoc astico sobre \mathbb{A} dado \mathbb{S} , como $P(\mathbb{A} \mid \mathbb{S})$.

Sea Φ el conjunto de todos los k erneos estoc asticos φ en $P(\mathbb{A} \mid \mathbb{S})$ tales que para toda $s \in \mathbb{S}$ se tiene que $\varphi(\mathbb{A}(s) \mid s) = 1$.

Observaci on: Se dice que $\pi(\cdot \mid h)$ est a concentrada en $g(h)$, si $\pi(C \mid h) = I_C(g(h))$ para cada $C \in \mathfrak{B}(\mathbb{A})$ (ver Ap endice A.1). Donde I_C es la funci on indicadora del conjunto C .

Definici on 4.3 Una pol itica $\pi := \{\pi_n\} \in \Pi$ es:

- **Markoviana Aleatorizada** (Π_{RM}). Si existe una sucesi on $\{\varphi_n\}$ de k erneos estoc asticos con $\varphi_n \in \Phi$ (definidas sobre \mathbb{A} dado \mathbb{S}), tales que, $\pi_n(\cdot \mid h_n) = \varphi_n(\cdot \mid s_n)$ para toda $h_n \in \mathbb{H}_n$ y $n \in \mathbb{N}$.
- **Markoviana Aleatorizada Estacionaria** (Π_{RS}). Si existe $\varphi \in \Phi$ k eruel estoc astico, tal que $\pi_n(\cdot \mid h_n) = \varphi(\cdot \mid s_n)$ para toda $h_n \in \mathbb{H}_n$ y $n \in \mathbb{N}$.
- **Determinista** (Π_D). Si existe una sucesi on $\{g_n\}$ de funciones medibles con $g_n : \mathbb{H}_n \rightarrow \mathbb{A}$, tales que, para cada $h_n \in \mathbb{H}_n$ y $n \in \mathbb{N}$, se tiene que $g_n(h_n) \in \mathbb{A}(s_n)$ y $\pi_n(\cdot \mid h_n)$ est a concentrada en $g_n(h_n)$.
- **Determinista Markoviana** (Π_{DM}). Si existe una sucesi on $\{f_n\}$ de funciones medibles $f_n : \mathbb{S} \rightarrow \mathbb{A}$ (o $f \in \mathbb{F}$), tales que $f_n(s_n) \in \mathbb{A}(s_n)$ y $\pi_n(\cdot \mid h_n)$ est a concentrada en $f_n(s_n)$ para cada $h_n \in \mathbb{H}_n$ y $n \in \mathbb{N}$.
- **Determinista Markoviana Estacionaria** (Π_{DS}). Si existe una funci on medible $f : \mathbb{S} \rightarrow \mathbb{A}$ (o $f \in \mathbb{F}$), tal que $f(s_n) \in \mathbb{A}(s_n)$ y $\pi_n(\cdot \mid h_n)$ est a concentrada en $f(s_n)$ para cada $h_n \in \mathbb{H}_n$ y $n \in \mathbb{N}$.

Observaci on: Note que $\Pi_{RS} \subset \Pi_{RM} \subset \Pi$ y $\Pi_{DS} \subset \Pi_{DM} \subset \Pi_D \subset \Pi$.

Sin p erdida de generalidad, en adelante se identificar a al conjunto de todas las pol iticas estacionarias con \mathbb{F} .

Sea (Ω, \mathcal{F}) el espacio medible canónico en el cual Ω es el espacio producto $(\mathbb{K} \times \mathbb{R}_+)^{\infty}$ y \mathcal{F} la σ -álgebra producto correspondiente. Se observa que un elemento $\omega \in \Omega$ tiene la forma $\omega = (s_0, a_0, \delta_1, s_1, a_1, \delta_2, \dots)$.

A las variables $s_k \in \mathbb{S}$, $a_k \in \mathbb{A}(s_k)$ y $\delta_k \in \mathbb{R}_+$, se les llamará variables de estado, control y tiempo de transición, respectivamente.

De acuerdo al Teorema de C. Ionescu-Tulcea (ver Apéndice A.2), para cada $s \in \mathbb{S}$ y cada $\pi \in \Pi$, existe una medida de probabilidad P_s^π sobre (Ω, \mathcal{F}) . Además, para cada $B \in \mathfrak{B}(\mathbb{A})$, $C \in \mathfrak{B}(\mathbb{S})$, $h_n \in \mathbb{H}_n$ y $n \in \mathbb{N}$ se tiene que:

$$P_s^\pi(s_0 = s) = 1, \quad (4.2)$$

$$P_s^\pi(a_n \in B \mid h_n) = \pi_n(B \mid h_n), \quad (4.3)$$

$$P_s^\pi(s_{n+1} \in C \mid h_n, a_n, \delta_{n+1}) = Q(C \mid s_n, a_n), \quad (4.4)$$

$$P_s^\pi(\delta_{n+1} \leq t \mid h_n, a_n) = H(t \mid s_n, a_n). \quad (4.5)$$

Observación 4.2 De (4.5) se tiene que las v.a.'s $\delta_1, \delta_2, \dots$, son condicionalmente independientes dado el proceso $s_0, a_0, s_1, a_1, \dots$.

Para una política arbitraria $\pi \in \Pi$, la variable s_n describe el estado del sistema en el tiempo de la n -ésima transición (o época de decisión) t_n y a_n representa el control elegido de acuerdo a la política π .

Nótese que en general, el estado s_n depende de la evolución del sistema en las primeras $n - 1$ transiciones, no obstante en el caso de una política estacionaria f , $\{s_n\}_{n=0}^{\infty}$ es una cadena de Markov con probabilidad de transición $Q(\cdot \mid s, f(s))$, lo cual es una consecuencia de las propiedades de esperanza condicional, así como de (4.4) que en lo sucesivo se identificará como la *propiedad de Markov*.

Se denotará por E_s^π al operador esperanza correspondiente a P_s^π . Además, para una política estacionaria $f \in \mathbb{F}$, se usará la notación (x, f) en lugar de $(x, f(x))$.

4.3. Índice de Funcionamiento y Problema de Control Óptimo con Costo Descontado

Cada MCSM está dotado de una función real, llamada función objetivo o criterio de rendimiento, cuyo fin es medir el comportamiento de los costos por etapa, para ello se usará como índice de funcionamiento el *costo total esperado α -descontado*.

Consideremos un MCSM fijo y un conjunto de políticas Π , tenemos la siguiente

Definición 4.4 *Se define el costo total esperado α -descontado, para cada $s \in \mathbb{S}$, $\pi \in \Pi$ y $\alpha > 0$ como*

$$\begin{aligned} V_\alpha^\pi(s) &:= E^\pi \left[\sum_{n \in \mathbb{N}} e^{-\alpha t_n} c(s_n, a_n) \mid s_0 = s \right] \\ &= E_s^\pi \left[\sum_{n \in \mathbb{N}} e^{-\alpha t_n} c(s_n, a_n) \right], \end{aligned}$$

donde $c(\cdot, \cdot)$ es una función de costo.

Definición 4.5 *Una política $\pi^* \in \Pi$ es óptima si*

$$V_\alpha^{\pi^*}(s) := \inf_{\pi \in \Pi} V_\alpha^\pi(s), \quad \forall s \in \mathbb{S}.$$

Definición 4.6 *Se define para cada $s \in \mathbb{S}$*

$$V_\alpha^*(s) := \inf_{\pi \in \Pi} V_\alpha^\pi(s),$$

V_α^* se llama *función de valores óptimos o valor óptimo*.

Dado el MCSM, el problema de control óptimo (PCO) consiste en encontrar una política óptima, es decir, minimizar la función $\pi \rightarrow V_\alpha^\pi(s)$ sobre Π , para todo $s \in \mathbb{S}$.

4.3.1. Condiciones Sobre el Modelo de Control

Supongamos que \mathbb{S} y \mathbb{A} son subconjuntos numerables de \mathbb{R} . Para resolver el PCO, se requiere que el modelo de control satisfaga un conjunto de condiciones que se darán a continuación.

Condición 4.1 *Condiciones de continuidad y compacidad.*

Sean $s, s' \in \mathbb{S}$

- a) $\mathbb{A}(s)$ es un conjunto compacto, para cada $s \in \mathbb{S}$,
- b) Las funciones de costo, $D(s, a)$ y $d(s, a)$ son reales medibles y continuas en $a \in \mathbb{A}(s)$, para cada $s \in \mathbb{S}$,
- c) La ley de transición $Q(\cdot | s, a)$ es fuertemente continua en $a \in \mathbb{A}(s)$, es decir, para cada función medible y acotada v sobre \mathbb{S} , se tiene que,

$$a \mapsto \sum_{s' \in \mathbb{S}} v(s')Q(s' | s, a),$$

es una función continua y acotada sobre $\mathbb{A}(s)$,

- d) La función $H(t | s, a)$ es continua en $a \in \mathbb{A}(s)$ para cada $s \in \mathbb{S}$ y $t \in \mathbb{R}$.

Las condiciones de crecimiento nos permiten analizar el PCO con costos no acotados. Sea $w : \mathbb{S} \rightarrow [1, \infty)$ una función medible. Se denota por $\mathbb{B}_w(\mathbb{S})$ el espacio lineal normado el cual consiste de todas las funciones medibles $u : \mathbb{S} \rightarrow \mathbb{R}$, tales que satisfacen la condición,

$$\| u \|_w := \sup_{s \in \mathbb{S}} \frac{|u(s)|}{w(s)} < \infty. \quad (4.6)$$

Condición 4.2 *Condiciones de crecimiento.*

- a) Existe una función medible $w : \mathbb{S} \rightarrow [1, \infty)$, así como constantes $c_1, c_2 > 0$, $\varepsilon \in (0, 1)$ y $b > 0$ tales que

$$\sup_{a \in \mathbb{A}(s)} |D(s, a)| \leq c_1 w(s), \quad \sup_{a \in \mathbb{A}(s)} |d(s, a)| \leq c_2 w(s), \quad (4.7)$$

y

$$\sum_{s' \in \mathbb{S}} w(s')Q(s' | s, a) \leq \varepsilon w(s) + b, \quad s \in \mathbb{S}, a \in \mathbb{A}(s). \quad (4.8)$$

b) La función $a \rightarrow \sum_{s' \in \mathbb{S}} w(s')Q(s' | s, a)$, es continua en $\mathbb{A}(s)$, para cada $s \in \mathbb{S}$.

Lema 4.1 *Mediante la Condición 4.2, para cada $\pi \in \Pi$ y $s \in \mathbb{S}$ se cumple que*

$$\sup_{n \in \mathbb{N}} E_s^\pi[w(s_n)] < \infty.$$

Demostración. Sea $s \in \mathbb{S}$, $\pi \in \Pi$, de (4.4) y (4.8) se tiene que

$$E_s^\pi[w(s_n) | h_{n-1}, a_{n-1}, \delta_n] = \sum_{s' \in \mathbb{S}} w(s')Q(s' | s_{n-1}, a_{n-1}) \leq \varepsilon w(s_{n-1}) + b.$$

Al tomar esperanzas en ambos lados de la expresión anterior se obtiene que

$$E_s^\pi[w(s_n)] \leq \varepsilon E_s^\pi[w(s_{n-1})] + b, \quad (4.9)$$

iterando (4.9) resulta

$$\begin{aligned} E_s^\pi[w(s_n)] &\leq \varepsilon^n E_s^\pi[w(s_0)] + b(1 + \varepsilon + \varepsilon^2 + \dots + \varepsilon^{n-1}) \\ &= \varepsilon^n E_s^\pi[w(s_0)] + b \sum_{k=0}^{n-1} \varepsilon^k. \end{aligned}$$

Puesto que $\varepsilon \in (0, 1)$, tenemos

$$E_s^\pi[w(s_n)] \leq w(s) + \frac{b}{1 - \varepsilon} < \infty, \quad \forall n \in \mathbb{N}.$$

Además, como $w(\cdot) \geq 1$,

$$E_s^\pi[w(s_n)] \leq \left(1 + \frac{b}{1 - \varepsilon}\right) w(s) < \infty, \quad \forall n \in \mathbb{N}. \quad (4.10)$$

Por consiguiente,

$$\sup_{n \in \mathbb{N}} E_s^\pi[w(s_n)] < \infty, \quad \forall n \in \mathbb{N}. \quad \blacksquare$$

La siguiente condición garantiza que los PCSM's son regulares, esto es, que experimentan un número finito de transiciones en intervalos de tiempo acotados, este hecho se ve en la Proposición 4.1.

Condición 4.3 *Condición de regularidad.*

Existen reales $\varepsilon > 0$ y $\theta > 0$ tales que

$$H(\theta \mid s, a) \leq 1 - \varepsilon, \quad \forall (s, a) \in \mathbb{K}.$$

Notación. Para cada $(s, a) \in \mathbb{K}$, definimos

$$\Delta_\alpha(s, a) := \int_0^\infty e^{-\alpha t} H(dt \mid s, a). \quad (4.11)$$

Observación. Bajo la Condición 4.1 se tiene que la función $\Delta_\alpha(s, \cdot)$ es continua.

Los siguientes resultados serán de utilidad posteriormente.

Proposición 4.1 *Si se cumple la Condición 4.3, entonces:*

- a) $\rho_\alpha := \sup_{\mathbb{K}} \Delta_\alpha(s, a) < 1$,
- b) $P_s^\pi[\sum_{n=1}^\infty \delta_n = \infty] = 1, \quad \forall s \in \mathbb{S}, \pi \in \Pi$.

Demostración. Véase, por ejemplo, [22]. ■

Lema 4.2 *Sea $s \in \mathbb{S}$, supóngase que $\mathbb{A}(s)$ es compacto (Condición 4.1(a)) y sea $v : \mathbb{K} \rightarrow \mathbb{R}$ una función medible.*

- (a) *Si $v(s, a)$ es continua en $a \in \mathbb{A}(s)$ para $s \in \mathbb{S}$, entonces*

$$v^*(s) := \inf_{a \in \mathbb{A}(s)} v(s, a),$$

es medible, y además existe $f^ \in \mathbb{F}$ tal que*

$$v^*(s) = v(s, f^*) = \min_{a \in \mathbb{A}(s)} v(s, a), \quad \forall s \in \mathbb{S}.$$

- (b) *Sean v y v_n ($n \in \mathbb{N}$) funciones medibles en \mathbb{K} y continuas sobre $\mathbb{A}(s)$, tales que $v_n \uparrow v$. Entonces para cada $s \in \mathbb{S}$,*

$$\min_{a \in \mathbb{A}(s)} v_n(s, a) \rightarrow \min_{a \in \mathbb{A}(s)} v(s, a), \quad \text{cuando } n \rightarrow \infty$$

Demostración. Para una demostración ver [25]. ■

4.4. Criterio de Optimalidad con Costo Descontado

En esta sección se muestra la existencia de políticas óptimas estacionarias bajo las Condiciones 4.1-4.3, ésto se consigue (en el Teorema 4.1) mostrando que la función de valor óptimo satisface la ecuación de optimalidad.

El siguiente resultado muestra una forma alternativa de expresar el costo total esperado α -descontado (ver Definición 4.4).

Proposición 4.2 *Para todo $s \in \mathbb{S}$ y $\pi \in \Pi$ se tiene que*

$$V_{\alpha}^{\pi}(s) = E_s^{\pi} \left[c(s_0, a_0) + \sum_{n=1}^{\infty} \prod_{j=0}^{n-1} \Delta_{\alpha}(s_j, a_j) c(s_n, a_n) \right],$$

donde $\Delta_{\alpha}(\cdot, \cdot)$ se introdujo en (4.11).

Demostración. De la definición de $\Delta_{\alpha}(s, a)$ puede observarse que

$$\begin{aligned} & E_s^{\pi} \left[c(s_0, a_0) + \sum_{n=1}^{\infty} \prod_{k=0}^{n-1} \Delta_{\alpha}(s_k, a_k) c(s_n, a_n) \right] \\ &= E_s^{\pi} \left[c(s_0, a_0) + \sum_{n=1}^{\infty} \prod_{k=0}^{n-1} \int_0^{\infty} e^{-\alpha t} H(dt \mid s_k, a_k) c(s_n, a_n) \right] \\ &= E_s^{\pi} \left[c(s_0, a_0) + \sum_{n=1}^{\infty} \prod_{k=0}^{n-1} E_s^{\pi} \left[e^{-\alpha \delta_{k+1}} \mid s_k, a_k \right] c(s_n, a_n) \right]. \end{aligned}$$

Luego, de la Observación 4.2 note que

$$\begin{aligned} \prod_{k=0}^{n-1} E_s^{\pi} [e^{-\alpha \delta_{k+1}} \mid s_k, a_k] &= E_s^{\pi} \left[\prod_{k=0}^{n-1} e^{-\alpha \delta_{k+1}} \mid s_k, a_k \right] \\ &= E_s^{\pi} \left[e^{-\alpha(\delta_1 + \dots + \delta_n)} \mid s_0, a_0, \dots, s_{n-1}, a_{n-1} \right], \end{aligned}$$

entonces

$$\begin{aligned}
& E_s^\pi \left[c(s_0, a_0) + \sum_{n=1}^{\infty} \prod_{k=0}^{n-1} \Delta_\alpha(s_k, a_k) c(s_n, a_n) \right] \\
&= E_s^\pi \left[\sum_{n=0}^{\infty} E_s^\pi [e^{-\alpha t_n} \mid s_0, a_0, \dots, s_{n-1}, a_{n-1}] c(s_n, a_n) \right] \\
&= \sum_{n=0}^{\infty} E_s^\pi [e^{-\alpha t_n} c(s_n, a_n)] \\
&= V_\alpha^\pi(s),
\end{aligned}$$

donde t_n , $n \in \mathbb{N}$, representa la n -ésima época de decisión y que admite la expresión $t_0 = 0$ y $t_n = \sum_{k=1}^n \delta_k$, para $n \in \mathbb{N}^*$. La penúltima igualdad se debe a las propiedades de esperanza condicional. ■

Para $s \in \mathbb{S}$, $\pi \in \Pi$ y $n \in \mathbb{N}^*$, denotemos, sin pérdida de generalidad, por $V_n^\pi(s)$ el costo esperado α -descontado hasta la n -ésima transición, es decir,

$$V_n^\pi(s) = E_s^\pi \left[\sum_{k=0}^{n-1} e^{-\alpha t_k} c(s_k, a_k) \right].$$

Obsérvese que

$$\lim_{n \rightarrow \infty} V_n^\pi(s) = V_\alpha^\pi(s).$$

Por otra parte, con argumentos similares a los de la demostración correspondiente a la Proposición 4.2, $V_n^\pi(s)$ puede ser expresado como

$$\begin{aligned}
V_1^\pi(s) &= E_s^\pi [c(s_0, a_0)] \\
V_n^\pi(s) &= E_s^\pi \left[c(s_0, a_0) + \sum_{k=1}^{n-1} \prod_{j=0}^{k-1} \Delta_\alpha(s_j, a_j) c(s_k, a_k) \right], \quad n = 2, 3, \dots,
\end{aligned}$$

introduciendo la notación

$$\begin{aligned}
\Lambda_\alpha^0 &:= 1, \\
\Lambda_\alpha^k &:= \prod_{j=0}^{k-1} \Delta_\alpha(s_j, a_j), \quad k \in \mathbb{N}^*,
\end{aligned} \tag{4.12}$$

tenemos

$$V_n^\pi(s) = E_s^\pi \left[\sum_{k=0}^{n-1} \Lambda_\alpha^k c(s_k, a_k) \right], \quad n \in \mathbb{N}^*, \tag{4.13}$$

y

$$V_\alpha^\pi(s) = \lim_{n \rightarrow \infty} V_n^\pi(s) := E_s^\pi \left[\sum_{k=0}^{\infty} \Lambda_\alpha^k c(s_k, a_k) \right]. \quad (4.14)$$

4.4.1. Ecuación de Optimalidad

Sea $\Delta_\alpha(s, a)$ como en (4.11) y $\alpha > 0$. Diremos que una función medible $u : \mathbb{S} \rightarrow \mathbb{R}$ es una solución de la *ecuación de optimalidad* (EO) con costo α -descontado, si satisface

$$u(s) = \min_{a \in A(s)} \left\{ c(s, a) + \Delta_\alpha(s, a) \sum_{s' \in \mathbb{S}} u(s') Q(s' | s, a) \right\}, \quad \forall s \in \mathbb{S}.$$

El objetivo de esta sección es demostrar que la función de valor óptimo satisface la EO. Esto nos permitirá mostrar la existencia de políticas óptimas.

Denotemos por v_n la función de valor óptimo hasta la n -ésima transición. Esto es,

$$\begin{aligned} v_0(s) &:= 0, \\ v_n(s) &:= \inf_{\pi \in \Pi} V_n^\pi(s), \quad s \in \mathbb{S}. \end{aligned}$$

Del Algoritmo de Programación Dinámica (ver [23]) observemos que las funciones v_n pueden ser obtenidas de forma iterada, esto es, para $s \in \mathbb{S}$:

$$\begin{aligned} v_0(s) &= 0, \\ v_n(s) &= \min_{a \in A(s)} \left\{ c(s, a) + \Delta_\alpha(s, a) \sum_{s' \in \mathbb{S}} v_{n-1}(s') Q(s' | s, a) \right\}, \quad (4.15) \end{aligned}$$

para $n \in \mathbb{N}^*$. Por lo tanto, si se muestra que $v_n \rightarrow V_\alpha^*$ se esperaría que V_α^* satisfaga la EO. Esto se formaliza en el Teorema 4.1.

Nota 4.1 *De ahora en adelante consideraremos los costos $c(s, a) = D(s, a)$ y $d(s, a) = 0$.*

Proposición 4.3 *Bajo la Condición 4.2(a), $V_\alpha^* \in \mathbb{B}_w(\mathbb{S})$.*

Demostración. Notemos que de la Condición 4.2(a)

$$\begin{aligned} E_s^\pi [c(s_n, a_n)] &= E_s^\pi [D(s_n, a_n)] \\ &\leq E_s^\pi [c_1 w(s_n)], \end{aligned}$$

hacemos uso de (4.10) para obtener

$$E_s^\pi [c(s_n, a_n)] \leq \left(1 + \frac{b}{1-\varepsilon}\right) c_1 w(s). \quad (4.16)$$

Ahora, por las Proposiciones 4.2 y 4.1(a)

$$V_\alpha^\pi(s) \leq \left(1 + \frac{b}{1-\varepsilon}\right) c_1 w(s) \sum_{n \in \mathbb{N}} (\rho_\alpha)^n \quad (4.17)$$

$$\leq c_1 \left(1 + \frac{b}{1-\varepsilon}\right) \left(\frac{1}{1-\rho_\alpha}\right) w(s), \quad (4.18)$$

por la Definición 4.6 y tomando $M := c_1 \left(1 + \frac{b}{1-\varepsilon}\right) \left(\frac{1}{1-\rho_\alpha}\right)$, de lo anterior se tiene

$$V_\alpha^*(s) \leq Mw(s). \quad (4.19)$$

En consecuencia, V_α^* pertenece a $\mathbb{B}_w(\mathbb{S})$. ■

Observación 4.3 *De forma similar que en la proposición previa se demuestra que para cada $n \in \mathbb{N}$,*

$$v_n(s) \leq Mw(s), \quad s \in \mathbb{S}.$$

Teorema 4.1 *Supóngase que se cumplen las Condiciones 4.1, 4.2 y 4.3, entonces*

a) $\|v_n - V_\alpha^*\|_w \rightarrow 0$ cuando $n \rightarrow \infty$.

b) V_α^* satisface la EO, es decir,

$$V_\alpha^*(s) = \min_{a \in \mathbb{A}(s)} \left\{ c(s, a) + \Delta_\alpha(s, a) \sum_{s' \in \mathbb{S}} V_\alpha^*(s') Q(s' | s, a) \right\}, \quad s \in \mathbb{S}. \quad (4.20)$$

c) Existe $f^* \in \mathbb{F}$ que minimiza el lado derecho en (4.20), es decir

$$V_\alpha^*(s) = c(s, f^*) + \Delta_\alpha(s, f^*) \sum_{s' \in \mathbb{S}} V_\alpha^*(s') Q(s' | s, f^*), \quad \forall s \in \mathbb{S}, \quad (4.21)$$

y la política estacionaria $\pi = \{f^*, f^*, \dots\}$ es óptima.

Demostración. (a) Para $s \in \mathbb{S}$ y $\pi \in \Pi$, de (4.13) y (4.14) tenemos

$$\begin{aligned}
V_\alpha^\pi(s) &= E_s^\pi \left[\sum_{k=0}^{\infty} \Lambda_\alpha^k c(s_k, a_k) \right] \\
&= E_s^\pi \left[\sum_{k=0}^{n-1} \Lambda_\alpha^k c(s_k, a_k) \right] + E_s^\pi \left[\sum_{k=n}^{\infty} \Lambda_\alpha^k c(s_k, a_k) \right] \\
&= V_n^\pi(s) + E_s^\pi \left[\Lambda_\alpha^n \sum_{k=n}^{\infty} \Lambda_\alpha^{k-n} c(s_k, a_k) \right]. \tag{4.22}
\end{aligned}$$

Luego, para cada $s \in \mathbb{S}$,

$$\begin{aligned}
|v_n(s) - V_\alpha^*(s)| &= \left| \inf_{\pi \in \Pi} V_n^\pi(s) - \inf_{\pi \in \Pi} V_\alpha^\pi(s) \right| \\
&\leq \sup_{\pi \in \Pi} |V_n^\pi(s) - V_\alpha^\pi(s)|,
\end{aligned}$$

usando (4.16) y (4.22), así como de la Proposición 4.1(b) se tiene lo siguiente

$$\begin{aligned}
|v_n(s) - V_\alpha^*(s)| &\leq \rho_\alpha^n \sup_{\pi \in \Pi} \sum_{k=n}^{\infty} \rho_\alpha^{k-n} E_s^\pi [c(s_k, a_k)] \\
&\leq \rho_\alpha^n \sup_{\pi \in \Pi} \sum_{k=n}^{\infty} \left(1 - \frac{b}{1-\varepsilon}\right) c_1 w(s) \rho_\alpha^{k-n} \\
&\leq \rho_\alpha^n \left(1 - \frac{b}{1-\varepsilon}\right) c_1 w(s) \frac{1}{1-\rho_\alpha} \\
&= M \rho_\alpha^n w(s).
\end{aligned}$$

Por consiguiente, de la definición de la norma $\|\cdot\|_w$ (ver (4.6)), lo anterior implica que

$$\|v_n - V_\alpha^*\|_w \leq M \rho_\alpha^n.$$

Haciendo tender n a infinito, y usando de la Proposición 4.1 y el hecho de que $\rho_\alpha < 1$, se concluye la demostración de la parte (a).

(b) Tomando límite inferior en ambos lados de (4.15), por el Lema de Fatou generalizado (ver [26] pág. 318) se obtiene que para cada $s \in \mathbb{S}$,

$$\begin{aligned}
\liminf_{n \rightarrow \infty} v_n(x) &= \min_{a \in \mathbb{A}(s)} \left\{ c(s, a) \right. \\
&\quad \left. + \Delta_\alpha(s, a) \liminf_{n \rightarrow \infty} \left(\sum_{s' \in \mathbb{S}} v_{n-1}(s') Q(s' | s, a) \right) \right\} \\
&\geq \min_{a \in \mathbb{A}} \left\{ c(s, a) \right. \\
&\quad \left. + \Delta_\alpha(s, a) \sum_{s' \in \mathbb{S}} \liminf_{n \rightarrow \infty} v_{n-1}(s') Q(s' | s, a) \right\},
\end{aligned}$$

luego, aplicando lo demostrado en la parte (a), lo anterior implica que

$$V_\alpha^*(s) \geq \min_{a \in \mathbb{A}} \left\{ c(s, a) + \Delta_\alpha(s, a) \sum_{s' \in \mathbb{S}} V_\alpha^*(s') Q(s' | s, a) \right\}. \quad (4.23)$$

Ahora obsérvese que (4.15) implica

$$v_n(s) \leq c(s, a) + \Delta_\alpha(s, a) \sum_{s' \in \mathbb{S}} v_{n-1}(s') Q(s' | s, a), \quad s \in \mathbb{S}, a \in \mathbb{A}. \quad (4.24)$$

Nuevamente usando Lema de Fatou, se tiene que

$$V_\alpha^*(s) \leq c(s, a) + \Delta_\alpha(s, a) \sum_{s' \in \mathbb{S}} V_\alpha^*(s') Q(s' | s, a). \quad (4.25)$$

Por consiguiente, la parte (b) se obtiene combinando (4.23) y (4.25).

(c) Puesto que $\Delta_\alpha(s, a)$ es continua en $\mathbb{A}(s)$ para $s \in \mathbb{S}$, entonces por el Lema 4.2, existe $f^* \in \mathbb{F}$ que satisface (4.21), es decir,

$$V_\alpha^*(s) = c(s, f^*) + \Delta_\alpha(s, f^*) \sum_{s' \in \mathbb{S}} V_\alpha^*(s') Q(s' | s, f^*), \quad s \in \mathbb{S},$$

iterando la ecuación obtenemos

$$V_\alpha^*(s) = E_s^{f^*} \left[\sum_{n=0}^{N-1} \Lambda_\alpha^n c(s_n, a_n) + \Lambda_\alpha^N V_\alpha^*(s_N) \right],$$

de donde

$$V_\alpha^*(s) \geq V_N(f^*, s) \quad \forall s \in \mathbb{S}, N \in \mathbb{N}^*, \quad (4.26)$$

pues $\Lambda_\alpha^N V_\alpha^*(s_N) \geq 0$ para cada $N \in \mathbb{N}^*$.

Haciendo $N \rightarrow \infty$ en (4.26) se tiene

$$V_{\alpha}^{*}(s) \geq V(f^{*}, s), \quad \forall s \in \mathbb{S}. \quad (4.27)$$

Por otra parte, de las Definiciones 4.5 y 4.6 se deduce que

$$V_{\alpha}^{*}(s) \leq V(f^{*}, s), \quad \forall s \in \mathbb{S}. \quad (4.28)$$

En consecuencia

$$V_{\alpha}^{*}(s) = V(f^{*}, s), \quad \forall s \in \mathbb{S},$$

y por lo tanto, $\pi = \{f^{*}, f^{*}, \dots\}$ es óptima. ■

4.5. Optimalidad de una Política Umbral en el Sistema $M/M/1$ con Periodos de Recesos Repetidos

Los sistemas de colas con periodos de recesos permitidos en el servidor ya han sido estudiados y discutidos en [9] y [10]. Los modelos con receso se pueden clasificar en dos categorías: los modelos donde al servidor se le permite recesos repetidos (múltiples) y modelos donde el servidor puede ser extraído. En el primer caso, la duración de un período de receso es impulsado por un proceso externo (es decir, los periodos de receso son v.a.i.i.d. ver, por ejemplo, [11] y [12]), mientras que en el segundo caso, el tiempo de un período de receso es impulsado por el proceso de llegada (ver [13] y [14]). En ambos casos, el servidor puede ser apagado en cualquier etapa de culminación de servicio de un cliente y puede ser activado sólo en una etapa de llegada de un cliente en el caso del servidor extraíble, y sólo al final de un período de receso en el caso de recesos repetidos. De particular interés es la política estacionaria que activa al servidor sólo cuando el número de clientes en la cola es mayor o igual que un valor dado, y apaga el servidor cuando la cola está vacía. Esta política se conoce como la **política umbral** (ver [8]).

Un objetivo es el de buscar una política de receso que optimice una función de costo dada entre ciertas clases de políticas. Para la cola

$M/G/1$ con servidor extraíble [13] y [15] han demostrado que un criterio de costo medio bastante general se minimiza mediante una política umbral entre la clase de todas las políticas. Pocos resultados de optimización se han reportado hasta el momento para el problema más difícil de controlar el proceso de servicio de una cola con recesos repetidos. En [12] se ha calculado la mejor política umbral para un sistema $M/G/1$ con periodos de recesos repetidos, y en [16] se ha llevado a cabo el mismo análisis en el caso de llegadas en grupo. En el caso de que la decisión de tomar otro período de receso esté basado en un resultado aleatorio en función del número de recesos consecutivos ya tomado, [12] ha demostrado que una política de control de una clase de límite minimiza un criterio de costo medio a largo plazo.

Lo que se quiere es establecer la optimización de una política umbral sobre la clase de todas las políticas, incluidas las que puedan depender de la historia del sistema (es decir, el número de clientes en cualquier momento y las decisiones realizadas anteriormente).

Más precisamente consideremos un sistema $M/M/1$ con periodos de recesos repetidos, donde las longitudes de estos periodos son v.a.i.i.d., con una distribución exponencial. Suponga que un costo de almacenamiento $h > 0$ se incurre por unidad de tiempo y por cliente y que un costo fijo $\gamma > 0$ es incurrido cuando el servidor se activa o reanuda. *Se mostrará que existe una política umbral óptima que minimiza el criterio de costo total esperado α -descontado.*

En [17] se ha demostrado que la optimalidad de una política umbral en realidad se extiende al sistema de colas $M/G/1$ con los periodos de recesos repetidos arbitrarios. Sin embargo, el umbral óptimo no se puede determinar de forma explícita en este caso.

Esta sección se organiza de la siguiente manera. Primero, el problema se formula como un Problema de Decisión de Markov, después se introduce el problema de costo total esperado α -descontado y se resuelve a través de Programación Dinámica.

4.5.1. Descripción del Modelo

Consideramos un sistema $M/M/1$ donde el servidor atiende a los clientes de acuerdo con una disciplina arbitraria hasta que la cola se vacía, y después toma un receso de tiempo arbitrario. Al volver de este período, el servidor puede o bien seguir apagado (i.e. tomar otro receso) o puede ser activado (reanudar el trabajo), siempre que la cola no esté vacía, de acuerdo con una *política de receso* (una definición de política de receso se dará más adelante). Las longitudes de los recesos, son v.a.i.i.d con una distribución exponencial. Suponemos que los procesos de tiempos entre llegadas, los de servicios y de receso son independientes entre sí. Sean $1/\lambda$, $1/\mu$ y $1/\nu$ el tiempo medio entre llegadas, el tiempo medio de servicio y el tiempo medio de recesos, respectivamente.

Asumimos que un costo de almacenamiento $h > 0$ se incurre por unidad de tiempo y por cliente en la cola y que un costo de reinicio $\gamma > 0$ se incurre cada vez que el servidor se reanuda. Nuestro objetivo es encontrar una política que minimice el costo total α -descontado con horizonte infinito.

Para poder formular el problema de control como un Proceso de Decisión de Markov (PDM), tenemos que introducir más puntos de observación. Concretamente, vamos a suponer que el sistema (para ser más precisos) se observa en cada llegada, en cada finalización de servicio y en cada tiempo de salto de un proceso Poisson \mathbf{V} , independiente de los procesos de entrada y de servicio, con parámetro ν . El proceso \mathbf{V} desempeñará el papel de un proceso de receso *virtual* en el sentido de que, si se produce un salto en \mathbf{V} cada vez que el servidor está en reposo, entonces este salto puede tomarse como el tiempo de finalización del próximo receso. Esto se desprende de la propiedad de pérdida de memoria de la distribución exponencial, la independencia de los tiempos de llegadas, servicios y el proceso \mathbf{V} , y el hecho de que el tiempo medio de intereventos para el proceso \mathbf{V} es el mismo que el tiempo medio de recesos. En cada punto de observación que corresponde a un salto en \mathbf{V} , una decisión será tomada ya sea en reanudar o no el servidor (siempre

que la cola no esté vacía). Esto se deduce de la descripción del problema de control y de la definición del proceso virtual \mathbf{V} . En todos los demás puntos de observación, se tomarán acciones ficticias que no afectará el comportamiento del sistema.

Con lo anterior debemos construir un PDM tal que el estado del proceso en la n -ésima época de decisión t_n , $n \in \mathbb{N}$, se represente por una tupla $(X_n, Y_n, Z_n) \in \mathbb{N} \times \{0, 1\}^2$, donde X_n es la longitud de la cola al tiempo t_n^+ , $Y_n \in \{0, 1\}$ describe la actividad del servidor al tiempo t_n^- . ($Y_n = 1$ si el servidor está activo y $Y_n = 0$ si está en reposo) y $Z_n = I_{(t_n \in \mathbf{V})}$, esto es $Z_n = 1$ si t_n es un tiempo de salto del proceso \mathbf{V} y $Z_n = 0$, en caso contrario. Se puede ver de la definición de X_n , Y_n y Z_n , $n \in \mathbb{N}$, que los estados $(0, 0, 0)$ y $(0, 1, 1)$ no son accesibles.

El PDM puede construirse de la siguiente manera.

Sea $\mathbb{S} := \mathbb{N} \times \{0, 1\}^2 - \{(0, 0, 0), (0, 1, 1)\}$ el espacio de estados, y sea $\mathbb{A}(x, y, z) \subset \mathbb{A} = \{0, 1\}$ el conjunto de todas las acciones disponibles cuando el sistema se encuentra en el estado $s = (x, y, z) \in \mathbb{S}$. Asumimos que

$$\begin{aligned} \mathbb{A}(x, 1, 1) &= \{1\}, \quad x \geq 1; \\ \mathbb{A}(x, 1, 0) &= \begin{cases} \{1\}, & x \geq 1; \\ \{0\}, & x = 0; \end{cases} \\ \mathbb{A}(x, 0, 1) &= \begin{cases} \{0, 1\}, & x \geq 1; \\ \{0\}, & x = 0; \end{cases} \\ \mathbb{A}(x, 0, 0) &= \{0\}, \quad x \geq 1, \end{aligned}$$

donde por convención la acción 1 (respectivamente 0) se toma si la decisión es reanudar el servidor (apagar resp.). Notemos que los únicos estados cuando más de una acción está disponible son los estados $(x, 0, 1)$ con $x \geq 1$, esto es, cuando ocurre un salto en \mathbf{V} ($z = 1$), el servidor está en reposo ($y = 0$) y que la cola no esté vacía ($x \geq 1$). Esto refleja dicho problema de control. En todos los demás casos, la definición de

una sola acción disponible es arbitraria, excepto para el estado $(0, 1, 0)$ cuando necesariamente la acción 0 necesita tomarse para seguir con el esquema de la definición de receso.

Una *política de receso* π , (ver [18]), se define como una sucesión de probabilidades condicionales $\pi_n : \mathbb{H}_n \rightarrow \mathcal{P}(\{0, 1\})$, $\mathbb{H}_1 = \mathbb{S}$, $\mathbb{H}_{n+1} = \mathbb{H}_n \times \{0, 1\}$, para toda $n \in \mathbb{N}$, tal que $\pi_n(h_n; \bullet)$ le asigna probabilidad uno al conjunto $\mathbb{A}(s_n)$, donde $h_n = (s_1, a_1, \dots, s_{n-1}, a_{n-1}, s_n)$, ($\mathcal{P}(\{0, 1\})$ es el conjunto de todas las medidas de probabilidad en $\{0, 1\}$). Una política es *estacionaria* si $\pi_n(h_n; \bullet)$ está concentrada en el punto $\alpha(s_n)$ para toda $h_n = (s_0, a_0, \dots, s_{n-1}, a_{n-1}, s_n) \in \mathbb{H}$, $n \in \mathbb{N}$, donde α es un mapeo medible de \mathbb{S} a $\{0, 1\}$.

Sea $Q(\bullet \mid s; a)$ la distribución de probabilidad del siguiente estado visitado por el sistema dado que se encuentra en el estado $s = (x, y, x) \in \mathbb{S}$ y la acción $a \in \mathbb{A}(x, y, z)$ es tomada. Las probabilidades de transición están dadas por (con $\beta := \lambda + \mu + \nu$)

$$Q(x-1, 1, 0 \mid x, 1, 1; 1) = \mu/\beta, \quad x \geq 1;$$

$$Q(x+1, 1, 0 \mid x, 1, 1; 1) = \lambda/\beta, \quad x \geq 1;$$

$$Q(x, 1, 1 \mid x, 1, 1; 1) = \nu/\beta, \quad x \geq 1;$$

$$Q(x-1, 1, 0 \mid x, 1, 0; 1) = \mu/\beta, \quad x \geq 1;$$

$$Q(x+1, 1, 0 \mid x, 1, 0; 1) = \lambda/\beta, \quad x \geq 1;$$

$$Q(x, 1, 1 \mid x, 1, 0; 1) = \nu/\beta, \quad x \geq 1;$$

$$Q(0, 0, 1 \mid 0, 1, 0; 0) = \nu/(\lambda + \nu);$$

$$Q(1, 0, 0 \mid 0, 1, 0; 0) = \lambda/(\lambda + \nu);$$

$$Q(x-1, 1, 0 \mid x, 0, 1; 1) = \mu/\beta, \quad x \geq 1;$$

$$Q(x+1, 1, 0 \mid x, 0, 1; 1) = \lambda/\beta, \quad x \geq 1;$$

$$Q(x, 1, 1 \mid x, 0, 1; 1) = \nu/\beta, \quad x \geq 1;$$

$$Q(x+1, 0, 0 \mid x, 0, 1; 0) = \lambda/(\lambda + \nu), \quad x \in \mathbb{N};$$

$$Q(x, 0, 1 \mid x, 0, 1; 0) = \nu/(\lambda + \nu), \quad x \in \mathbb{N};$$

$$\begin{aligned} Q(x+1, 0, 0 \mid x, 0, 0; 0) &= \lambda/(\lambda + \nu), & x \geq 1; \\ Q(x, 0, 1 \mid x, 0, 0; 0) &= \nu/(\lambda + \nu), & x \geq 1. \end{aligned}$$

Dada una distribución inicial sobre \mathbb{S} y la ley de transición Q , cualquier política $\pi \in \Pi$ define una medida de probabilidad sobre el espacio producto $(\mathbf{X} \times \mathbb{A})^\infty$ dotado de la σ -álgebra producto. Sea E^π el operador esperanza asociado con esta medida de probabilidad. Sobre este espacio de probabilidad está definido el vector aleatorio (X_n, Y_n, Z_n) que describe el estado del sistema en la n -ésima época de decisión así como la v.a. A_n que describe la acción tomada en la n -ésima época de decisión, $n \in \mathbb{N}$.

Definición 4.7 Sean $s = (x, y, z) \in \mathbb{S}$ y $a \in \mathbb{A}(x, y, z)$, recordemos de la Nota 4.1 que $c(s, a) = D(s, a)$, definamos

$$c(s, a) = D(x, y, z; a) := (h/\nu)xz + \gamma(1-y)z \cdot I_{(a=1)}, \quad (4.29)$$

donde h es el costo por unidad de tiempo y por cliente en el sistema, γ es el costo que se incurre cada vez que el servidor se reanuda y $I_{(a=1)}$ es la función indicadora de $(a = 1)$, es decir, toma valor uno si $a = 1$ y cero en otro caso.

Distribución de tiempo de servicio.

Denotemos por $\hat{\beta}$ la tasa tasa de servicio para cada estado dado la acción. Sea $s = (x, y, z) \in \mathbb{S}$ y $a \in \mathbb{A}(s)$ entonces

$$\hat{\beta}(s, a) = \begin{cases} \beta, & x \geq 1, y, z \in \{0, 1\}, a = 1; \\ \lambda + \nu, & (x, y, z) \in \mathbb{S}, a = 0; \end{cases}$$

así

$$H(t \mid s, a) = \begin{cases} \beta e^{-\beta t}, & x \geq 1, y, z \in \{0, 1\}, a = 1; \\ (\lambda + \nu)e^{-(\lambda + \nu)t}, & (x, y, z) \in \mathbb{S}, a = 0, \end{cases} \quad (4.30)$$

luego de (4.30) desarrollamos (4.11), para el caso $a = 1$,

$$\begin{aligned}\Delta_\alpha(s, 1) &= \int_0^\infty e^{-\alpha t} H(dt \mid s, 1) \\ &= \int_0^\infty e^{-\alpha t} (\beta e^{-\beta t}) dt \\ &= \beta \int_0^\infty e^{-(\alpha+\beta)t} dt \\ &= \frac{\beta}{\alpha + \beta},\end{aligned}$$

es análogo para el caso $a = 0$, así

$$\Delta_\alpha(s, a) = \begin{cases} \frac{\beta}{\alpha+\beta}, & a = 1; \\ \frac{\lambda+\nu}{\alpha+\lambda+\nu}, & a = 0. \end{cases} \quad (4.31)$$

Nota 4.2 *A menos que se mencione lo contrario, se supondrá de ahora en adelante que $h/\nu = 1$.*

Sea $(x, y, z) \in \mathbb{S}$, $x \geq 1$ y $a \in \mathbb{A}(x, y, z)$, a partir del Teorema 4.1 tenemos

$$\begin{aligned}V_\alpha(x, 1, 1) &= \min_{a \in \mathbb{A}(x, 1, 1)} \left\{ c(x, 1, 1; a) \right. \\ &\quad \left. + \Delta_\alpha(s, a) \sum_{(x', y', z') \in \mathbb{S}} V_\alpha(x', y', z') Q(x', y', z' \mid x, 1, 1; a) \right\}\end{aligned}$$

como $a \in \mathbb{A}(x, 1, 1) = \{1\}$, para $x \geq 1$, entonces de (4.29) y de la Nota 4.2 tenemos que

$$c(x, 1, 1; 1) = x,$$

así

$$\begin{aligned}V_\alpha(x, 1, 1) &= x + \Delta_\alpha(s, a) \left(V_\alpha(x-1, 1, 0) \frac{\mu}{\beta} + V_\alpha(x+1, 1, 0) \frac{\lambda}{\beta} \right. \\ &\quad \left. + V_\alpha(x, 1, 1) \frac{\nu}{\beta} \right),\end{aligned}$$

donde $\beta = (\lambda + \mu + \nu)$, de (4.31) tenemos

$$\begin{aligned}V_\alpha(x, 1, 1) &= x + \left(\frac{\beta}{\alpha + \beta} \right) \frac{1}{\beta} \left(V_\alpha(x-1, 1, 0) \mu + V_\alpha(x+1, 1, 0) \lambda \right. \\ &\quad \left. + V_\alpha(x, 1, 1) \nu \right),\end{aligned}$$

finalmente

$$(\alpha + \beta)V_\alpha(x, 1, 1) = x(\alpha + \beta) + \mu V_\alpha(x - 1, 1, 0) + \lambda V_\alpha(x + 1, 1, 0) + \nu V_\alpha(x, 1, 1), \text{ para } x \geq 1. \quad (4.32)$$

Análogamente se deducen las siguientes ecuaciones de optimalidad

$$(\alpha + \beta)V_\alpha(x, 1, 0) = \mu V_\alpha(x - 1, 1, 0) + \lambda V_\alpha(x + 1, 1, 0) + \nu V_\alpha(x, 1, 1), \text{ para } x \geq 1; \quad (4.33)$$

$$(\alpha + \beta + \nu)V_\alpha(0, 1, 0) = \lambda V_\alpha(1, 0, 0) + \nu V_\alpha(0, 0, 1); \quad (4.34)$$

$$(\alpha + \beta + \nu)V_\alpha(0, 0, 1) = \lambda V_\alpha(1, 0, 0) + \nu V_\alpha(0, 0, 1); \quad (4.35)$$

$$V_\alpha(x, 0, 1) = \min \left\{ x + \frac{\lambda}{\alpha + \lambda + \nu} V_\alpha(x + 1, 0, 0) + \frac{\nu}{\alpha + \lambda + \nu} V_\alpha(x, 0, 1); \right. \\ \left. x + \gamma + \frac{\mu}{\alpha + \beta} V_\alpha(x - 1, 1, 0) + \frac{\lambda}{\alpha + \beta} V_\alpha(x + 1, 1, 0) + \frac{\nu}{\alpha + \beta} V_\alpha(x, 1, 1) \right\}, \text{ para } x \geq 1; \quad (4.36)$$

$$(\alpha + \lambda + \nu)V_\alpha(x, 0, 0) = \lambda V_\alpha(x + 1, 0, 0) + \nu V_\alpha(x, 0, 1), \text{ para } x \geq 1. \quad (4.37)$$

A continuación se reducirá el número de incógnitas involucradas en el conjunto de ecuaciones (4.32)-(4.37).

Primero, combinando (4.34) y (4.35) tenemos

$$V_\alpha(0, 1, 0) = V_\alpha(0, 0, 1) = \frac{\lambda}{\alpha + \lambda} V_\alpha(1, 0, 0). \quad (4.38)$$

Usando (4.33) y (4.37) vemos que (4.36) puede ser reescrito como

$$V_\alpha(x, 0, 1) = x + \min \left\{ V_\alpha(x, 0, 0); \gamma + V_\alpha(x, 1, 0) \right\} \text{ para } x \geq 1. \quad (4.39)$$

Introduciendo ahora (4.39) en (4.37) para $x \geq 1$

$$(\alpha + \lambda + \nu)V_\alpha(x, 0, 0) = \nu x + \lambda V_\alpha(x + 1, 0, 0) + \nu \min \left\{ V_\alpha(x, 0, 0); V_\alpha(x, 1, 0) + \gamma \right\}. \quad (4.40)$$

Por otro lado, observamos de (4.32) y (4.33) que

$$V_\alpha(x, 1, 1) = x + V_\alpha(x, 1, 0), \quad \text{para } x \geq 1. \quad (4.41)$$

Introduciendo (4.41) en (4.33) finalmente tenemos, con $x \geq 1$,

$$(\alpha + \lambda + \mu)V_\alpha(x, 1, 0) = \nu x + \lambda V_\alpha(x + 1, 1, 0) + \mu V_\alpha(x - 1, 1, 0). \quad (4.42)$$

Las relaciones (4.38), (4.40), (4.41) y (4.42) contienen toda la información que en la ecuación de PD (4.20). (4.42) indica que define una ecuación en diferencias. Esto nos lleva al siguiente resultado:

Lema 4.3

$$V_\alpha(x, 1, 0) = \left(V_\alpha(1, 0, 0) \left(\frac{\lambda}{\alpha + \lambda} \right) + \nu \left(\frac{\mu - \lambda}{\alpha^2} \right) \right) \beta_{\alpha,1}^x + \frac{\nu}{\alpha} x + \nu \left(\frac{\lambda - \mu}{\alpha^2} \right), \quad (4.43)$$

para toda $x \in \mathbb{N}$, donde $\beta_{\alpha,1}$, $0 < \beta_{\alpha,1} < 1$, es la raíz más pequeña (en z) del polinomio $\lambda z^2 - (\alpha + \lambda + \mu)z + \mu$.

Demostración. Veamos que la solución general a la ecuación en diferencias (4.42) es

$$V_\alpha(x, 1, 0) = a\beta_{\alpha,1}^x + b\beta_{\alpha,2}^x + cx + d, \quad (4.44)$$

para $x \in \mathbb{N}$, donde $\beta_{\alpha,1}$ y $\beta_{\alpha,2}$ son las raíces del polinomio (en z) $\lambda z^2 - (\alpha + \lambda + \mu)z + \mu$, con $0 < \beta_{\alpha,1} < 1 < \beta_{\alpha,2}$.

La relación (4.42) la podemos ver como

$$-\lambda V_\alpha(x + 1, 1, 0) + (\alpha + \lambda + \mu)V_\alpha(x, 1, 0) - \mu V_\alpha(x - 1, 1, 0) = \nu x, \quad (4.45)$$

que es una ecuación en diferencias de orden dos, primero encontramos la solución para la ecuación homogénea y después para la no-homogénea.

Para el primer caso

$$-\lambda V_\alpha(x + 1, 1, 0) + (\alpha + \lambda + \mu)V_\alpha(x, 1, 0) - \mu V_\alpha(x - 1, 1, 0) = 0,$$

la ecuación característica es

$$-\lambda z^2 + (\alpha + \lambda + \mu)z - \mu = 0, \quad (4.46)$$

ahora supongamos que $\beta_{\alpha,1}$ y $\beta_{\alpha,2}$ son las soluciones de (4.46), luego la solución general de la ecuación homogénea es

$$V_{\alpha}(x, 1, 0) = a\beta_{\alpha,1}^x + b\beta_{\alpha,2}^x. \quad (4.47)$$

Para el caso no-homogéneo proponemos una solución particular, digamos

$$V_{\alpha}(x, 1, 0) = cx + d, \quad (4.48)$$

sustituimos (4.48) en (4.45)

$$-\lambda(c(x+1) + d) + (\alpha + \lambda + \mu)(cx + d) - \mu(c(x-1) + d) = \nu x, \quad (4.49)$$

de (4.49) obtenemos que

$$\alpha cx = \nu x$$

y

$$(\mu - \lambda)c + \alpha d = 0,$$

por lo tanto

$$c = \frac{\nu}{\alpha}$$

y

$$d = \nu \left(\frac{\lambda - \mu}{\alpha^2} \right).$$

Luego, la solución general de (4.45) es la suma de (4.47) y (4.48), i.e. tenemos (4.44) .

Recordando que $V_{\alpha}(x, 1, 0)/x$ es uniformemente acotada en \mathbb{N}^* , podemos ver de (4.44) que necesariamente $b = 0$ puesto que $\beta_{\alpha,2} > 1$. Para encontrar el coeficiente a usamos como condición inicial (4.38), entonces de (4.44) para $x = 0$ tenemos

$$V_{\alpha}(0, 1, 0) = a + \nu \left(\frac{\lambda - \mu}{\alpha^2} \right),$$

finalmente, usando (4.38) tenemos que

$$a = V_{\alpha}(1, 0, 0) \left(\frac{\lambda}{\alpha + \lambda} \right) + \nu \left(\frac{\mu - \lambda}{\alpha^2} \right).$$

Luego, se sigue (4.43). ■

En resumen, se ha demostrado que para todo $(x, y, z) \in \mathbb{S}$, $V(x, y, z)$ se expresa únicamente de $V(1, 1, 0)$, $V(x, 0, 1)$ y $V(x, 0, 0)$ para $x \geq 1$, mediante las ecuaciones (4.38), (4.40), (4.41) y (4.43).

Del Teorema 4.1 se sigue que la política estacionaria $\pi^* = (f^*, f^*, \dots)$ derivado de la regla de decisión, con $s = (x, y, z)$

$$f^*(s) = \begin{cases} 1, & V_\alpha^*(x, 1, 0), V_\alpha^*(x, 1, 1), x \geq 1; \\ 0, & V_\alpha^*(x, 0, 0), V_\alpha^*(0, 1, 0) \text{ y } V_\alpha^*(0, 0, 1), x \geq 1, \end{cases} \quad (4.50)$$

es óptima.

4.5.2. Ejemplo Numérico

A continuación se muestran los resultados de los costos $V_n(x, y, z)$ en forma de la siguiente matriz,

$$\begin{pmatrix} V_n(0, 0, 0) & V_n(0, 1, 0) & V_n(0, 0, 1) & V_n(0, 1, 1) \\ V_n(1, 0, 0) & V_n(1, 1, 0) & V_n(1, 0, 1) & V_n(1, 1, 1) \\ V_n(2, 0, 0) & V_n(2, 1, 0) & V_n(2, 0, 1) & V_n(2, 1, 1) \\ \vdots & \vdots & \vdots & \vdots \\ V_n(m, 0, 0) & V_n(m, 1, 0) & V_n(m, 0, 1) & V_n(m, 1, 1) \end{pmatrix}.$$

A continuación se muestran los resultados de B.5, que son los costos $V_n(x, y, z)$ en forma de matriz, para valores de $n = \{1, 2, 5, 10, 13, 14, 15, 16\}$, donde $\alpha = 1/2$, $\lambda = 1/5$, $\mu = 1/2$, $\nu = 1/2$, $\gamma = 2$ y $m = 10$.

$N = 1$				$N = 2$			
-	0	0	-	-	0.1190	0.1190	-
0	0.4108	1.0000	1.4108	0.4167	0.4130	1.4167	1.4130
0	1.4002	2.0000	3.4002	0.8333	1.4002	2.8333	3.4002
0	2.4000	3.0000	5.4000	1.2500	2.4000	4.2500	5.4000
0	3.4000	4.0000	7.4000	1.6667	3.4000	5.6667	7.4000
0	4.4000	5.0000	9.4000	2.0833	4.4000	7.0833	9.4000
0	5.4000	6.0000	11.4000	2.5000	5.4000	8.5000	11.4000
0	6.4000	7.0000	13.4000	2.9167	6.4000	9.9167	13.4000
0	7.4000	8.0000	15.4000	3.3333	7.4000	11.3333	15.4000
0	8.4000	9.0000	17.4000	3.7500	8.4000	12.7500	17.4000
0	9.4000	10.0000	19.4000	4.1667	9.4000	14.1667	19.4000

$N = 5$				$N = 10$			
-	0.3159	0.3159	-	-	0.3911	0.3911	-
1.1056	0.4165	2.1056	1.4165	1.3689	0.4179	2.3689	1.4179
1.9898	1.4003	3.9898	3.4003	2.3611	1.4003	4.3611	3.4003
2.8740	2.4000	5.8740	5.4000	3.3533	2.4000	6.3533	5.4000
3.7582	3.4000	7.7582	7.4000	4.3455	3.4000	8.3455	7.4000
4.6424	4.4000	9.6424	9.4000	5.3377	4.4000	10.3377	9.4000
5.5266	5.4000	11.5266	11.4000	6.3298	5.4000	12.3298	11.4000
6.4108	6.4000	13.4108	13.4000	7.3220	6.4000	14.3220	13.4000
7.2950	7.4000	15.2950	15.4000	8.3141	7.4000	16.3141	15.4000
8.1793	8.4000	17.1793	17.4000	9.3056	8.4000	18.3056	17.4000
9.0635	9.4000	19.0635	19.4000	10.2958	9.4000	20.2958	19.4000

70 **4.5. Optimalidad de una Política Umbral en $M/M/1$ con
Recesos Repetidos**

$N = 13$				$N = 14$			
-	0.3979	0.3979	-	-	0.3987	0.3987	-
1.3925	0.4180	2.3925	1.4180	1.3954	0.4180	2.3954	1.4180
2.3910	1.4003	4.3910	3.4003	2.3945	1.4003	4.3945	3.4003
3.3894	2.4000	6.3894	5.4000	3.3936	2.4000	6.3936	5.4000
4.3878	3.4000	8.3878	7.4000	4.3927	3.4000	8.3927	7.4000
5.3863	4.4000	10.3863	9.4000	5.3917	4.4000	10.3917	9.4000
6.3847	5.4000	12.3847	11.4000	6.3908	5.4000	12.3908	11.4000
7.3831	6.4000	14.3831	13.4000	7.3899	6.4000	14.3899	13.4000
8.3814	7.4000	16.3814	15.4000	8.3888	7.4000	16.3888	15.4000
9.3795	8.4000	18.3795	17.4000	9.3877	8.4000	18.3877	17.4000
10.3772	9.4000	20.3772	19.4000	10.3863	9.4000	20.3863	19.4000

$N = 15$				$N = 16$			
-	0.3992	0.3992	-	-	0.3995	0.3995	-
1.3971	0.4180	2.3971	1.4180	1.3982	0.4180	2.3982	1.4180
2.3966	1.4003	4.3966	3.4003	2.3979	1.4002	4.3979	3.4002
3.3961	2.4000	6.3961	5.4000	3.3976	2.4000	6.3976	5.4000
4.3956	3.4000	8.3956	7.4000	4.3973	3.4000	8.3973	7.4000
5.3950	4.4000	10.3950	9.4000	5.3970	4.4000	10.3970	9.4000
6.3945	5.4000	12.3945	11.4000	6.3967	5.4000	12.3967	11.4000
7.3939	6.4000	14.3939	13.4000	7.3964	6.4000	14.3939	13.4000
8.3933	7.4000	16.3933	15.4000	8.3960	7.4000	16.3933	15.4000
9.3926	8.4000	18.3926	17.4000	9.3955	8.4000	18.3926	17.4000
10.3917	9.4000	20.3917	19.4000	10.3950	9.4000	20.3917	19.4000

Conclusiones

Este trabajo de tesis se enfocó en el estudio de la Teoría de Colas y la Teoría de los Procesos de Control Semi-Markovianos bajo el criterio de costo descontado.

El PCO consiste en encontrar una política que optimice el criterio de rendimiento; utilizando resultados conocidos se encontraron políticas de control óptimas. Una manera de resolver un PDSM está basado en Programación Dinámica y nos permitió resolver el problema de control. El principio de Programación Dinámica proporciona una técnica para determinar de manera eficiente las estrategias que optimizan un problema.

En el Capítulo 2, se presentaron los conceptos generales de la Teoría de Colas. Se hizo un estudio analítico de los modelos $M/M/1$ y $M/M/s$.

En el Capítulo 3, se realizó la simulación del modelo $M/M/1$, se compararon estos resultados con la teoría vista en el Capítulo 2 y la simulación para el caso *no-homogéneo* mediante el método de simulación con eventos discretos.

En el Capítulo 4, se vieron los conceptos generales de la Teoría de Procesos de Control Semi-Markovianos y se estudió un sistema $M/M/1$ con recesos permitidos; los resultados fueron analizados en un ejemplo numérico, el cual se elaboraron algoritmos computacionales en MATLAB.

Cabe mencionar que en el Capítulo 2, los sistemas estudiados sólo son observados en el tiempo sin ningún tipo de control, contrario a lo estudiado en el Capítulo 4, ya que en estos sistemas ingresamos variables

de control y sobre ello (políticas) se toma una acción y en base a esto se optimizan (en nuestro caso) costos.

En resumen, las aportaciones de este trabajo de tesis son:

1. Se presentó un estudio analítico de los sistemas de colas $M/M/1$ y $M/M/s$.
2. Se realizó la simulación de dichos modelos de colas y se compararon los resultados con la teoría vista en el Capítulo 2, para los estos modelos se proveen algoritmos computacionales elaborados en MATLAB.
3. Se considera un ejemplo del sistema $M/M/1$ controlado, para el cual se caracteriza una estrategia óptima de operación.

Una posible extensión de la tesis es la de trabajar con el costo medio óptimo a largo plazo, es decir mostrar que existe una política umbral que minimiza el criterio de costo medio a largo plazo.

Para la elaboración del trabajo se hizo uso de la siguiente bibliografía.

Para la Teoría de Colas se estudió [1] y [2]. Sobre la simulación de los sistemas no-homogéneos se consultó [3]. De la teoría y resultados acerca de los PCSM se consultaron [7] y [25]. El resto de la bibliografía sirvió de apoyo.

Apéndice A

Otros resultados

A.1. Funciones

Definición A.1 Sea (X, τ) un espacio topológico, la mínima σ -álgebra que contiene a τ es la σ -álgebra de Borel, es decir, la σ -álgebra generada por τ . La denotaremos por $\mathfrak{B}(X)$.

Cuando se hable de conjuntos o funciones medibles, se entenderán como Borel medibles.

Definición A.2 X es un espacio de Borel, si X es un subconjunto de Borel de un espacio métrico separable y completo.

Definición A.3 Un kernel estocástico definido sobre \mathbb{X} dado \mathbb{Y} es una función $Q(\cdot | \cdot)$ tal que:

1. $Q(\cdot | y)$ es una medida de probabilidad en \mathbb{X} para cada $y \in \mathbb{Y}$.
2. $Q(B | \cdot)$ es una variable aleatoria (función medible) en \mathbb{Y} para cada $B \in \mathfrak{B}(\mathbb{X})$.

La familia de todos los kernels estocásticos lo denotamos por $P(X | Y)$.

Definición A.4 Sean $x \in X$ y $f(x)$ una función definida para cada $x \in X$, con X numerable. Se define $\arg \max$ como

$$x^* \in \arg \max_{x \in X} f(x) \Leftrightarrow f(x^*) = \max_{x \in X} f(x)$$

es decir

$$\arg \max f = \{x^* \in X : f(x^*) = \max_{x \in X} f(x)\}.$$

Definición A.5 Sea \mathbb{X} un espacio topológico y v una función de $\mathbb{X} \in \mathbb{R} \cup \{\infty\}$. Se dice que la función v es:

A.2. Teorema de Ionescu-Tulcea

Teorema A.1 Sea $\mathbb{X}_0, \mathbb{X}_1, \dots$ una sucesión de espacios de Borel y, para cada $n \in \mathbb{N}$, defínase

$$\mathbb{Y}_n := \mathbb{X}_0 \times \mathbb{X}_1 \times \dots \times \mathbb{X}_n$$

y

$$\mathbb{Y} := \mathbb{X}_0 \times \mathbb{X}_1 \times \dots$$

Sea v una medida de probabilidad arbitraria en \mathbb{X}_{n+1} y, para cada $n \in \mathbb{N}$, sea $Q_n(dx_{n+1} | y_n)$ un kernel estocástico sobre \mathbb{X}_{n+1} dado \mathbb{Y}_n . Entonces existe una única medida de probabilidad Q_v sobre \mathbb{Y} tal que, para cada rectángulo medible $B_0 \times \dots \times B_n$ en \mathbb{Y}_n ,

$$\begin{aligned} Q_v(B_0 \times \dots \times B_n) &= \int_{B_0} v(dx_0) \int_{B_1} Q_0(dx_1 | x_0) \int_{B_2} Q_1(dx_2 | x_0, x_1) \\ &\quad \dots \int_{B_n} Q_{n-1}(dx_n | x_0, \dots, x_{n-1}). \end{aligned}$$

Más aun, para cada función medible no-negativa u en \mathbb{Y} , la función

$$x \mapsto \int u(y) Q_x(dy)$$

es medible sobre \mathbb{X}_0 , donde Q_x denota a Q_v cuando v es la probabilidad concentrada en $x \in \mathbb{X}_0$.

Demostración A.1 Véase [6]. ■

Apéndice B

Códigos

Los siguientes códigos fueron realizados y ejecutados en el programa MATLAB R2011b.

B.1. Código Para M/M/1 Homogéneo

El código que se muestra a continuación simula un sistema de colas visto en la Sección 2.4, cuando éste es ejecutado, arroja el promedio que esperaría un cliente en la cola y la longitud media de la cola. Para esto, λ tomó el valor de 8/9 y μ se hizo igual a 1, pero pueden tomar valores cualesquiera, siempre y cuando se cumpla la condición $\frac{\lambda}{\mu} < 1$.

```
%calcula la longitud media de la cola
%y el tiempo medio de espera en la cola
lambda=8/9;mu=1; tt=1000; T=10000;
t=0;E=0;Tc=0;Ts=Inf;Lq=0;
VLq=[];Vt=[]; TlWq=[];Tswq=[];
Wq=[]; Suma=0; TlWq(1)=0; suma1=0;
for k=1:tt
while t<T
    if Tc<Ts
        t=Tc; %Llegada de un cliente
        if E==0 %Servidor Libre
            E=1;
            ts=expinv(rand,1/mu);%Tiempo de servicio al cliente.
            Ts=t+ts; %Instante de tiempo en que el cliente saldrá.
            Tswq=[Tswq,Ts];
        else
```

```

Lq=Lq+1; %servidor ocupado y cliente a la cola.
VLq=[VLq,Lq]; %Guardamos información de la evolución de la cola.
Vt=[Vt,t];

end

tc=expinv(rand,1/lambda); %Tiempo que tardará en producirse la siguiente llegada
Tc=t+tc; %Instante de tiempo en que se producirá la siguiente llegada
TlWq=[TlWq,Tc];
else %Salida de un cliente
t=Ts;
if Lq>0
Lq=Lq-1; %Siguiete cliente atendido y la cola decrece.
VLq=[VLq,Lq];
Vt=[Vt,t];
ts=expinv(rand,1/mu); %Tiempo que tarda en dar servicio.
Ts=t+ts; %Instante de tiempo en que el cliente saldrá.
Tswq=[Tswq,Ts];
else
E=0; %No hay cola luego el servidor queda libre
Ts=Inf; %No hay cliente luego nunca saldrá un cliente servido
end
end

end

q=length(Tswq);
for j=1:q
Wq(j)=Tswq(j)-TlWq(j);
end
Wq=cumsum(Wq);
suma1=suma1+Wq(end)/q;
end
%Cálculo del tamaño esperado de la cola.
for i=1:(length(VLq)-1)
Suma=Suma+VLq(i)*(Vt(i+1)-Vt(i));
end
stairs(Vt,VLq)
xlabel('tiempo');
ylabel('número de clientes en la cola');
disp(Suma/T) %Número medio de clientes en la cola
disp(suma1/tt) %Promedio de espera en la cola

```

El siguiente código arroja el promedio que esperaría un cliente en el sistema y la longitud media del sistema completo. Para esto, $\lambda = 1$ y $\mu = 2$, pero, como antes, se pueden cambiar los valores de λ y μ siempre y cuando se cumpla la condición de no saturación $\frac{\lambda}{\mu} < 1$.

```

%calcula la longitud y el tiempo de espera del sistema
t=1000; l=1; mu=2; n=100;
T=[]; S=[]; Ts=[]; ins=[];
T(1)=0; S(1)=0; Ts(1)=0; W=[];
suma=0; e=[]; e1=[]; suma1=0;

```

```

for k=1:n
for j=2:t
    T(j)=expinv(rand,1/l); %tiempos entre llegadas
    S(j)=expinv(rand,1/mu); %tiempos de servicio
end
Tl=cumsum(T); %instantes de llegadas al sistema de cada cliente
for j=2:t
    Ts(j)=max(Tl(j),Ts(j-1))+S(j); %instantes de salidas del sistema
end

for i=1:t
ins(i)=Tl(i);
e(i)=1;
end
for j=(t+1):(2*t)
    if Ts(j-t)<Tl(t)
        ins(j)=Ts(j-t);
        e(j)=-1;
    end
end
m=length(ins); V=ins;
for i=1:m % Ordenamiento creciente
for j=1:m-i
if V(j)>V(j+1)
aux=V(j);
V(j)=V(j+1);
V(j+1)=aux;
end
end
end

for i=2:m
    for j=2:t
        if V(i)==ins(j)
            e1(i)=1;
            break
        else
            e1(i)=-1;
        end
    end
end
e1(1)=1;
e2=cumsum(e1);
ss=cumsum(e2);
suma=suma+ss(length(e2))/length(e2);
W=Ts-Tl;
W=cumsum(W);
suma1=suma1+W(end)/length(W);
end

```

```
stairs(V,e2)
xlabel('tiempo');
ylabel('número de clientes en el sistema');
disp(suma/n) %Número medio de clientes en el sistema
disp(suma1/n) %Tiempo medio de espera en el sistema
```

B.2. Subrutina Para Generar T_s

El siguiente código genera el valor de una variable aleatoria T_s , definida como el tiempo de la primera llegada después del instante s (ver Sección 3.3)

```
%subrutina para generar Ts
function Ts=genTs(lambda,lambdat,s)
t=s;
w=1;
while w==1
    t=t-log(rand)/lambda;
    v=rand;
    if v<=(lambdat/lambda)
        Ts=t;
        w=0;
    end
end
end
```

B.3. Código Para M/M/1 no Homogéneo

El siguiente código simula un sistema de colas no homogéneo, con $\lambda = 1$ y $\mu = 2$. Mas los parámetros pueden tomar valores distintos, sólo basta cambiarlos en la primera línea de este código, T es el tiempo de paro.

```
lambda=1; mu=2; T=100;
t=0; NA=0; ND=0; lambdat=[]; Tp=[];
A=[]; D=[]; TOTAL=[]; i=1;l=1; n=0;
for j=1:T
    v=-log(rand)/lambda;
    if v<=lambda
        lambdat=[lambdat,v];
    end
end
tA=genTs(lambda,lambdat(1),0); tD=inf;
for i=1:T
    if tA<=tD && tA<=T
        t=tA; %Nos movemos hasta el tiempo tA
        NA=NA+1; %Pues hay una llegada adicional en el instante tA
```

```

n=n+1;      %Se tiene un cliente más
tA=genTs(lambda,lambdat(1+1),1); %Tiempo de la siguiente llegada
l=l+1;
if n==1
    Y=-log(rand)/mu;
    tD=t+Y;  %Pues el sistema ha quedado vacío
end
A(NA)=t;   %Pues el cliente NA llega en el instante t.
elseif tD<tA && tD<=T
    t=tD;
    n=n-1;
    ND=ND+1; %Pues ha ocurrido una salida en instante t
    if n==0
        tD=inf;
    else
        Y=-log(rand)/mu;
        tD=t+Y;
    end
    D(ND)=t; %Pues el cliente ND acaba de salir
elseif min(tA,tD)>T && n>0
    t=tD;
    n=n-1;
    ND=ND+1;
    if n>0
        Y=-log(rand)/mu;
        tD=t+Y;
    end
    D(ND)=t;
elseif min(tA,tD)>T && n==0
    Tp=[Tp,max(t-T,0)];
end
end
q=min(length(A),length(D));
for i=1:q
    TOTAL(i)=D(i)-A(i);
end
TOTAL=cumsum(TOTAL);
disp(A) %Imprime las llegadas de cada cliente
disp(D) %Imprime los tiempos en que terminaron su servicio
disp(TOTAL(end)/length(TOTAL)) %Imprime el promedio en que son servidos.

```

B.4. Código Para M/M/2 no Homogéneo

El siguiente código simula un sistema de colas no homogéneo, para el modelo $M/M/2$ (ver 3.3.2).

```

lambda=.9; mu=1; t=0; NA=0; T=15;
lambdat=zeros(1,T+1); A=[]; D=[]; j=1;

```

```

p=1; Es=zeros(1,3); C1=0; C2=0;
while lambdat(T+1)==0
    v=-log(rand)/lambda;
    if v<=lambda
        lambdat(j)=v; j=j+1;
    end
end
tA=genTs(lambda,lambdat(p),0); t1=inf; t2=inf;
for j=1:T
    if tA<min(t1,t2)
        t=tA; NA=NA+1;
        tA=genTs(lambda,lambdat(p+1),p);
        p=p+1; A(NA)=t;
        if Es==0
            Es(1)=1;Es(2)=NA; Es(3)=0;
            Y=-log(rand)/mu; t1=t+Y;
        elseif Es(1)==1 && Es(2)~=0 && Es(3)==0
            Es(1)=2;Es(3)=NA; Y=-log(rand)/mu; t2=t+Y;
        elseif Es(1)==1 && Es(2)==0 && Es(3)~=0
            Es(1)=2; Es(2)=NA; Y=-log(rand)/mu; t1=t+Y;
        elseif Es(1)>1
            Es(1)=Es(1)+1;
        end
    elseif t1<tA && t1<=t2
        t=t1; C1=C1+1; D(Es(2))=t;
        if Es(1)==1
            Es=zeros(1,3);t1=inf;
        elseif Es(1)==2
            Es(1)=1;Es(2)=0; t1=inf;
        elseif Es(1)>2
            m=max(Es(2),Es(3));
            Es(1)=Es(1)-1; Es(2)=m+1;
            Y=-log(rand)/mu; t1=t+Y;
        end
    elseif t2<tA && t2<t1
        t=t2;C2=C2+1; D(Es(3))=t;
        if Es(1)==1
            Es=zeros(1,3); t2=inf;
        elseif Es(1)==2
            Es(1)=1;Es(3)=0;t2=inf;
        elseif Es(1)>2
            m=max(Es(2),Es(3));
            Es(1)=Es(1)-1; Es(3)=m+1;
            Y=-log(rand)/mu; t2=t+Y;
        end
    end
end
end
A %imprime los tiempos de llegadas
D %imprime lso tiempos de salidas

```

C1 %número de clientes atendidos por el servidor 1
 C2 %número de clientes atendidos por el servidor 2

B.5. Código Para el Costo Total Esperado α -Descontado

Para la simulación del Costo Total Esperado, se muestran tres programas que son funciones, es decir que en algún momento son llamados por el programa raíz o fuente, y este se muestra al final de esta sección.

El código que se muestra a continuación genera la acción para cualquier estado $(x, y, z) \in \mathbb{S}$.

```
%subrutina para generar la acción a
function A=genA(x,y,z)
if x>=1 && y==1 && z==1
    A=1;
elseif y==1 && z==0
    if x>=1
        A=1;
    else A=0;
    end
elseif y==0 && z==1
    if x>=1
        A(1)=0;
        A(2)=1;
    else A=0;
    end
elseif x>=1 && y==0 && z==0
    A=0;
else
    error('estado no accesible');
end
end
end
```

El siguiente código genera el costo para cualquier estado $(x, y, z) \in \mathbb{S}$.

```
%subrutina para generar el Costo
function C=genC(x,y,z,g)
a=genA(x,y,z);
if length(a)==1
    C=x*z+g*(1-y)*z*a;
else
    C(1)=x*z+g*(1-y)*z*a(1);
    C(2)=x*z+g*(1-y)*z*a(2);
end
end
end
```

82 B.5. Código Para el Costo Total Esperado α -Descontado

Este código es una función que es mandada llamar por el programa raíz; este calcula el costo total esperado $V(x, y, z)$, $(x, y, z) \in \mathbb{S}$, para distintos valores de α , λ , μ , ν , γ .

```
function V=vn(x,y,z,al,l,mu,v,g,N)
b=1+mu+v; L=(1+v)/(al+1+v); B=b/(al+b);
l1=1/(1+v); v1=v/(1+v);
c=genC(x,y,z,g); a=genA(x,y,z);
if N>0 && x>0
    if length(a)==1 && a==0
        V=c(1)+L*(vn(x+1,0,0,al,l,mu,v,g,N-1)*l1
            +vn(x,0,1,al,l,mu,v,g,N-1)*v1);
    elseif length(a)==1 && a==1
        V=c(1)+B*(vn(x-1,1,0,al,l,mu,v,g,N-1)*mu/b
            +vn(x+1,1,0,al,l,mu,v,g,N-1)*1/b
            +vn(x,1,1,al,l,mu,v,g,N-1)*v/b);
    else
        V=min(c(1)+L*(vn(x+1,0,0,al,l,mu,v,g,N-1)*l1
            +vn(x,0,1,al,l,mu,v,g,N-1)*v1),
            c(2)+B*(vn(x-1,1,0,al,l,mu,v,g,N-1)*mu/b
            +vn(x+1,1,0,al,l,mu,v,g,N-1)*1/b
            +vn(x,1,1,al,l,mu,v,g,N-1)*v/b));
    end
elseif N>0 && x==0
    V=c(1)+L*(vn(0,0,1,al,l,mu,v,g,N-1)*v1
        +vn(1,0,0,al,l,mu,v,g,N-1)*l1);
elseif N==0
    V=0;
else
    error('no válido');
end
end
```

Este último es el programa fuente o raíz que calcula el costo total esperado hasta la N -ésima etapa y a su vez imprime, en forma de matriz, hasta los estado $V(n, y, z)$, es decir $V_N(x, y, z)$ para distintos valores de α , λ , μ , ν , γ . Los parámetros pueden tomar distintos valores así como el valor de n y N .

```
al=1/2; l=1/5; mu=1/2; v=1/2; g=2; %condiciones l<mu, (g,v,al)>0
n=6; N=8; V=zeros(n+1,4); %n=#estados, N=iteraciones
beta1=(al+1+mu-sqrt((al+1+mu)^2-4*1*mu))/2*1; %raíces del polinomio
beta2=(al+1+mu+sqrt((al+1+mu)^2-4*1*mu))/2*1;
beta=min(beta1,beta2); V(1,1)=inf; V(1,4)=inf;%estados no accesibles
for i=2:n+1
    for j=1:4
        if j==1
            V(i,j)=vn((i-1),0,0,al,l,mu,v,g,N);
        elseif j==2
            V(i,j)=(V(2,1)*(1/(al+1))+v*(mu-1)/al^2)*beta^(i-1)
```

```

                                +v*(i-1)/a1+v*(1-mu)/a1^2;
elseif j==3
    V(i,j)=(i-1)+min(V(i,1),g+V(i,2));
else
    V(i,j)=(i-1)+V(i,2);
end
end
end
V(1,2)=(1/(a1+1))*V(2,1); V(1,3)=V(1,2);
V
```


Bibliografía

- [1] Medhi, J. *Stochastic models in queueing theory*, 2th ed., Florida, Academic Press, (1991).
- [2] Gross, Donald and Harris, Carl M. *Fundamentals of queueing theory*, 3th ed., New York, Wiley, (1998).
- [3] Ross, Sheldon M. *Simulación*, 2a ed., México, Prentice Hall, (1999).
- [4] Banks, Jerry and Carson, John S. *Discrete-event system simulation*, New Jersey, Prentice Hall, (1984).
- [5] Hillier, Frederick S. and Lieberman, Gerald J. *Introducción a la investigación de operaciones*, 9a ed., Mc Graw Hill, (2010).
- [6] R.B Ash and Doléans-Dade C. A. *Probability and Measure Theory*, San Diego, Academic Press Elsevier, (2005).
- [7] Puterman, M. L. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, Nueva York, Wiley, (1994).
- [8] Hassin, R. and Haviv M. *To queue or not to queue: Equilibrium Behavior in Queueing Systems*, Boston, Kluewer Academic Publisher, (1990).
- [9] Doshi, B. *Queueing systems with vacations - a survey*, Queueing Syst 1:29-66, (1986).
- [10] Teghem, J. Jr. *Control of the service process in a queueing system*, European J. of Opns. Res. 23:141-158, (1986).

-
- [11] Gelenbe, E. Mitrani, I. *analysis and synthesis of computer systems*, London, Academic Press, (1980).
- [12] Kella, O. *Optimal control of the vacation scheme in an M/G/1 queue*, Opns. Res. 38:724-728.
- [13] Heyman, D.P. Sobel, M.J. *Stochastic models in operations research, Stochastic Optimization*, New York, McGraw-Hill Book Company, (1984).
- [14] Yadin, M. Naor, P. *Queueing systems with a removable service station*, Opns. Res. Quat. 14:393-405 (1963).
- [15] Talman, A.J.J. *A simple proof of the optimality of the best n-policy in the M/G/1 queueing control problem with removable server*, Statistica Neerlandica 33:143-150 (1979).
- [16] Lee, H-S. Srinivasan, M.M. *Control policies for the $M^x/G/1$ queueing system*, Mgmt. Sci. 35: 708-721 (1989).
- [17] Altman, E. Nain, P. *Optimality of a threshold policy in the M/M/1 queue with repeated vacations*, INRIA Report, RR-1419, (1991).
- [18] Schäl, M. *Conditions for optimality in dynamic programming and for the limit of the n-stage optimal policies to be optimal*, Z. Wahrscheinlichkeitsth 32:179-196, (1975).
- [19] Howard, R. A. *Semi-Markovian Decision Processes*, Proc. Intern. Stat. Inst, (1963).
- [20] Lippman, S. A. *On dynamic programming with unbounded rewards*, Mgmt. Sci. 21: 1225-1233, (1975).
- [21] Altman, E. and Nain, P. *Optimality of a Threshold Policy in the M/M/1 Queue with Repeated Vacations*, Math. Meth. Op. Res. 44:75-96, (1996).
- [22] Aguirre, M. *Procesos de Control Semi-Markovianos en Líneas de Espera*, Puebla,(2012).

-
- [23] Hernández, O. Lasserre, J.B. *Discrete-Time Markov Control Processes: Basic optimality criteria*, Springer-Verlag, New York, (1996).
- [24] Hernández, O. Lasserre, J.B. *Further Topics on Discrete-Time Markov Control Processes*, Springer-Verlag, New York, (1999).
- [25] Rosas, L.C. *Procesos de Control Semi-Markovianos con Distribución de Tiempo de Permanencia Desconocida*, Sonora, (2006).
- [26] Dshalalow J. H. *Real analysis: an introduction to the theory of real functions and integration*, Chapman and Hall/crc, Florida, (2001).
- [27] Bellman, R. E. *Dynamic Programming*, (1957)