

29.
36

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

SIMULACION DIGITAL COMO
ALTERNATIVA A LA SOLUCION
DE PROBLEMAS EN SISTEMAS
OPERATIVOS

Tesis que presenta:
MATEO JAVIER SANCHEZ FLORES
para obtener el título de MATEMÁTICO

MÉXICO, D.F.

1984



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

CONTENIDO

Pág.

Introducción .

1

Capítulo I. Conceptos básicos y terminología en simulación

Introducción

7

1.1 El concepto de sistema

8

1.2 Sistemas determinísticos y sistemas estocásticos

10

1.3 Sistemas continuos y sistemas discretos

11

1.4 Modelos para sistemas

13

1.5 Modelos físicos y modelos matemáticos

15

1.6 Simulación de sistemas

16

1.7 Aplicación de la simulación ventajas y desventajas

18

1.8 El proceso de la simulación

19

Referencias

20

Capítulo II. Conceptos de probabilidad en simulación

Introducción

22

2.1 Probabilidad

24

2.2 Teoría de conjuntos, eventos compuestos

27

2.3 Probabilidad condicional, eventos independientes

30

2.4 Distribuciones discretas

32

2.5 Distribuciones continuas

37

2.6 Funciones de una variable aleatoria, momentos, función generatriz

42

2.7	Distribuciones conjuntas	48
2.8	Algunas distribuciones comunes	52
2.8.1.	Distribución Bernoulli	52
2.8.2	Distribución Binomial	53
2.8.3	Distribución Geométrica	55
2.8.4	Distribución Poisson	56
2.8.5	Distribución Uniforme	57
2.8.6	Distribución Normal	58
2.8.7	Distribución Exponencial	60
2.8.8	Distribución Chi-Cuadrada	62
2.8.9	Distribución t-de-Student	63
2.8.10	Distribución F	63
	Referencias	64

Capítulo III Estimación y Pruebas estadísticas.

	Introducción	66
3.1	Estimación	67
3.2	Prueba de hipótesis	69
3.3	La prueba t	77
3.4	La prueba F	81
3.5	La prueba chi-cuadrado de bondad de ajuste	83
3.6	La prueba Kolmogorov - Smirnov	87
	Referencias	88

Capítulo IV Generación de números aleatorios

	Introducción	90
4.1	Números pseudoaleatorios	91
4.2	Algoritmos para generar números pseudoaleatorios	94
4.2.1	Método de los medios cuadrados	94
4.2.2	Método Congruencial lineal	96
4.3	Pruebas y validez de secuencias pseudoaleatorias	100
4.4	Generación de variables aleatorias	101
4.4.1	Método de inversión	101
4.4.2	Método de rechazo	102
4.5	Programoteca de generación de distribuciones	104
4.5.1	Distribución Uniforme	104
4.5.2	Distribución Bernoulli	105
4.5.3	Distribución Binomial	106
4.5.4	Distribución Exponencial	107
4.5.5	Distribución Erlang	109
4.5.6	Distribución Geométrica	110
4.5.7	Distribución Normal	111
4.5.8	Distribución Poisson	112
4.6	Formulas estadísticas	115
	Referencias	116

Capítulo V Simulación de sistemas discretos

	Introducción	119
--	--------------	-----

	Pág.
5.1 Métodos que manejan el tiempo	120
5.1.1 Método rastreo periódico	120
5.1.2 Método rastreo de eventos	123
5.2 Generación de entidades	124
5.3 Eventos y su sincronización	127
5.4 Manejo de colas y listas	129
5.5 Resultados de una simulación	135
5.6 Lenguajes de programación para simulación	138
Referencias	143

Capítulo VI Solución por simulación, de problemas en sistemas operativos

Introducción	145
6.1 Modelo de un centro de servicio	146
6.2 Sistema Round Robin	162
6.3 Sistema de memoria virtual	179
Referencias	195

Capítulo VII Conclusiones y extensión

7.1 Conclusiones	197
7.2 Extensiones, complementos y trabajos relacionados	198
Apéndices	201

INTRODUCCIÓN

Esta tesis está constituida de seis capítulos, y tiene como objetivo principal el familiarizar al investigador con inclinación en el área de los sistemas operativos, con una herramienta disponible para la solución de algunos problemas que aparecen en el área de sistemas operativos. Nos referimos a la simulación digital como alternativa de solución de problemas en esta área.

Para lograr lo anterior se desarrolla la teoría necesaria sobre simulación discreta. Esto comprenderá la implementación de algunas rutinas necesarias en cualquier simulación sobre sistemas operativos. Además se atacan algunos problemas de sistemas operativos por medio de simulación.

El uso de la simulación es muy común como aplicación del computador. Existen varias razones de lo anterior, entre otras, el hecho de que la simulación permite una visualización, de una posible conducta del sistema aún antes de que esté funcionando (éste es el caso de los sistemas operativos sobre computadoras digitales). Además, la simulación permite la comparación de varias estrategias de operación de un sistema ya implantado, sin afectar las funciones actuales del sistema. Por otra parte, la simulación permite una compresión del tiempo de la operación del sistema. Esto es, es posible simular horas ó años de actividad de un sistema, en pocos minutos de tiempo de computador.

Es oportuno mencionar que en ocasiones, la

simulación, por la ausencia de una solución analítica para determinado problema, es usada como la única o factible alternativa de solución.

Una simulación será tan buena como lo sean las técnicas usadas en la construcción de ésta, y la validez de los resultados obtenidos de una simulación se verán influidos por factores tales como las técnicas usadas en la recolección de datos y las usadas al resumir los mismos. En este trabajo se presenta una discusión sobre la teoría requerida para el desarrollo adecuado de una simulación, para un cierto tipo de sistemas (los sistemas discretos).

Se ha intentado presentar una discusión sobre los aspectos básicos de simulación, todo con la inclinación a tener aplicaciones en sistemas operativos, disciplina que hace un uso extenso de la simulación.

Estos aspectos básicos incluyen fundamentos de sistemas, conceptos de probabilidad y estadística, generación de números aleatorios, simulación de sistemas discretos y finalmente se hace una aplicación de todo lo anterior en el campo de sistemas operativos.

El contenido de cada capítulo es el siguiente:

El capítulo 1 introduce los conceptos básicos y la terminología en simulación, se discuten los términos sistema, modelo y simulación.

Los capítulos 2 y 3 dan los elementos

de probabilidad y estadística necesarias para el estudio que el capítulo 5 hace de la simulación de sistemas discretos.

El capítulo 4 trata la generación de números aleatorios, discute las técnicas más comunes en la generación de secuencias aleatorias, además se estudian las pruebas estadísticas aplicadas a estas secuencias. Para las distribuciones de uso común se forma una programación donde se encuentran las rutinas que generan secuencias aleatorias con una distribución específica. La programación se hace en el lenguaje FORTRAN por la facilidad de cálculos numéricos en este lenguaje y la disponibilidad del lenguaje en el laboratorio de computación del Departamento de Matemáticas de la Facultad de Ciencias.

La simulación de sistemas discretos es discutida en el capítulo 5. Además, se discuten las facilidades de los lenguajes de programación existentes, para trabajos de simulación. Se da una serie de referencias acerca de los lenguajes con orientación a la simulación de sistemas discretos.

El capítulo 6 viene a ser una aplicación de todo lo discutido en los capítulos anteriores. La inclinación que se da a las aplicaciones es el área de los sistemas operativos. Se presenta una serie de problemas en esta área y su solución por simulación. El primer problema a diferencia de los restantes se presenta con una solución analítica con el objeto de mostrar que en ocasiones la simulación parece ser la única alternativa de solución.

Se termina esta tesis presentando una serie de conclusiones a las que se llega, así como diferentes posibilidades de extensión o complemento para este trabajo.

CAPITULO I

Introducción

El concepto de modelo, sobre el cual se basa la simulación, se ha usado desde hace muchos años. Algunos ejemplos de modelos históricos son la segunda Ley de Newton, $F = m \cdot a$, relacionando la fuerza aplicada a un cuerpo con su aceleración; el modelo de Kepler del universo en el cual se postula que el sol más que la tierra es el centro del universo. Así, el concepto de modelo tiene tiempo de haber aparecido y con el advenimiento de las computadoras se ha incrementado el número de problemas resueltos con éxito con el uso de la simulación.

El término simulación es definido por Shannon [1.1] como "El procedimiento de diseñar un modelo computarizado de un sistema y conducir experimentos con este modelo con el propósito ya sea de entender la conducta del sistema o evaluar varias estrategias para la operación del sistema". Aunque la literatura da muchas definiciones para el término "simulación", esta definición parece comprender los aspectos más importantes de este proceso usado para resolver problemas. En este capítulo se discuten los conceptos de modelo, sistema y simulación.

1.1 El Concepto de Sistema

El término sistema es usado en una gran variedad de formas, esto hace difícil producir una definición lo suficientemente general como para cubrir las distintas formas de darle uso a este término, lo que se hace aquí es dar una definición de diccionario y ubicar a ésta en el contexto de simulación.

El término sistema es definido como una "colección de objetos relacionados lógicamente". Cuando se usa en el contexto de un estudio de simulación, el término sistema generalmente se refiere a una colección de objetos con un conjunto bien definido de interacciones entre ellos. Un ejemplo es el sistema solar. Los planetas y el sol forman la colección de objetos del sistema; la fuerza gravitacional es una de las interacciones entre los objetos.

Estando en el contexto de simulación, el término "ENTIDAD" se usa para denotar un objeto de interés en el sistema; el término "ATRIBUTO" denota una propiedad de una entidad. Cualquier proceso que ocasione cambio en el sistema lo llamaremos una "ACTIVIDAD". El término "ESTADO DE SISTEMA" lo usaremos para denotar todas las entidades, los atributos y actividades del sistema en algún punto del tiempo.

En la figura 1.1 se ilustran ejemplos de sistemas con algunas entidades, atributos y actividades.

SISTEMA	ENTIDAD	ATRIBUTO	ACTIVIDAD
Tráfico de autos.	autos	velocidad, distancia.	MANEJAR
Banco	clientes	balance, Edo. de cuenta	DEPOSITAR
Comunicaciones	Mensajes	longitud, prioridad	TRANSMITIR

Figura 1.1 Ejemplos de Sistema

Si consideramos el movimiento de autos como un sistema de tráfico "Los Autos" pueden ser vistos como entidades, cada uno teniendo como atributo su velocidad y su distancia recorrida. Una actividad es el manejo de los autos pues este hecho causa cambios en el sistema.

En el caso del sistema bancario, los clientes del banco son las entidades, con el balance de su cuenta y el estado de su crédito sus atributos. Una acción que cumple con ser una actividad sería hacer un depósito.

El sistema de comunicaciones, último en la ilustración tiene como entidades a los mensajes, atributos su longitud y prioridad y una transmisión se puede considerar como una actividad.

En la figura 1.1 no se muestra una lista completa de todas las entidades, atributos y actividades para los sistemas. En efecto, una lista completa no se puede elaborar sin conocer el propósito de la descripción del sistema, dependiendo del propósito, algunos

aspectos del sistema podrían resultar de interés y se determina así la necesidad de ser identificados.

Existen varias maneras de clasificar sistemas. Una clasificación obvia distingue entre los sistemas naturales y los que son hechos por el hombre. El sistema solar es un sistema natural mientras que el sistema de tráfico de la figura 1.1 es hecho por el hombre.

Otras clasificaciones que se pueden hacer son; determinístico contra estocástico, continuo contra discreto, cerrado contra abierto. En las siguientes dos secciones se hace una discusión de lo anterior.

1.2 Sistemas determinísticos y sistemas estocásticos

Un sistema determinístico es un sistema en el cual el nuevo estado está completamente determinado por el estado previo y por la actividad. Este tipo de sistemas se ilustran en la figura 1.2, donde E_0 refiere al estado del sistema antes de la actividad A y E_1 se refiere al estado del sistema después de la ocurrencia de la actividad.

Un sistema estocástico contiene una cierta carga de aleatoriedad en su transición de un estado a otro. Un sistema estocástico es ilustrado en la figura 1.3 donde E_n y E_m son dos posibles estados en los que el sistema puede entrar después de estar en E_0 y bajo la actividad A . Estos sistemas son no determinísticos en el sentido de que el próximo estado no puede ser predicho aun que

conozcamos el estado actual y la actividad.



Figura 1.2 Un sistema determinístico

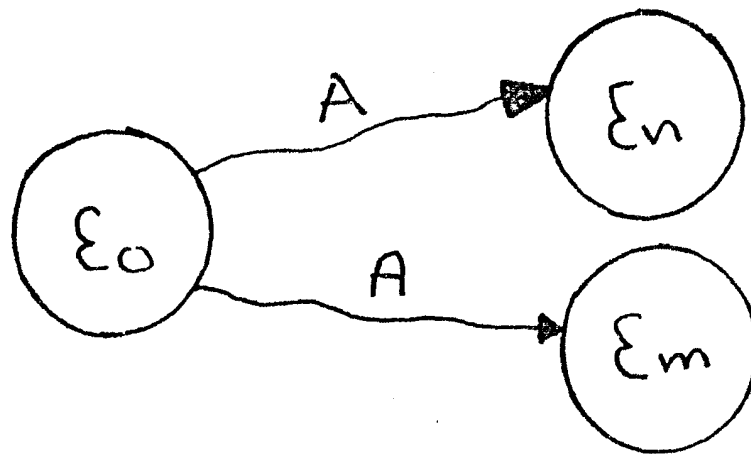


Figura 1.3 Un sistema Estocástico

El sistema bancario puede ser considerado un sistema estocástico ya que la actividad "DEPÓSITO" por la aleatoriedad de su monto puede enviar el sistema a diferentes estados.

1.3 Sistemas Continuos y Sistemas Discretos

Los términos continuo y discreto se aplican a sistemas refiriéndose a la naturaleza de los cambios con respecto al tiempo. Un sistema en el cual ocurren los cambios de manera continua en el tiempo son llamados

sistemas continuos; los sistemas en los que ocurren cambios en puntos del tiempo son conocidos como sistemas discretos.

Pocos sistemas son del todo continuos o discretos. En los ejemplos de la figura 1.1 podemos decir que el sistema de tráfico es continuo ya que los cambios, cuando ocurren, se hacen de manera continua, tanto el cambio de velocidad como el cambio en la distancia. El sistema bancario acepta cambios discretos ya que los depósitos ocurren en puntos del tiempo.

Estos sistemas parecen ser totalmente continuos y discretos respectivamente, sin embargo, los mismos sistemas con más entidades, atributos y actividades pueden dejar de ser totalmente continuos o discretos, como quiera que sea habrá un tipo de cambios que predomine y así se puede seguir clasificando como continuo o discreto.

Generalmente la descripción de un sistema continuo será en forma de ecuaciones mostrando cómo los atributos del sistema cambian continuamente con el tiempo.

La descripción de un sistema discreto se centra en la localización y manejo de eventos que producen cambios en el sistema. Esto último se discutirá ampliamente en el capítulo V.

1.4 Modelos para sistemas

Para estudiar sistemas, una alternativa es experimentar con el sistema mismo, sin embargo, hay ocasiones en las que se desea predecir el comportamiento de un sistema aún antes de que éste exista, aquí claramente no podemos aplicar la alternativa antes mencionada, una solución que se intenta en algunas ocasiones es la de construir un cierto número de prototipos y probar sobre éstos aunque aquí una de las principales desventajas es lo costoso del método y su a veces gran consumo de tiempo, aún con un sistema existente puede ser impráctico o a veces imposible experimentar con el sistema, por ejemplo en el sistema de tráfico parece impráctico estar midiendo velocidades y distancias ya que resultaría esto de alguna forma costoso.

Consecuentemente el estudio de sistemas se conduce generalmente con un modelo del sistema. Definimos un "MODELO" de sistema como la representación de la información acerca de un sistema, para los propósitos de muchos estudios no es necesario considerar todos los detalles de un sistema; así, un modelo no es solamente un suplente del sistema sino además es una simplificación. Cuando ha sido dado el propósito del estudio de un sistema, esto determina la naturaleza de la información que será recolectada en el modelo, así pues podríamos tener varios modelos para un sistema producidos por diferentes propósitos de estudio del sistema.

Una gran variedad de tipos de modelos se

usan en el estudio de sistemas y se han clasificado en diferentes maneras. La clasificación se hace a veces en términos de la naturaleza del sistema que es modelado, tal como continuo contra discreto o determinístico contra estocástico.

Para nuestros propósitos serán tratados como modelos físicos y modelos matemáticos. Una segunda distinción para ambos, se hace entre modelos estáticos y modelos dinámicos. En el caso de los modelos matemáticos una tercera distinción es la técnica empleada para resolver el modelo. Esta distinción se hace entre métodos analíticos y métodos numéricos. Como se discutirá en la sección 1.6 la simulación es una técnica numérica usada en la solución de modelos matemáticos dinámicos. En la sección siguiente se hace una breve distinción de los tipos de modelos mencionados y que se encuentran resumidos en la figura 1.4.

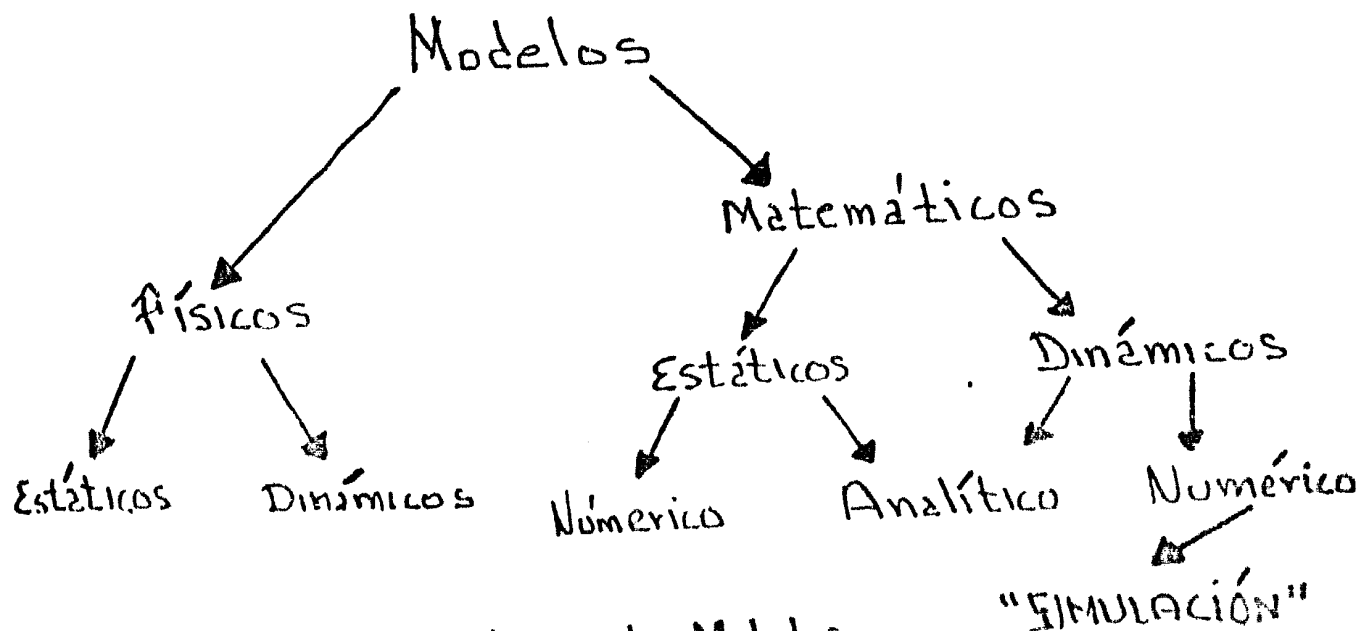


Figura 1.4 Tipos de Modelos

1.5 Modelos Físicos y Modelos Matemáticos

Cuando se discutieron los modelos para sistemas en la sección previa no se intentó hacer implícito que un modelo es necesariamente una descripción matemática de un sistema. Es posible construir modelos físicos cuyo funcionamiento representa el sistema estudiado. Los atributos de las entidades del sistema son representados por medidas físicas tal como voltaje o resistencia. Las actividades del sistema son reflejadas en leyes físicas que gobiernan el modelo.

El más conocido modelo físico es el modelo de escala, este tipo de modelo lo podemos ver en modelos arquitectónicos a escala que al mismo tiempo lo podemos ver como un modelo a escala estático o un modelo a escala de aviones o barcos que sirven para estudiar la estabilidad de los mismos, en este caso se trata de un modelo físico de escala dinámico.

Otro modelo físico es el conocido como modelo icónico, éstos son, modelos en los cuales se "parece" al sistema modelado, por ejemplo el modelo de la estructura molecular que es hecho de esferas representando a los átomos y alambres representando las ligas atómicas.

En un modelo matemático, las entidades del sistema y sus atributos son representados por variables matemáticas. Las actividades se describen por funciones matemáticas que interrelacionan las variables. Los modelos matemáticos a su vez se clasifican en estáticos y dinámicos.

Un modelo matemático estático es aquel que no permite cambios en los atributos del sistema a través del tiempo, si por el contrario el modelo permite cambios en los atributos, derivados como función del tiempo, entonces el modelo es dinámico. Un ejemplo de un modelo matemático dinámico sería el derivado de modelar el sistema de tráfico de la sección 1.1, aquí los atributos que son la velocidad y distancia cambian con el tiempo.

Los modelos matemáticos dinámicos pueden ser resueltos analíticamente pero más frecuentemente se resuelven por métodos numéricos y como es indicado en la figura 1.4 la simulación es uno de tales métodos. En el capítulo VI se presenta uno de estos modelos con solución analítica y solución por simulación.

1.6 Simulación de Sistemas

Dado un modelo matemático de un sistema, es posible en algunos casos tener una solución analítica que nos da información acerca del sistema. Cuando esto no es posible, es necesario usar métodos numéricos para resolver las ecuaciones. Se han desarrollado una gran cantidad de métodos de este tipo para resolver ecuaciones de modelos matemáticos.

En el caso de modelos matemáticos dinámicos, una técnica particular de solución numérica es la simulación de sistemas, técnica en la cual todas las ecuaciones del modelo son resueltas simultáneamente con un incremento uniforme del tiempo.

Por lo anterior, definimos la "simulación de sistemas" como la técnica de resolver el problema por seguir los cambios al sistema a través del tiempo. La definición es lo suficientemente amplia para incluir los modelos físicos dinámicos, en este caso las variables del modelo toman como valores, medidas físicas más que valores numéricos.

Para nuestro propósito, en este momento hacemos a un lado la discusión sobre modelos físicos y en adelante nos referimos a la simulación en términos de modelos matemáticos dinámicos.

En sistemas continuos donde los cambios se dan de manera continua en el tiempo, generalmente se usan sistemas de ecuaciones diferenciales para describir el sistema. Las simulaciones basadas en tales sistemas son llamadas simulaciones continuas. El computador puede usarse para resolver las ecuaciones diferenciales por uso de pequeños incrementos en el tiempo para integrar las ecuaciones.

Para sistemas discretos, donde el principal interés son los eventos, las ecuaciones son esencialmente ecuaciones lógicas que dan las condiciones para que un evento ocurra. La simulación consiste en seguir los cambios en el estado del sistema que resultan de la ocurrencia de un evento.

Tal simulación es llamada simulación discreta. Las técnicas para un buen desarrollo de una simulación discreta son tratadas en el capítulo V.

1.7 Aplicación de la simulación ventajas y desventajas

Una vez que hemos aceptado un modelo para un sistema, si este modelo es matemático y dinámico entonces ocurre en ocasiones que no se puede hallar una solución analítica del modelo en cuyo caso se hace uso de la simulación.

Al hacer uso de la simulación como alternativa de solución nos encontramos con que:

1) A través de la simulación se pueden estudiar los efectos de ciertos cambios en la información o de organización de la misma, en el comportamiento del sistema.

2) Por medio de la simulación se puede hacer una "compresión" del tiempo en el sentido de simular un período grande de tiempo en algunos minutos de máquina.

3) La simulación puede ser usada para experimentar sobre un sistema aún si no se logra tener acceso al sistema real.

De lo anterior podemos decir que la simulación nos da las siguientes ventajas:

- 1) Permite experimentación controlada
- 2) Permite análisis de sensibilidad a los cambios
- 3) Compresión del tiempo
- 4) No se afecta el sistema real

Sin embargo podríamos caer en una de las siguientes desventajas:

- 1) Un modelo de simulación podría ser costoso
- 2) El tiempo de desarrollo podría ser grande
- 3) Se podría tener una expansión del tiempo.

Una discusión más amplia de lo expuesto en esta sección puede ser encontrada en [1.2].

1.8 El Proceso de la Simulación

Terminamos este capítulo dando una serie de pasos necesarios para realizar una simulación.

1) Formulación del problema. La definición del problema a ser atacado, incluyendo el planteamiento del objetivo que se pretende alcanzar en la solución del problema.

2) La construcción del modelo. La abstracción del sistema a relaciones lógicas matemáticas de acuerdo con la formulación del problema.

3) Identificación de elementos del modelo, su especificación y la recopilación de datos.

4) Solución del modelo. La preparación del modelo para ser procesado por computador.

5) Verificación. El proceso de verificar que el programa de computador realiza lo pretendido.

6) Experimentación. La ejecución del modelo de simulación para obtener valores de salida.

7) Análisis de Resultados. Analizar los resultados para extraer conclusiones.

En los siguientes dos capítulos se hace una exposición de las herramientas de probabilidad y de estadística que son necesarias para el estudio de simulación de sistemas discretos.

Referencias

1.1) Shannon, Robert E. "Simulation: A Survey with Research Suggestions." AIEE Transactions 7, No. 3 (Septiembre 1975) : 289-301

1.2) Adkins, Gerald, and Pooch, Udo W.

"Computer Simulation: A tutorial."

Computer . 10 No. 4 (Abril 1977) : 12-17

CAPITULO II

Introducción

Uno de los problemas que se presenta con mayor frecuencia mientras se intenta modelar sistemas del mundo real, es que pocos sistemas se exhiben como constantes, esto es, con una conducta predecible. Algunas medidas del sistema tienen cierta carga aleatoria, refiriéndonos a los sistemas mencionados en el capítulo I, algunas medidas del sistema que son aleatorias serían: El tiempo entre llegadas de clientes al banco, la longitud de un mensaje, monto de un depósito. Con estas características inherentes de algunos sistemas, algún modelo o estructura matemática se requiere para describir el sistema.

Entonces, algunas partes del sistema pueden ser descritas por modelos probabilísticos que en muchas ocasiones son desarrollados a través de experimentación. Se experimenta sobre el sistema (cuando se puede), los resultados de estos experimentos se consideran y, así, el modelo se postula en base a los resultados. En la simulación de sistemas discretos podríamos encontrar que algunos componentes son aleatorios y entonces se hace uso de la teoría de la probabilidad para tratar de simular el sistema.

En ocasiones, el sistema aún no existe, pero se sabe de un comportamiento aleatorio en éste (por ejemplo en un sistema operativo por diseñar). En estos casos generalmente se adoptan modelos probabilísticos específicos, para las medidas aleatorias (tiempos entre llegada, de espera, etc.). Se simula entonces para observar el comportamiento del sistema bajo estos supuestos.

En una situación como la descrita anteriormente, el beneficio más importante que nos ofrece la simulación es el hecho de poder probar diferentes disciplinas en la organización del sistema y lograr una optimización de parámetros (como tiempos de respuesta) aún antes de implantar un sistema.

Consecuentemente es necesario dedicar una parte de este trabajo a introducir los conceptos de probabilidad necesarios para el buen desarrollo de una simulación de sistemas discretos.

En este capítulo hacemos tal discusión, necesariamente evitando profundizar en detalles matemáticos de la teoría. Además, se intenta señalar el vínculo de cada parte desarrollada con la simulación de un sistema.

2.1 Probabilidad

Para entender el desarrollo de un modelo de probabilidad y la importancia de este modelo en la descripción de sistemas que tienen cierto comportamiento aleatorio, debemos atender antes el estudio de algunos conceptos de probabilidad.

Estos conceptos son simples e intuitivos y podrían ser discutidos fácilmente a través de ejemplos.

La probabilidad es una medida de la incertidumbre. Esta incertidumbre se maneja en ocasiones como, "probablemente", "cierta verosimilitud", "hay chance de:", esta definición, como quiera que sea, es poco precisa, para nuestros propósitos; damos aquí una definición más completa después de discutir algunos ejemplos.

Ejemplo 2.1 Consideremos el experimento de lanzar una moneda. Si suponemos una moneda lo suficientemente adecuada como para considerar sólo dos posibles resultados del experimento entonces tendremos sólo los resultados: águila o sol. ¿Podríamos predecir con certeza el resultado del experimento? La respuesta parece ser que no; el lado de la moneda que caerá es impredecible o incierto, ya que el resultado del experimento se verá influido por la construcción de la moneda, la manera de lanzarla, la regularidad de la superficie y otros factores.

Un experimento como el del ejemplo 2.1 en el cual el resultado es incierto, es llamado experi-

mento aleatorio, los resultados de este experimento (águila o sol) son llamados eventos aleatorios simples. Cada uno de estos eventos tiene alguna medida de verosimilitud o certidumbre asociada - la probabilidad de su ocurrencia. Esta medida de verosimilitud es determinada a través de experimentación. El experimento se hace algunas veces y los resultados se anotan. La frecuencia relativa de ocurrencia de un resultado se estabilizará, o tenderá a un valor límite. Este valor límite es definido como la probabilidad del evento (definición frecuentista de probabilidad).

Una vez que todos los posibles resultados (eventos) de un experimento aleatorio se determinan y una probabilidad ha sido asignada a cada uno, entonces se ha establecido un modelo probabilístico.

Ejemplo 2.2 Consideremos el experimento de lanzar un dado, los resultados del experimento pueden pensarse como el siguiente conjunto $\{1, 2, 3, 4, 5, 6\}$. Supóngase que la experimentación muestra que la verosimilitud de que una cara dada aparezca es proporcional al número de puntos en la cara (dado cargado). Entonces el modelo de probabilidad para este experimento es:

EVENTO :	1	2	3	4	5	6
PROBABILIDAD :	$1/21$	$2/21$	$3/21$	$4/21$	$5/21$	$6/21$

En el ejemplo 2.2, la probabilidad de cada evento simple es diferente. Éste es el caso general; sin embargo

existen experimentos en los que la verosimilitud o probabilidad es igual para cada evento. Si este es el caso, los eventos se dicen equiprobables.

Ejemplo 2.3 Considerérese la conocida rueda de la fortuna, con 4 divisiones del mismo tamaño; podemos pensar los resultados como numéricos, esto es, podemos decir que los posibles resultados son: $\{1, 2, 3, 4\}$. Supongamos que la experimentación indica que estos cuatro eventos son equiprobables. Entonces el modelo de probabilidad para este experimento es:

EVENTO :	1	2	3	4
PROBABILIDAD:	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$

En los tres ejemplos podemos pensar en que los resultados de los experimentos no son numéricos (pueden ser: águila o sol por ejemplo), aunque notamos que existe una evidente correspondencia numérica y sería ventajoso tener los resultados de un experimento como numéricos. Por esta razón se introduce la idea de variable aleatoria. Con cierta simpleza podemos decir que una variable aleatoria es una cantidad cuyo valor es dado por el resultado de un experimento aleatorio. Más precisamente, una variable aleatoria X es una función cuyo dominio es el conjunto de los eventos y cuyo rango es algún subconjunto de los números reales. La definición es resumida como:

$$X: \text{Eventos} \rightarrow C \subseteq \mathbb{R}.$$

En el ejemplo 2.2 se consideró implícitamente esta variable aleatoria al manejar números en vez de caras de dado.

2.2 Teoría de Conjuntos, eventos compuestos

Como definimos en la sección previa, un evento aleatorio simple es un resultado de un experimento aleatorio dado. En algunos casos, más que estar interesados en estos, el interés es por algún tipo de combinación de estos eventos. Esta sección revisa algunos de los fundamentos de la teoría de conjuntos y así como en base a los eventos simples formamos eventos compuestos (de aquí en adelante llamados simplemente eventos).

Definición 2.1 Un evento es algún subconjunto del espacio de eventos de un experimento aleatorio.

El espacio de eventos para un experimento aleatorio, es definido como la colección de todos los posibles resultados del experimento. Tal conjunto lo denotaremos por la letra griega Ω .

Ejemplo 2.5 Considérese el experimento de lanzar dos dados donde la cantidad de interés es la suma de las caras. El espacio de eventos es:

$$\Omega = \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12\} \text{ y algunos eventos}$$

son: $E_1 = \{2\}$ (cada dado cae 1)

$$E_2 = \{3, 4, 5\} \text{ (la suma es 3, 4, o 5)}$$

Esta definición para el término evento, incluye todos los posibles subconjuntos del espacio de eventos, los simples, compuestos etc.

En ocasiones nos interesa conocer la probabilidad de que algún evento no ocurra. Esto nos lleva a la siguiente:

Definición 2.2. El complemento de un evento E , denotado por \bar{E} es el conjunto contenido en Ω de los elementos que no están en E .

Existen dos formas básicas de combinar dos eventos para formar un tercer evento, éstas son: la unión y la intersección.

Definición 2.3 La intersección de dos eventos E_1, E_2 denotada por $E_1 \cap E_2$ es definida como los resultados que los eventos E_1, E_2 tienen en común. Dos eventos que no tienen resultados comunes se llaman mutuamente excluyentes.

Definición 2.4 La unión de dos eventos E_1 y E_2 denotada por $E_1 \cup E_2$, es definida como los resultados tanto de E_1 ó E_2 ó de ambos.

Ejemplo 2.6 La unión e intersección de los eventos definidos en el ejemplo 2.5 son:

$$E_1 \cap E_2 = \emptyset, \quad E_1 \cup E_2 = \{2, 3, 4, 5\}$$

Por lo tanto podemos usar estas reglas para combinar y construir una buena cantidad de eventos a partir de un conjunto inicial dado. Para cualquier evento E nótese que

$$E \cup \bar{E} = \Omega \quad \text{y} \quad E \cap \bar{E} = \emptyset.$$

Vemos claramente que la probabilidad de que alguno de los eventos en el espacio de eventos ocurra es 1 (certeza) mientras que la probabilidad de que ninguno de los eventos en el espacio de eventos ocurra, es 0 (imposible).

Propiedad 2.1 Para cualquier experimento aleatorio se cumple que :

$$P(\Omega) = 1 \text{ y } P(\emptyset) = 0.$$

Si los resultados de un experimento son equiprobables, el cálculo de la probabilidad de algunos eventos compuestos de un número de diferentes resultados es fácil (definición clásica de probabilidad).

Definición 2.5 Supongamos que un experimento tiene N resultados equiprobables. Si E es un evento compuesto de n resultados entonces la probabilidad del evento E , denotada $P(E)$, es dada por:

$$P(E) = n/N.$$

La probabilidad del complemento de E , $P(\bar{E})$ es

$$P(\bar{E}) = 1 - P(E) = 1 - n/N.$$

Ejemplo 2.7 Considérese los eventos definidos en el ejemplo 2.5. Si los resultados son equiprobables (La probabilidad de cada uno es $1/11$). Las probabilidades de E_1 y E_2 son:

$$P(E_1) = 1/11 ; P(\bar{E}_1) = 10/11$$
$$P(E_2) = 3/11 ; P(\bar{E}_2) = 8/11.$$

Existe un resultado fundamental acerca de la probabilidad de la unión y la intersección de dos eventos E_1 y E_2 .

Propiedad 2.2 Si E_1 y E_2 son dos eventos de un experimento aleatorio entonces

$$P(E_1 \oplus E_2) = P(E_1) + P(E_2) - P(E_1 \cap E_2).$$

Ejemplo 2.8 Supóngase que los resultados del experimento discutido en el ejemplo 2.5 son equiprobables. Considérese los eventos definidos en el ejemplo 2.6. Entonces

$$P(E_1 \oplus E_2) = P(E_1) + P(E_2) - P(E_1 \cap E_2) = \frac{1}{11} + \frac{3}{11} - 0 = \frac{4}{11}$$

$$P(E_1 \oplus \bar{E}_1) = P(E_1) + P(\bar{E}_1) - P(E_1 \cap \bar{E}_1) = \frac{1}{11} + \frac{10}{11} - 0 = 1$$

Nótese que en este caso (resultados equiprobables), estas probabilidades pueden ser calculadas sin hacer uso de las fórmulas de la propiedad 2.2, simplemente enumerando contenidos en cada evento y aplicando directamente la definición 2.5. La propiedad 2.2 es más útil para experimentos aleatorios que no tienen resultados equiprobables.

2.3 Probabilidad Condicional, eventos independientes

En ocasiones se tiene de antemano cierto conocimiento acerca del resultado de un experimento aleatorio. Esta información podría modificar la verosimilitud de un evento dado. Un ejemplo ayudará a aclarar lo anterior.

Ejemplo 2.9 Suponer que se arrojan dos dados. Antes de

enterarnos del resultado, alguien nos dice que la suma de las caras es par. Esta información afecta obviamente la probabilidad de que la suma sea exactamente 2. En efecto, si el dado es honesto (equiprobabilidad) entonces la probabilidad de que la suma sea 2 es $1/6$ si sabemos que la suma es par (posiblemente 2, 4, 6, 8, 10 ó 12), en tanto que esta probabilidad es $1/11$ sin tal informe. Cuando cierta información afecta la probabilidad de que ocurra un evento dado, se dice que se condiciona la probabilidad.

Definición 2.6. Si E_1 y E_2 son eventos de un experimento aleatorio, entonces la probabilidad de que ocurra E_1 , dado que ocurre el evento E_2 es llamada la probabilidad condicional de E_1 dado E_2 . Esta probabilidad es denotada por $P(E_1/E_2)$ y es calculada por:

$$P(E_1/E_2) = P(E_1 \cap E_2) / P(E_2)$$

Esta definición tiene una buena justificación intuitiva si vemos E_2 como un condicionador (reductor) del espacio de eventos. Esto es, una vez que sabemos que ocurre E_2 , reducimos el espacio total de posibles resultados. La probabilidad de E_1 sería calculada entonces en el espacio reducido.

Ejemplo 2.10. Considérese el experimento aleatorio de extraer una carta de un naipes ordinario de 52 cartas. Si después de extraerla sabemos que es roja. ¿Cuál es la probabilidad de que sea el dos de corazones rojo?

Definimos el evento E_1 como el evento de que la

carta sea el dos de corazones rojo, y E_2 como el evento de que la carta sea roja. Entonces

$$P(E_1|E_2) = \frac{P(E_1 \cap E_2)}{P(E_2)} = \frac{1/52}{1/2} = \frac{1}{26}.$$

En este ejemplo sabemos que 12 carta fue roja con lo cual condicionamos el espacio de eventos a sólo 26 en vez de 52 resultados equiprobables. El evento de que se trate del dos de corazones esta contenido en estos 26 resultados.

A veces el conocimiento de que un evento ocurra, no afecta para la ocurrencia de otro. En este caso decimos que los eventos son independientes.

Definición 2.7 Dos eventos E_1 y E_2 se dicen independientes si:

$$P(E_1|E_2) = P(E_1)$$

Combinando la definición con la propiedad 2.2 tenemos

Propiedad 2.3 Si E_1 y E_2 son independientes. Entonces

$$P(E_1 \cap E_2) = P(E_1) \cdot P(E_2)$$

2.4 Distribuciones Discretas

Como se definió anteriormente, una variable aleatoria es una función que asocia un valor numérico con cada evento aleatorio simple (resultado) de un experimento aleatorio. La variable aleatoria podría ser

ya sea discreta ó continua dependiendo del tipo de valor asignado a los resultados. Si una variable aleatoria asume un número discreto (finito o infinito numerable) de valores, es llamada una variable aleatoria discreta. En otro caso se llamara una variable aleatoria continua.

Ejemplo 2.12 El valor de la suma de dos dados lanzados, varía discretamente entre 2 y 12. La variable aleatoria que asigna esta suma a cada resultado es una variable aleatoria discreta.

Ejemplo 2.13 El tiempo transcurrido entre dos llegadas a un centro de servicio varía continuamente en $0 < t < \infty$. La variable aleatoria que asigna este tiempo a cada llegada es una variable continua.

Cuando una frecuencia o función de probabilidad es asociada con una variable aleatoria discreta, decimos que tiene definida una distribución de probabilidades discretas. Esta distribución dice cómo los resultados de un experimento aleatorio (reflejados en los valores asumidos por la variable aleatoria) se distribuyen.

Ejemplo 2.14 Sea el experimento aleatorio de lanzar un dado. Definimos X como la variable aleatoria que cuenta los puntos de la cara aparecida hacia arriba. Entonces X puede asumir los valores 1, 2, 3, 4, 5 ó 6. Supóngase que el dado esta cargado de tal manera que la probabilidad de que aparezca una cara dada es proporcional a la cantidad de puntos que tiene. La distribución de probabilidad discreta en este caso es:

x	1	2	3	4	5	6
$P(X=x)$	$1/21$	$2/21$	$3/21$	$4/21$	$5/21$	$6/21$

Esta distribución puede ser descrita gráficamente como en la figura 2.1. Este ejemplo ilustra algunas propiedades que cada función de probabilidades discretas tiene.

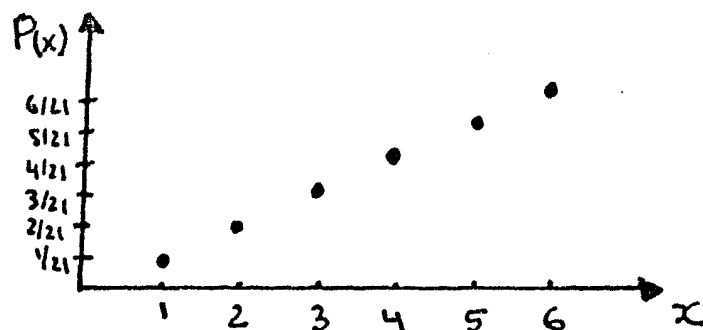


Figura 2.1 Función de probabilidad del dado cargado

Propiedad 2.4 Sea X una variable aleatoria discreta que toma los valores x_1, \dots, x_n , y sea P la función de probabilidad, con $P(x_i) = P(X=x_i)$. Entonces:

- 1) $P(x_i) \geq 0, \forall x_i, i=1, \dots, n$
- 2) $\sum_{i=1}^n P(x_i) = 1$

Otra función muy usada es la función de distribución. Esta función, denotada por $F(x)$, mide la probabilidad de que la variable aleatoria X asuma un valor más pequeño o igual a x . Esto es

$$F(x) = P(X \leq x).$$

Ejemplo 2.15 Considérese el experimento descrito en el ejemplo 2.14. La función de distribución es:

x	1	2	3	4	5	6
$F(x)$	$1/21$	$3/21$	$6/21$	$10/21$	$15/21$	$21/21$

Ilustrada gráficamente, aparece en la figura 2.2. Lo anterior apunta algunas propiedades que una función de distribución debe tener.

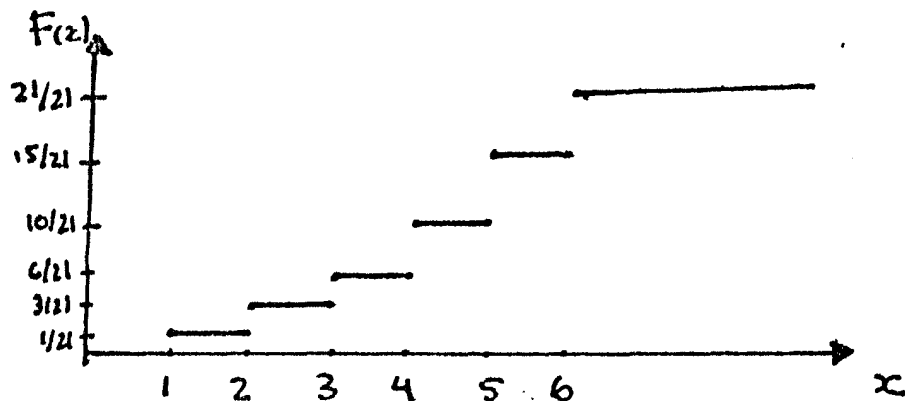


Figura 2.2 Función de Distribución para el dado cargado

Propiedad 2.5 Sea X una variable aleatoria discreta, y sea F la función de distribución asociada, entonces:

- 1) $0 \leq F(x) \leq 1$, $-\infty < x < +\infty$
- 2) Si $x_1 \leq x_2$, entonces $F(x_1) \leq F(x_2)$, (F no decrece)
- 3) $\lim_{x \rightarrow \infty} F(x) \equiv F(\infty) = 1$, $\lim_{x \rightarrow -\infty} F(x) \equiv F(-\infty) = 0$.

Nótese que la función de probabilidad P es definida sólo para valores que X asume, mientras F es definida para todo valor de x . Los escalones en la gráfica de F ocurren en valores del rango de X que tienen probabilidades no cero. Una de las formas de clasificación entre las distribuciones discretas es por su "esperanza" o valor promedio.

Definición 2.8 Sea X una variable aleatoria discreta que toma valores x_1, x_2, \dots, x_n y sea P la función de

probabilidad asociada. La esperanza o valor esperado, de la variable aleatoria X , denotada por $E(x)$ está dado por:

$$E(x) = x_1 P(x_1) + x_2 P(x_2) + \dots + x_n P(x_n) = \sum_{i=1}^n x_i P(x_i)$$

Ejemplo 2.16 Considérese el experimento discutido en el ejemplo 2.14. La esperanza de la variable aleatoria X es:

$$E(x) = 1 \cdot 1/21 + 2 \cdot 2/21 + 3 \cdot 3/21 + 4 \cdot 4/21 + 5 \cdot 5/21 + 6 \cdot 6/21 = 4.33$$

Esperamos que después de un gran número de repeticiones del experimento, el promedio tenderá al valor 4.33.

Otra característica para las variables aleatorias discretas es su varianza. Esta cantidad es una medida de dispersión y nos indica como se dispersa la variable aleatoria X con respecto a la esperanza.

Definición 2.9 Sea X una variable aleatoria discreta que toma los valores x_1, \dots, x_n , P la función de probabilidad asociada, y E el operador esperanza. La varianza de X , denotada $V(x)$ es dada por:

$$V(x) = (x_1 - E(x))^2 P(x_1) + \dots + (x_n - E(x))^2 P(x_n) = E[(x - E(x))^2].$$

Ejemplo 2.17 Considérese el experimento discutido en el ejemplo 2.14. La esperanza de la variable X fue calculada en el ejemplo 2.16. La varianza de X es

$$V(x) = (1 - 4.33)^2 (1/21) + (2 - 4.33)^2 (2/21) + (3 - 4.33)^2 (3/21) + (4 - 4.33)^2 (4/21) + (5 - 4.33)^2 (5/21) + (6 - 4.33)^2 (6/21) =$$

Como es evidente de la definición, la varianza está dada en términos de unidades cuadradas. Así, en ocasiones es poco manejable y en consecuencia se introduce el concepto de desviación estándar.

Definición 2.10 Sea X una variable aleatoria y $V(x)$ su varianza. La desviación estándar de X , denotada por $S(x)$ es dada por:

$$S(x) = [V(x)]^{1/2}.$$

El valor esperado $E(x)$ es denotado generalmente por la letra Griega μ , mientras que la desviación estándar se denota por la letra también Griega σ . La importancia de introducir estas medidas se hará evidente al avanzar en este trabajo.

2.5 Distribuciones Continuas

Una variable aleatoria continua es una variable aleatoria que asume un continuo de valores.

Ejemplo 2.18 Una manufacturera de un componente eléctrico es probada para determinar la longitud apropiada de garantía a ofrecer en los componentes. La variable aleatoria de interés es el tiempo antes de que falle el componente. Por lo tanto esta variable puede teóricamente asumir cualquier valor mayor a "0"; esto es, se trata de una variable aleatoria continua.

Así, una variable aleatoria continua puede asumir un conjunto de valores infinito, no numerable. El cálculo de probabilidades por enumeración, como

fue hecho en el caso discreto, no es posible. En efecto, cuando se trabaja con variables aleatorias continuas, se encuentra uno con la aparente contradicción de que la probabilidad de que la variable tome cualquier valor particular en su rango es "0", mientras que la probabilidad de que ésta asuma algún valor en su rango será 1. Así, para tener alguna medida de la probabilidad, necesitamos recurrir al análisis continuo de la función de distribución, dada por $F(x) = P(X \leq x)$.

Propiedad 2.6 La función de distribución de una variable aleatoria continua X tiene las siguientes propiedades:

- 1) $F(x)$ es continua
- 2) $F'(x)$ existe excepto en a lo más un conjunto finito de puntos.
- 3) $F'(x)$ es continua, al menos por pedazos.

Sin entrar demasiado en aspectos matemáticos de estas propiedades, podemos observar que éstas implican que el rango de X consiste de uno o más intervalos, que la función de distribución es una curva suave en cada intervalo, y que la derivada puede ser integrada en todos los intervalos de interés.

Definición 2.11 La función de densidad de probabilidades, denotada por $f(x)$, es la derivada de la función de distribución.

$$f(x) = \frac{d}{dx} F(x)$$

El término densidad se refiere a la acumula-

ción de probabilidad a lo largo del rango de f . La figura 2.3 muestra una densidad f que se acumula alrededor del punto μ , en el rango. Nótese, sin embargo, que la función de probabilidad f evaluada en un punto x_0 no es la probabilidad de que X asuma el valor x_0 .

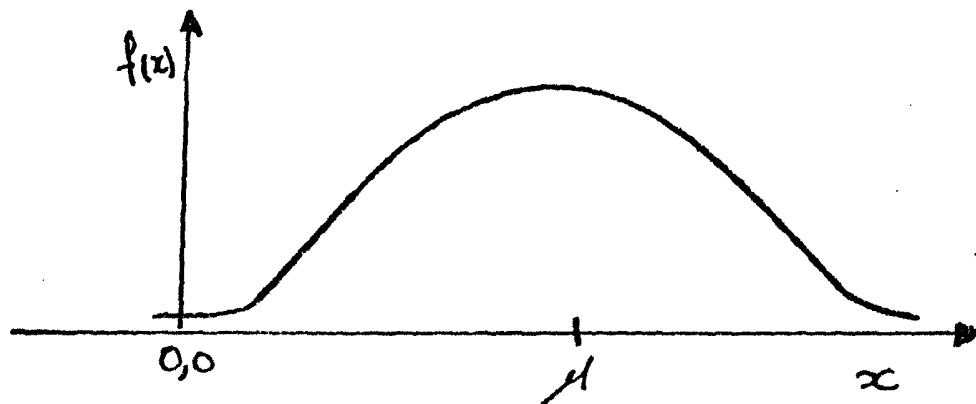


Figura 2.3 Una función de densidad f .

Propiedad 2.7 La función de densidad f de una variable aleatoria X cumple:

1.) $f(x) = 0$ si x no está en el rango de X

2.) $f(x) \geq 0 \quad \forall x$

3.) $\int_{-\infty}^{\infty} f(x) dx = 1$

4.) $F(x_1) = \int_{-\infty}^{x_1} f(x) dx$

En la figura 2.4 se ilustran estas propiedades al mismo tiempo se resalta la relación entre las funciones $f(x)$ y $F(x)$.

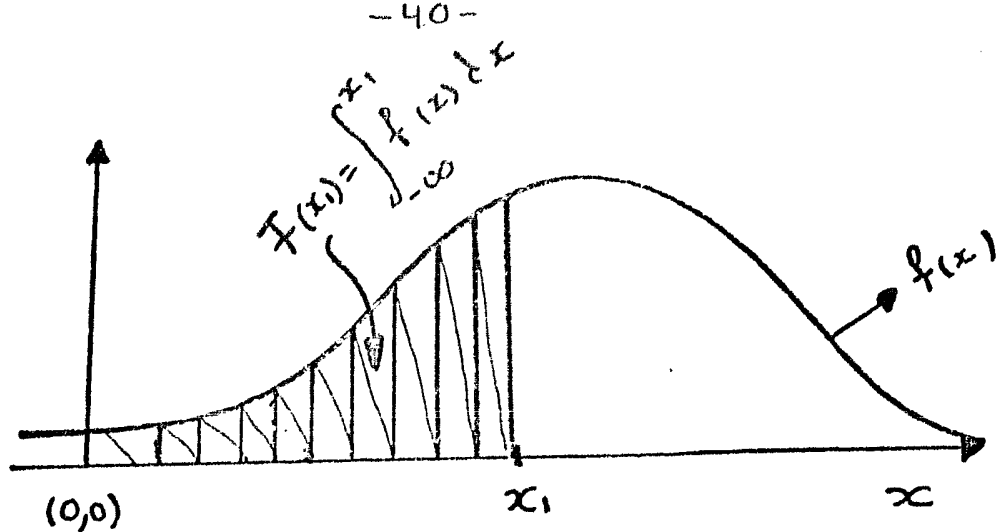


Figura 2.4 $f(x)$ y $F(x)$.

Ejemplo 2.19 Supóngase que el tiempo de vida medido en meses, del componente eléctrico referido en el ejemplo 2.18 se le halló la función de densidad de probabilidad siguiente:

$$f(x) = \begin{cases} e^{-x} & 0 \leq x < \infty \\ 0 & \text{en otro caso} \end{cases}$$

Esta función puede ser vista en la figura 2.5. La función de distribución correspondiente $F(x)$ aparece en la figura 2.6

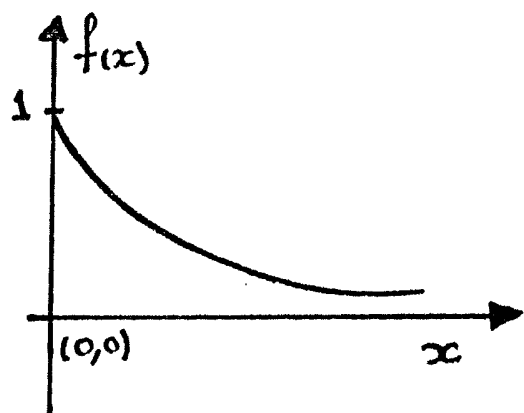


Figura 2.5 $f(x) = e^{-x}$

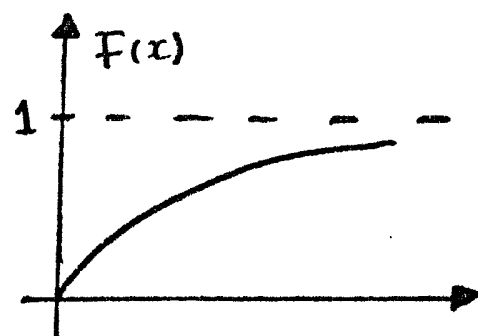


Figura 2.6 $F(x) = \int_{-\infty}^x f(y) dy$

Podemos calcular la probabilidad de que la componente falle en los 10 primeros meses como sigue:

$$P(x \leq 10) = F(10) = \int_{-\infty}^{10} f(x) dx = \int_{-\infty}^0 0 dx + \int_0^{10} e^{-x} dx = -e^{-x} \Big|_0^{10} = -e^{-10} + 1 = .99$$

Lo anterior corresponde a encontrar el área bajo la curva de $f(x)$ entre 0 y 10 (Ver figura 2.5).

Justamente como con las distribuciones discretas, una distribución continua puede ser resumida usando la esperanza (media) y la varianza.

Definición 2.12 Sea X variable aleatoria continua, f su densidad. La esperanza de X , denotada por $E(X)$ es:

$$E(X) = \int_{-\infty}^{\infty} x f(x) dx$$

Ejemplo 2.20 Considérese el experimento ya descrito en los ejemplos 2.16 y 2.19. La esperanza de vida del componente es:

$$E(X) = \int_{-\infty}^{\infty} x f(x) dx = \int_0^{\infty} x e^{-x} dx = -x e^{-x} \Big|_0^{\infty} + \int_0^{\infty} (-e^{-x}) dx = e^{-x} \Big|_0^{\infty} = 1$$

Definición 2.13 Sea X variable aleatoria continua, f su densidad. La varianza de X , denotada por $V(X)$ es:

$$V(X) = \int_{-\infty}^{\infty} (x - E(X))^2 f(x) dx = \int_{-\infty}^{\infty} x^2 f(x) dx - (E(X))^2$$

La desviación estándar $S(X)$ es dada por

$$S(X) = [V(X)]^{1/2}$$

Ejemplo 2.21 Considérese el experimento descrito en los ejemplos 2.16 - 2.20. La varianza en el tiempo de vida del componente eléctrico puede ser calculada por

$$\begin{aligned} V(X) &= \int_{-\infty}^{\infty} (x - E(X))^2 f(x) dx = \int_{-\infty}^{\infty} x^2 f(x) dx - (E(X))^2 = \int_0^{\infty} x^2 e^{-x} dx - 1 \\ &= [-x^2 e^{-x} + 2x e^{-x} - 2e^{-x}] \Big|_0^{\infty} - 1 = 2 - 1 = 1. \end{aligned}$$

La desviación estándar es: $S(X) = V(X)^{1/2} = 1^{1/2} = 1$.

Parece imposible obtener una distribución continua en un experimento del mundo real, ya que todos los dispositivos de medición producen valores discretos, sin embargo, cuando una variable aleatoria discreta es medida en suficientes puntos, ésta empieza a tomar la apariencia de una variable aleatoria continua, por lo tanto es conveniente aproximar la distribución discreta por una continua. En esta forma podemos usar las herramientas del cálculo para medir probabilidades, medias y varianzas en vez del tedioso trabajo de calcular sumatorias.

2.6 Funciones de una variable aleatoria, Momentos, Función Generatriz.

En muchas aplicaciones, una vez que se lleva a cabo el experimento y la distribución de la variable aleatoria está determinada, alguna cantidad relativa a esta variable, requiere de examen. En tales casos, conviene definir una nueva variable como función de la variable aleatoria original.

Ejemplo 2.22 Sea un experimento que tiene como objeto el estudio de algún líquido, siendo su temperatura la variable de interés. X mide tal temperatura en grados centígrados. Si la distribución de X es conocida. Nos interesa la distribución de la temperatura en grados Fahrenheit. En vez de repetir el experimento (quizá muy costoso), podemos simplemente modificar las observaciones originales con la relación

$$Y = 9/5 X + 32 .$$

La variable Y como función de la variable X .

Este ejemplo ilustra una de las relaciones más simples entre variables aleatorias, una relación lineal.

Definición 2.4 Sea X una variable aleatoria correspondiente a los resultados de algún experimento aleatorio. Si la variable aleatoria Y es asociada con X por la relación $Y = aX + b$, a, b constantes arbitrarias. Y se dice en relación lineal con X .

Si la distribución de X es conocida, es simple determinar la distribución de Y . Cada observación X se multiplica por " a " y se le suma " b ". Las probabilidades correspondientes son calculadas de la manera siguiente $P(Y \leq y) = P(aX + b \leq y) = P(aX \leq y - b) = P(X \leq \frac{y - b}{a})$

Un cálculo simple también es suficiente en el cálculo de esperanzas y varianzas. Por ejemplo, si X es una variable aleatoria continua y Y está en relación lineal con X . Entonces

$$\begin{aligned} E(Y) &= E(aX + b) = \int_{-\infty}^{\infty} (aX + b) f(x) dx = a \int_{-\infty}^{\infty} x f(x) dx + b \int_{-\infty}^{\infty} f(x) dx \\ &= a E(X) + b . \end{aligned}$$

Similarmente

$$\begin{aligned} V(Y) &= V(aX + b) = \int_{-\infty}^{\infty} (ax + b)^2 f(x) dx - (E(ax + b))^2 = \\ &= \int_{-\infty}^{\infty} a^2 x^2 f(x) dx + 2ab \int_{-\infty}^{\infty} x f(x) dx + b^2 \int_{-\infty}^{\infty} f(x) dx - (E(ax + b))^2 \\ &= \int_{-\infty}^{\infty} a^2 x^2 f(x) dx + 2ab E(X) + b^2 - (a E(X) + b)^2 = a^2 \left[\int_{-\infty}^{\infty} x^2 f(x) dx - (E(X))^2 \right] \\ &= a^2 V(X) . \end{aligned}$$

Estos resultados podrían causar sorpresa, pero son en realidad muy naturales. Por ejemplo, la suma de una constante "b" no afecta a la varianza. Esto puede intuirse fácilmente; la suma de una constante "g" para todas las observaciones no afecta su posición relativa con respecto a la media. Estos resultados también funcionan en variables discretas con la substitución de la sumatoria (Σ) en vez de la integral (\int) y la función de probabilidad (P) en vez de la función de densidad (f).

Ejemplo 2.23 Si para el experimento del ejemplo 2.22, la variable aleatoria tiene una cierta distribución con media 30 grados y varianza 64 grados. Entonces

$$E(Y) = 9/5 E(X) + 32 = 9/5 (30) + 32 = 86 \text{ grados}$$

$$V(Y) = (9/5)^2 V(X) = (3.24)(64) = 207.3 \text{ grados}$$

$$S(Y) = V(Y)^{1/2} = 14.4 \text{ grados}$$

Usando las mismas técnicas ilustradas para relaciones lineales, podemos calcular la media de cualquier función arbitraria de una variable aleatoria.

Si $Y = g(X)$, entonces:

$$E(Y) = E(g(x)) = \sum_i g(x_i) P(x_i) \text{ en caso discreto}$$

$$\text{ó } E(Y) = E(g(x)) = \int_{-\infty}^{\infty} g(x) f(x) dx \text{ en caso continuo}$$

$$\text{y } V(Y) = V(g(x)) = \sum_i (g(x_i) - E(g(x_i)))^2 P(x_i) \text{ en caso discreto}$$

$$\text{ó } V(Y) = V(g(x)) = \int_{-\infty}^{\infty} (g(x) - E(g(x)))^2 f(x) dx \text{ en continuo}$$

Una función simple, cuya esperanza es de interés, es la función potencia.

$$g(x) = X^r, \quad r=1,2,\dots$$

Definición La esperanza de la función potencia

$$g(x) = X^r, \quad r=1,2,\dots$$

es llamado el r-ésimo momento de la variable X con respecto a cero y es denotado por μ_r .

Aplicando la definición de esperanza, podemos calcular el r-ésimo momento de la variable X con respecto a cero

$$\mu_r = \sum_i (x_i)^r P(x_i) \quad \text{si } X \text{ es discreta y}$$

$$\mu_r = \int_{-\infty}^{\infty} x^r f(x) dx \quad \text{si } X \text{ es continua.}$$

También podemos considerar momentos con respecto a otro punto que no sea el origen, digamos alguna constante "c". La fórmula para momentos con respecto a la constante "c" es:

$$\mu'_r = \sum_i (x_i - c)^r P(x_i) \quad X \text{ discreta}$$

$$\mu'_r = \int_{-\infty}^{\infty} (x - c)^r f(x) dx \quad X \text{ continua}$$

Estas cantidades son llamadas momentos por su analogía con los momentos de inercia en física. Algunos de estos momentos tienen significancia al resumir varias características de la distribución.

Primer momento. El primer momento ($r=1$) con respecto al cero nos da la esperanza o media de la distribución. Es denotada por la letra griega μ y sirve como una medida de la tendencia central o localización de la distribución.

Segundo momento. Este momento ($r=2$) con respecto a la media, da la varianza de la distribución. Es denotada por σ^2 y mide la dispersión de la distribución. La raíz cuadrada de la varianza es llamada la desviación estándar.

Tercer momento. El tercer momento con respecto a la media es una medida de la simetría de la distribución. Si es positivo, la distribución es asimétrica y sesgada positiva (el peso de la distribución está a la izquierda de la media). Si es negativo, ocurre lo contrario y si es cero es simétrica. Estos tres casos son mostrados en la figura 2.7

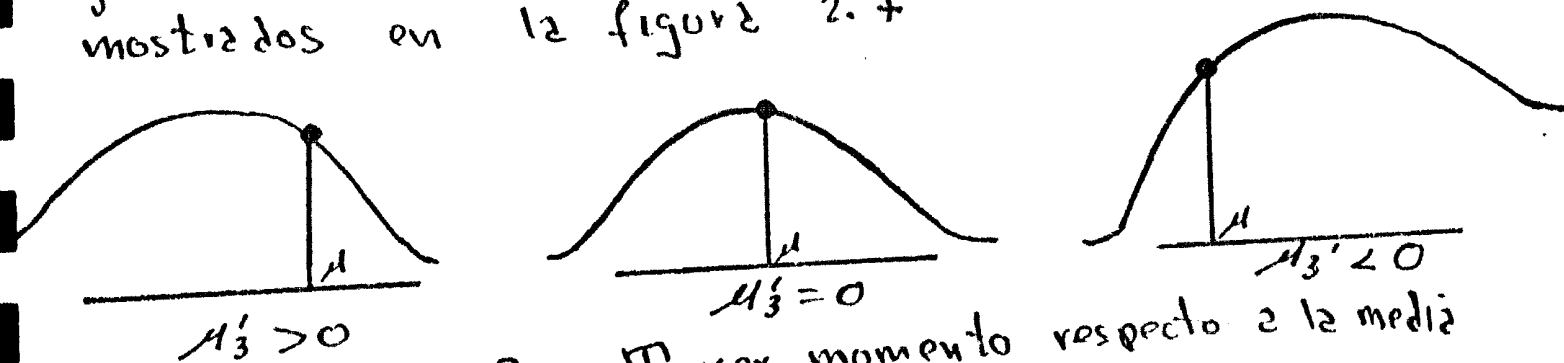


Figura 2.7

Tercer momento respecto a la media

Otra función de una variable aleatoria X la cual es de interés es la función

$$g(x) = e^{x\theta} \text{ con } \theta \text{ variable}$$

Definición 2.16 La función $M(\theta) = E[e^{x\theta}]$, donde E es el operador esperanza, es llamada

(si existe) la función generatriz de momentos de la variable X . Usando el concepto de esperanza, desarrollado antes, podemos calcular la función generatriz de momentos por:

$$M(\theta) = E[e^{x\theta}] = \sum_i e^{x_i\theta} p(x_i), \quad X \text{ discreta}$$

$$\text{ó } M(\theta) = E[e^{x\theta}] = \int_{-\infty}^{\infty} e^{x\theta} f(x) dx, \quad X \text{ continua}$$

Ejemplo 2.24 Si X es una variable aleatoria continua con función de densidad: $f(x) = e^{-x}, 0 \leq x < \infty$
 $= 0$ en otro caso

Entonces la función generatriz de momentos es:

$$M(\theta) = E(e^{x\theta}) = \int_0^{\infty} e^{x\theta} \cdot e^{-x} dx = \frac{1}{1-\theta} \quad \theta \neq 1$$

La función M es llamada función generatriz de momentos por que todos los momentos de X pueden ser calculados a partir de ésta.

En particular, $e^{x\theta}$ puede ser expandida en una serie infinita como:

$$e^{x\theta} = 1 + x\theta + \frac{x^2\theta^2}{2!} + \dots + \frac{x^n\theta^n}{n!} + \dots$$

Aplicando el operador E (lineal) tenemos

$$M(\theta) = 1 + E(x)\theta + E(x^2)\frac{\theta^2}{2!} + \dots + E(x^n)\frac{\theta^n}{n!} + \dots =$$

$$= 1 + \mu_1\theta + \mu_2\frac{\theta^2}{2!} + \dots + \mu_n\frac{\theta^n}{n!} + \dots$$

Entonces los momentos de X pueden ser

calculados por expandir $M(\theta)$ en serie infinita y fijarse en los coeficientes de las potencias de θ .

Ejemplo 2.25 Considérese la distribución discutida en el ejemplo 2.24 $M(\theta) = \frac{1}{1-\theta} = 1 + \theta + \theta^2 + \theta^3 + \dots + \theta^n + \dots$

$$= 1 + (1) \theta + (2!) \frac{\theta^2}{2!} + \dots + n! \frac{\theta^n}{n!} + \dots$$

Entonces $\mu = \mu_1 = 1, \mu_2 = 2, \dots, \mu_n = n!$.

2.7 Distribuciones Conjuntas

Todo lo desarrollado hasta el momento, supone que una variable aleatoria X se define sobre un espacio de eventos de algún experimento aleatorio. En la simulación de sistemas del mundo real a veces es conveniente o quizá necesario definir más de una variable aleatoria sobre el mismo espacio de eventos.

Debemos entonces considerar la distribución conjunta de las variables aleatorias. En esta sección consideraremos el caso especial de dos variables. Los resultados son de fácil extensión al caso en el cual hay dos o más variables definidas en el mismo espacio.

Si X y Y son dos variables aleatorias definidas en el mismo espacio de eventos entonces los resultados del experimento son parejas ordenadas (x, y) en un espacio de dos dimensiones. Los eventos serán combinación de estos resultados bi-dimensionales.

Ejemplo 2.26. Considérese las variables X y Y

definidas sobre el mismo espacio de eventos. Si X asume los valores $0, 1, 2, 3, 4$ y Y los valores 0 y 1 . Entonces las variables asumen conjuntamente los valores

$$\Omega = \{ (0,0), (0,1), (1,0), (1,1), (2,0), (2,1), (3,0), (3,1), (4,0), (4,1) \}$$

En el caso de variables aleatorias discretas, una función de probabilidad P_{XY} asignará probabilidad a cada evento.

Definición 2.17 Para cualesquiera dos variables aleatorias discretas X, Y . La función de densidad de probabilidades conjunta $P_{XY}(x, y)$ es definida como:

$$P_{XY}(x, y) = P(X=x \text{ y } Y=y).$$

Esto es, P_{XY} nos da la probabilidad de que la variable aleatoria X tome el valor x al mismo tiempo que Y toma el valor y . La función de distribución para un par de variables aleatorias puede ser definido de manera análoga.

Definición 2.18 X, Y variables definidas sobre el mismo espacio de eventos. La función de distribución conjunta $F_{XY}(x, y)$ se define como

$$F_{XY}(x, y) = P(X \leq x \text{ y } Y \leq y).$$

Notese que esta función se define para todos los valores de x, y . Además nos servirá tal definición, tanto para variables aleatorias discretas como continuas.

Recuérdese que la probabilidad de que una variable continua asuma un valor particular es 0 . Esta

observación se acerca a las distribuciones conjuntas. Si X, Y son variables aleatorias continuas definidas sobre el mismo espacio de eventos, entonces $P\{X=x, Y=y\}=0$ para cualquier x, y . Esto nos lleva a la siguiente.

Definición 2.19 Sean X, Y dos variables aleatorias continuas, definidas sobre el mismo espacio de eventos. La función de densidad conjunta $f_{XY}(x, y)$ se define como

$$f_{XY}(x, y) = \frac{\partial^2 F_{XY}(x, y)}{\partial x \partial y} \quad \text{si la derivada existe.}$$

$$\text{Así} \quad F_{XY}(x, y) = \int_{-\infty}^x \int_{-\infty}^y f(u, v) du dv$$

Al considerar distribuciones conjuntas de dos variables aleatorias podríamos estar interesados en la distribución de una de las variables. Este tipo de distribuciones se conoce como distribución marginal.

Definición 2.20 Sean X, Y variables aleatorias continuas, definidas en el mismo espacio de eventos. Sea f_{XY} la densidad conjunta. Las marginales de X, Y son:

$$f_X(x) = \int_{-\infty}^{\infty} f(x, y) dy, \quad f_Y(y) = \int_{-\infty}^{\infty} f(x, y) dx$$

Fórmulas similares (con \leq) se usan para discretas.

Definición 2.21 Sean X, Y dos variables aleatorias definidas en el mismo espacio. X, Y se dicen independientes si

$$F_{XY}(x, y) = F_X(x) \cdot F_Y(y)$$

Las funciones de densidad marginales para variables aleatorias continuas (funciones de probabilidad para discretas) permiten cálculo de media y varianza de las variables aleatorias.

Definición 2.22 Sean X, Y variables aleatorias continuas definidas sobre el mismo espacio de eventos. Entonces

$$E(x) = \mu_x = \int_{-\infty}^{\infty} x f_X(x) dx, \quad V(x) = \int_{-\infty}^{\infty} (x - \mu_x)^2 f_X(x) dx$$

Fórmulas similares son usadas para Y , y para el caso en que X, Y son discretas (con Σ). Cuando se trabaja con variables conjuntas otra cantidad de interés es la covarianza (asociación entre X, Y).

Definición 2.23 Sean X, Y dos variables aleatorias continuas con función de densidad conjunta $f_{XY}(x, y)$. La covarianza de X, Y denotada por δ_{XY} es:

$$\delta_{XY} = E[(X - \mu_x)(Y - \mu_y)] = \iint_{-\infty}^{\infty} (x - \mu_x)(y - \mu_y) f_{XY}(x, y) dx dy$$

Existe una definición análoga para el caso discreto.

La correlación entre X y Y se define como:

$$\rho_{XY} = \delta_{XY} / \sigma_x \sigma_y$$

Un valor de ρ_{XY} igual a cero indica que no existe correlación entre X, Y mientras un valor de ρ_{XY} de ± 1

implica una correlación lineal perfecta.

De particular importancia en análisis estadístico es la media y varianza de la suma de variables conjuntas.

Definición 2.24 Sean X y Y variables aleatorias conjuntas y defínase $U = a_1 X + b_1 Y$. Entonces

$$E(U) = a_1 \mu_x + b_1 \mu_y ; V(U) = a_1^2 \sigma_x^2 + b_1^2 \sigma_y^2 + 2 a_1 b_1 \rho_{xy}$$

Nótese que variables aleatorias independientes, no están correlacionadas. El inverso de lo anterior no es siempre cierto.

En esta sección se examinaron muy brevemente las variables conjuntas. Para un estudio más profundo puede consultarse [2.1].

2.8 Algunas distribuciones Comunes

Un buen número de distribuciones comunes surgen del estudio de sistemas del mundo real. No obstante en algunos casos los resultados de experimentos no siguen precisamente estas distribuciones, sin embargo podrían usarse dichas distribuciones para aproximar la distribución real. En esta sección exploraremos algunos detalles (aplicación, medias, varianzas etc) de ciertas distribuciones.

2.8.1 Distribución Bernoulli

La distribución Bernoulli, una distribución discreta, es una de las más simples y se usa en experimentos con sólo dos posibles resultados. Si una variable X

se define y toma el valor "0" con probabilidad "p" y el valor "1" con probabilidad "q = 1-p". Entonces la distribución de X es:

x	$P(x)$
0	p
1	q

La función de distribución de X es:

$$F_X(x) = \begin{cases} 0 & , x < 0 \\ p & , x \in (0, 1) \\ q & , x \geq 1 \end{cases}$$

La media, varianza y desviación estándar son:

$$\mu = 0 \cdot p + 1 \cdot q = q; \quad \sigma^2 = E(x^2) - E(x)^2 = 0 \cdot p + 1 \cdot q - q^2 = q(1-q); \quad \sigma = (q(1-q))^{1/2}$$

La figura 2.8 muestra la distribución de Bernoulli

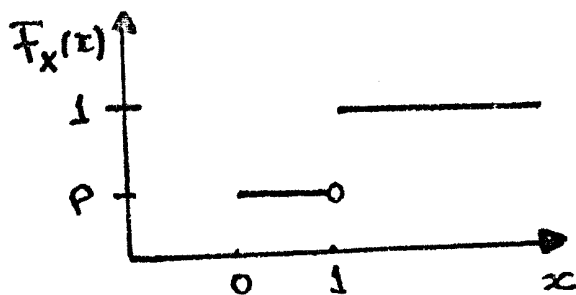
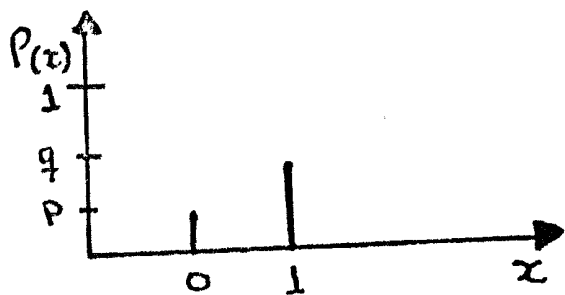


Figura 2.8 Distribución Bernoulli ($q > p$).

Ejemplo 2.27. Considérese el experimento de lanzar una moneda. X toma el valor "0" si cae sol y "1" si es águila. Si $p=q=1/2$ (moneda honesta) entonces:

$$\mu = 1/2, \quad \sigma^2 = 1/4, \quad \sigma = 1/2$$

2.8.2 Distribución Binomial

La Binomial es una distribución discreta que resulta de una secuencia de ensayos independientes de Bernoulli.

Supóngase que un experimento que tiene dos posibles resultados es realizado n veces, $n > 0$. Supóngase además que la probabilidad de uno de los resultados en cualquier ensayo (llamado el resultado "0") es p y que la probabilidad del otro resultado (llamado "1") es $q = 1 - p$. Así, el resultado "0" puede ocurrir desde 0 hasta n veces en los n ensayos, lo mismo para el resultado "1". La probabilidad de que el resultado "0" (a veces llamado fracaso) ocurra precisamente k veces ("1" o éxito ocurre $n - k$ veces) es:

$$P(x=k) = \frac{n!}{k!(n-k)!} p^k q^{n-k} \quad k=0..n$$

La función de distribución es:

$$F_x(k) = \sum_{s=0}^k \frac{n!}{s!(n-s)!} p^s q^{n-s}, \quad k=0..n$$

Esta distribución se ilustra en la figura 2.9.

La esperanza y varianza se calculan con facilidad tratando la binomial como suma de n variables Bernoulli X_1, \dots, X_n . Entonces

$$\mu = E(X_1 + \dots + X_n) = q + \dots + q = nq$$

$$\sigma^2 = V(X_1 + \dots + X_n) = pq + \dots + pq = npq.$$

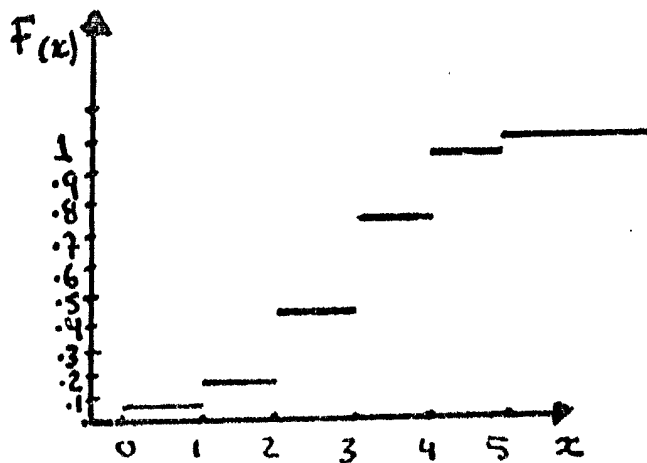
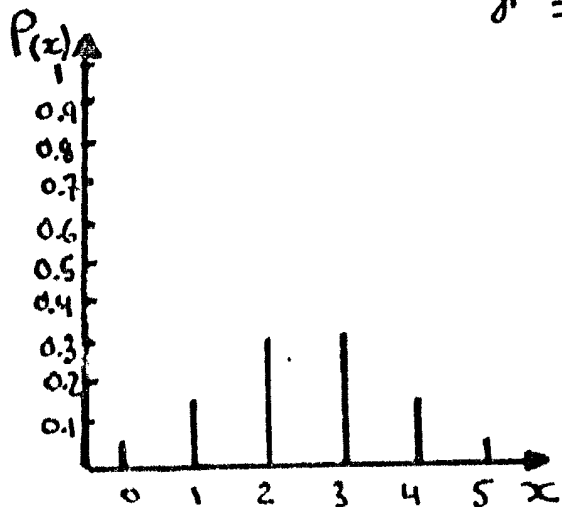


Figura 2.9 Distribución Binomial ($n=5, p=1/2$).

Ejemplo 2.28 Supóngase que 5 dados son lanzados, y deseamos calcular la probabilidad de obtener 0, 1, 2, 3, 4, 5, seises. Si el resultado "0" es "no es seis" y "1" es "seseis", entonces $p = 5/6$ y $q = 1/6$ (dados honestos) y la media y la varianza es dada por:

$$\mu = (5)(1/6) = 5/6, \quad \mu^2 = (5)(1/6)(1/6) = 5/36.$$

2.8.3 Distribución Geométrica

La distribución geométrica es también una distribución discreta que surge como resultado de n ensayos de Bernoulli independientes, justo como aparece la Binomial. Ésta mide el número de ensayos antes del primer éxito o fracaso. Si p es tomada como la probabilidad del fracaso en un ensayo dado y q la probabilidad del éxito, entonces la función de probabilidad es:

$$P_x(k) = P(X=k) = p^{k-1} q, \quad k=0,1,\dots$$

Donde X mide el número de ensayos hasta que ocurre el primer éxito. La función de distribución es

$$F_x(k) = \sum_{i=0}^k P(i) = \sum_{i=1}^k p^{i-1} q.$$

La figura 2.10 muestra la distribución geométrica para $p = 0.7$. La media y varianza para la distribución geométrica son: $\mu = 1/q, \quad \mu^2 = p/q^2.$

Ejemplo 2.29 Si se realiza un experimento en el cual una moneda cargada se lanza hasta que aparezca la primer "águila". Sea $1/5$ la probabilidad de "águila", mientras que es $4/5$ la de "sol". Entonces $\mu = 5, \mu^2 = 20.$

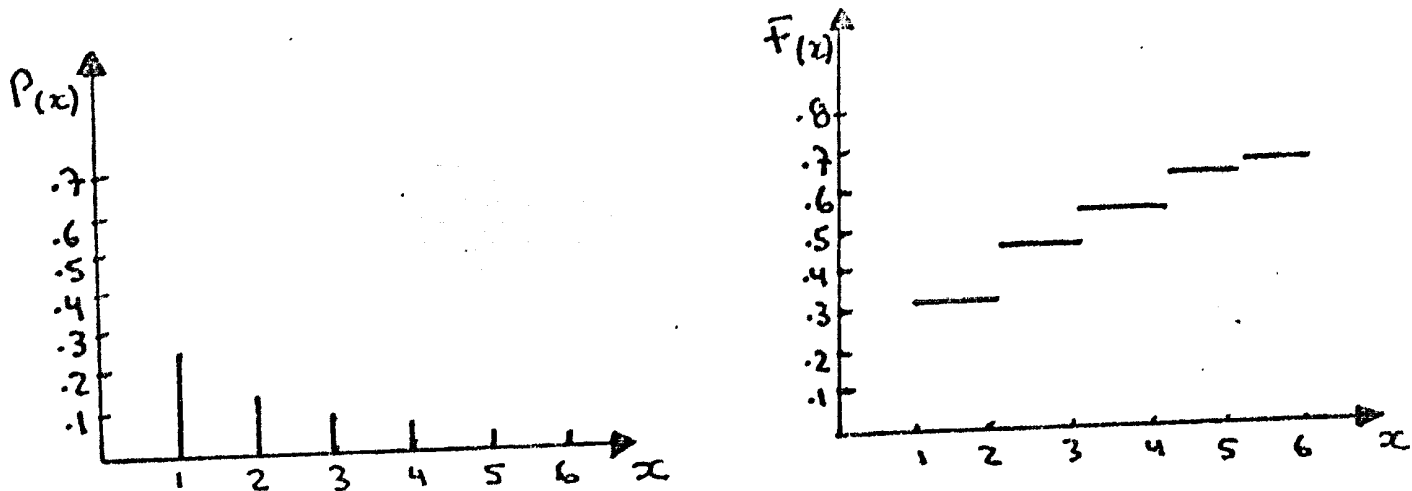


Figura 2.10 Distribución Geométrica ($p = 0.7$)

2.8.4 Distribución Poisson

La distribución de Poisson, otra distribución discreta, ha sido muy útil al modelar el comportamiento de arribos y otros procesos aleatorios parecidos. Tuvo su origen en el modelaje de llamadas telefónicas a un conmutador. La función de probabilidad es dada por:

$$P_x(x) = \frac{\lambda^x e^{-\lambda}}{x!}, \quad x = 0, 1, \dots$$

Donde λ es una constante, llamada parámetro de distribución. La función de distribución es:

$$F_x(x) = \sum_{i=0}^x \frac{\lambda^i e^{-\lambda}}{i!}$$

Su media y varianza son: $\mu = \lambda, \sigma^2 = \lambda$.

La figura 2.11 ilustra la distribución de Poisson para $\lambda = 4.5$

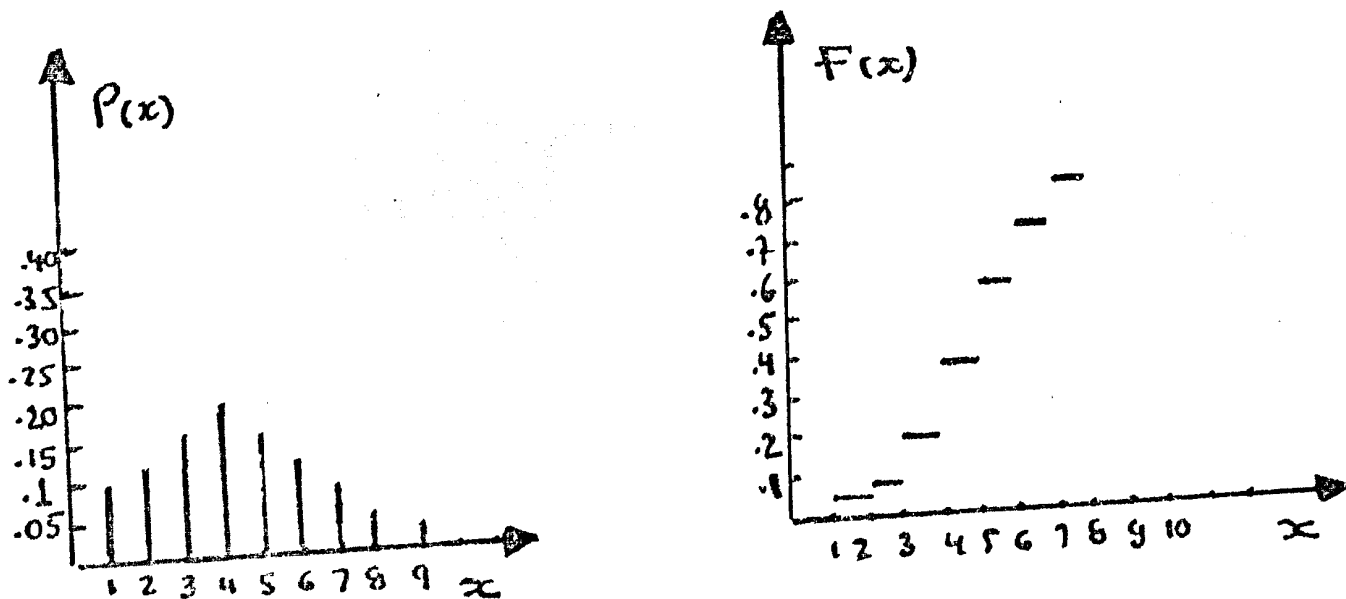


Figura 2.11 Distribución de Poisson con $\lambda = 4.5$

Ejemplo 2.30 Las llamadas a un conmutador se comportan como una distribución de Poisson con cinco llamadas por minuto como promedio. La probabilidad de que lleguen 10 llamadas en el siguiente minuto es:

$$P_{\lambda}(x=10) = \frac{\lambda^{10} e^{-\lambda}}{10!} = \frac{5^{10} e^{-5}}{10!} = 0.0001$$

2.8.5 Distribución Uniforme

Una distribución continua, la uniforme, es una distribución en la cual la densidad es una constante.

$$f(x) = \begin{cases} 1/(b-a) & , a \leq x \leq b \\ 0 & , \text{en otro caso.} \end{cases}$$

Esta distribución es llamada en ocasiones la distribución rectangular. El intervalo es llamado el rango de la distribución. La función de distribución es:

$$F_X(x) = \int_{-\infty}^x f(t) dt = \begin{cases} 0, & x < a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & x > b \end{cases}$$

La media y la varianza son: $\mu = \frac{a+b}{2}$; $\sigma^2 = \frac{(b-a)^2}{12}$

En la figura 2.12 se muestra la distribución uniforme.

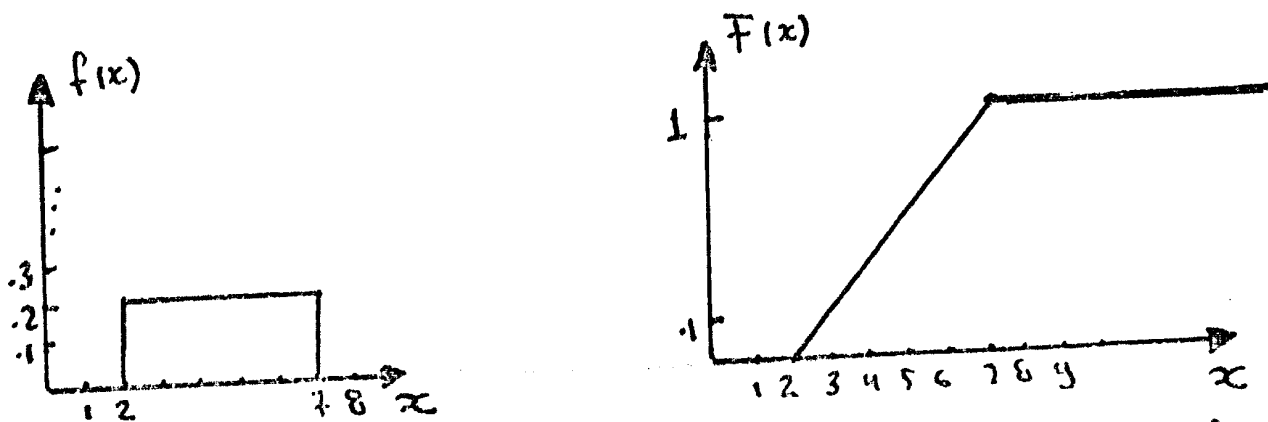


Figura 2.12 Distribución Uniforme ($a=2, b=7$).

Una distribución uniforme particular, es aquella en la cual $a=0$ y $b=1$, llamada la distribución uniforme estándar, ésta es de interés al generar secuencias aleatorias y es tratada en detalle en el capítulo 4.

2.8.6 Distribución Normal

Posiblemente la distribución más útil es la distribución continua conocida como la distribución normal. Esta distribución es usada para modelar muchas situaciones, tales como medir pesos y estaturas, medir errores que se cometen al elaborar componentes, y muchas otras situaciones. La distribución es simétrica (ver sección 2.6), con densidad dada por:

$$f(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{(x-\mu)^2}{2\sigma^2}\right\}, \quad -\infty < x < \infty$$

Esta distribución causa un poco de problemas al calcularse su función de distribución ya que no ha podido ser calculada con exactitud. Su función de distribución está dada por:

$$F(x) = \int_{-\infty}^x \frac{1}{(\sqrt{2\pi}\sigma)^2} \exp\left\{-\frac{(t-\mu)^2}{2\sigma^2}\right\} dt$$

Esta integral no se ha calculado. Para ayudar a resolver el problema, se han elaborado tablas de valores para la distribución normal estándar la cual tiene media 0 y varianza 1, esto es:

$$f(z) = \frac{1}{\sqrt{2\pi}} e^{-z^2/2} \quad -\infty < z < \infty$$

Esta función tiene solución numérica disponible en tablas de probabilidades acumuladas. Las probabilidades acumuladas para distribuciones normales no estándar pueden ser calculadas a partir de las probabilidades acumuladas estándar. Si X es normal con media μ y varianza σ^2 , entonces la variable aleatoria

$$Z = (x - \mu) / \sigma$$

Se distribuye normal con media 0 y varianza 1. La figura 2.13 muestra una normal con $\mu = 5.2$ y $\sigma = 2.2$. La distribución normal es quizá la distribución más usada en el modelaje del mundo real, ya que muchos fenómenos son de comportamiento normal, pero además existen poderosas razones como las

que a continuación citamos por lo que es importante esta distribución.

1.- La suma de n variables aleatorias idénticamente distribuidas e independientes tiende a una distribución normal cuando n crece. Este resultado es conocido como el teorema central de límite.

2.- Si X es binomial, si n crece, X tiende a una normal con $\mu = np$, $\sigma^2 = npq$.

3.- Si X se distribuye Poisson, entonces con λ apropiada, X tiende a una normal con $\mu = \lambda$.

Así la normal es usada no sólo por su propia naturaleza, si no también como aproximación a otras.

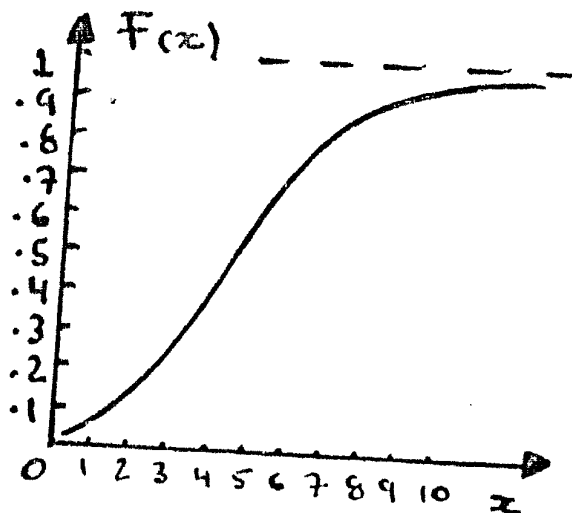
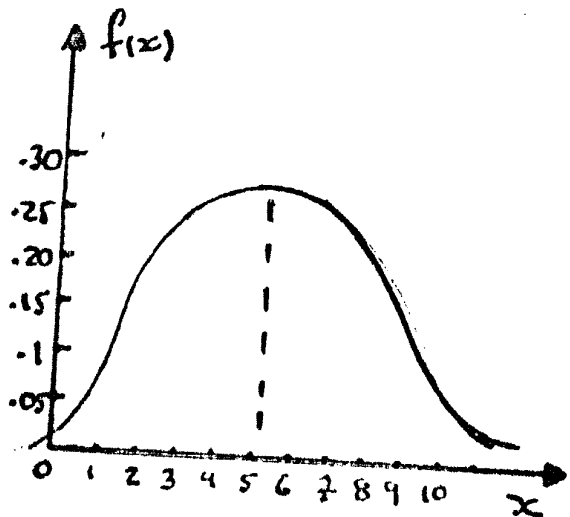


Figura 2.13 Distribución Normal ($\mu = 5.2, \sigma^2 = 2.2$)

2.8.7 Distribución exponencial

Otra distribución continua que ha sido muy usada

es la distribución exponencial. Es usada para modelar tiempos de vida en componentes del mundo real, tales como tiempos de vida de tornillos, focos, etc. también se usa para caracterizar tiempos de servicio y tiempos entre arribos en sistemas de colas. La función de densidad para esta distribución es:

$$f_x(x) = \lambda e^{-\lambda x}, \quad 0 \leq x < \infty.$$

Donde λ es una constante positiva. La función de distribución es:

$$F_x(x) = \lambda \int_0^{\infty} e^{-\lambda x} dx = 1 - e^{-\lambda x}$$

La media y varianza son:

$$\mu = \int_0^{\infty} x e^{-\lambda x} dx = \frac{1}{\lambda}; \quad \sigma^2 = \frac{1}{\lambda^2}$$

Entonces la media y desviación estándar son iguales. La distribución se muestra en la figura 2.14.

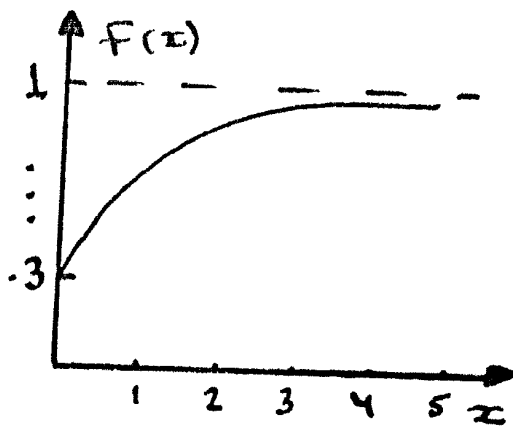
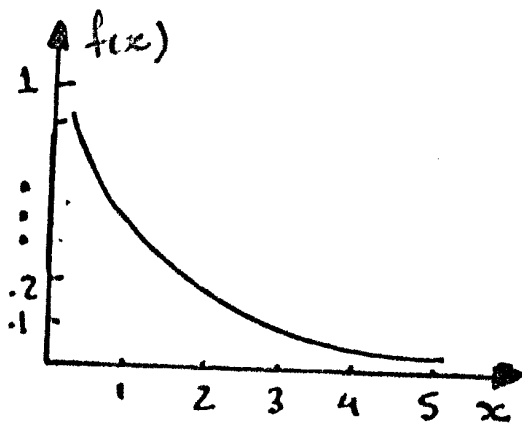


Figura 2.14 Exponencial ($\lambda=1$).

Posiblemente el aspecto más importante de la exponencial, es su propiedad de "pérdida de memoria".

Ésta es expresada matemáticamente en la siguiente

Propiedad 2.8 Supóngase que la variable aleatoria X tiene una distribución exponencial. Entonces cualesquiera $x_1, x_2 > 0$, $P(X \leq x_2 | X > x_1) = P(0 \leq X \leq x_2 - x_1)$.

El inverso también es cierto. Otro aspecto de interés es la relación entre la exponencial y la Poisson.

Propiedad 2.9 Si un proceso de arribos de clientes a un sistema de colas sigue la distribución Poisson, con promedio de arribos λ . Entonces la distribución del tiempo entre arribos es exponencial con parametro $1/\lambda$.

Estas propiedades se usaran en capítulos posteriores al estudiar sistemas de colas.

2.8.8 Distribución Chi-Cuadrada

Una distribución que usaremos en pruebas de bondad de ajuste es la distribución Chi-Cuadrada. La función de densidad es:

$$f(x^2) = \frac{x^{r/2 - 1} e^{-x/2}}{2^{r/2} \Gamma(r/2)}, \quad x^2 > 0$$

En esta fórmula, Γ es la función Gamma estándar (otra densidad). El parametro de la distribución es r , conocido como los grados de libertad. Esta

distribución aparece frecuentemente cuando se trabaja con cuadrados de densidades normales estándar. Si $Z_i, i=1, \dots, k$, son variables aleatorias distribuidas normal estándar y son independientes, entonces $Z_1^2 + \dots + Z_k^2$ es chi-cuadrada con k grados de libertad.

2.8.9 Distribución t-de Student.

La distribución t-de Student también es usada en la prueba de hipótesis. Sean Z y U variables aleatorias independientes, donde Z es normal estándar y U es chi-cuadrada con r grados de libertad. Entonces

$$t = Z / (U/r) \text{ es t-de student.}$$

Esta distribución tiene sólo un parámetro (r) y también se le conoce como los grados de libertad. Esta densidad aparece cuando se muestrea distribuciones normales. Si X denota la media muestral, S la desviación estándar muestral y N el tamaño de muestra. Entonces

$$t = N^{1/2} (X - \mu) / S$$

Sigue una densidad t-de Student con $N-1$ grados.

2.8.10 Distribución F

Igual que las dos últimas, esta distribución también es usada en pruebas de hipótesis.

Sean U y V dos variables aleatorias independientes

ambas distribuidas Chi-cuadrado con V_1 y V_2 grados de libertad respectivamente. Entonces

$$F = (U/V_1) / (V/V_2)$$

Es una distribución F con V_1, V_2 grados de libertad.

Esta distribución aparece cuando se muestrean dos poblaciones normales. Sea la primera muestra de N_1 puntos con media poblacional μ_1 y con varianza poblacional σ_1^2 . Sea la segunda muestra de N_2 puntos con media poblacional μ_2 y con varianza poblacional σ_2^2 . Sean las varianzas muestrales respectivas S_1^2 y S_2^2 . Entonces.

$$F = (S_1^2 / \mu_1^2) / (S_2^2 / \mu_2^2)$$

Se distribuye F con N_1-1, N_2-1 grados de libertad.

Referencias

- 2.1) Feller W. An Introduction to Probability theory and its applications. Vol. 1:
John Wiley and Sons, Inc.

CAPITULO 3

Introducción

En ocasiones una variable aleatoria X que se usa para representar algún aspecto en un modelo de simulación, corresponde con una distribución conocida. Si éste es el caso, la tarea de investigación se simplifica muchísimo. Sin embargo, podría ser que éste no fuera el caso, entonces con el fin de caracterizar dicha variable aleatoria, se toma un conjunto de valores muestrales a través de observaciones y se intenta en base a esto caracterizar la conducta de la variable aleatoria. Existen dos maneras muy usadas para lograr lo anterior. La primera es mediante la construcción de una distribución empírica usando mínimos cuadrados o alguna otra técnica de ajuste de curvas. Esto se usa cuando la variable no parece seguir alguna distribución común. La segunda forma propuesta consiste en establecer la hipótesis de que la variable aleatoria sigue cierta distribución y usar métodos estadísticos para probar la validez de la hipótesis. Esta última forma de atacar el problema es la de mayor uso de entre las dos citadas, y en caso de tener éxito, produce una función de distribución que se puede expresar analíticamente y cuya conducta es en muchos casos conocida. La distribución normal es una densidad bien conocida y el investigador podría tener la fortuna de encontrarse que la variable aleatoria de interés sigue una conducta normal, y así disponer de información acerca de esta distribución.

Una vez que a la variable aleatoria se le caracteriza como una distribución dada, la siguiente tarea es describir la distribución en términos que la resuman. Esto incluye la estimación de los parámetros (medidas de caracterización) de la distribución. Cuando trabajamos con la distribución normal, por ejemplo, el interés es estimar su media y su varianza. La distribución *t*-de Student por otra parte es caracterizada por sus grados de libertad.

El propósito de este capítulo es la revisión de los conceptos básicos de inferencia estadística y estimación. Cuatro de las más comunes pruebas estadísticas se incluyen. Estas pruebas se usarán en los capítulos siguientes.

Como complemento puede verse [3.1]

3.1 Estimación

Si una variable aleatoria de interés en un estudio de simulación se representa por una distribución empírica o se sabe que sigue una distribución particular, se presenta el problema de estimar los parámetros apropiados de la distribución. Sería casi nulo saber que una variable es normal sin saber qué valor tienen la media y varianza.

Existe un buen número de propiedades deseables en cualquier estimador de un parámetro. Algunas

de estas son las siguientes.

1. Insesgado. Un estimador $\hat{\theta}$ de algún parámetro θ se dice insesgado si $E(\hat{\theta}) = \theta$. Por ejemplo la media muestral

$$\bar{X} = n^{-1} \sum_{i=1}^n x_i$$

es insesgado para la media poblacional μ .

Por otro lado, la desviación estándar muestral

$$S = \left(\sum_{i=1}^n (x_i - \bar{X})^2 (n-1) \right)^{1/2}$$

no es insesgado para la desviación σ .

2. Varianza mínima. Un estimador $\hat{\theta}$ de un parámetro poblacional θ se dice de varianza mínima si

$$V_{\hat{\theta}}^2 \leq V_{\hat{\theta}^*}^2, \quad \forall \text{ estimador } \hat{\theta}^*$$

3. Suficiencia. Un estimador $\hat{\theta}$ de un parámetro θ es suficiente si usa toda la información muestral. Por ejemplo \bar{X} y S del punto 1 son suficientes.

4. Consistencia. Un estimador $\hat{\theta}$ se le conoce como consistente si el estimador se acerca al valor θ cuando el tamaño de la muestra crece.

No siempre es posible obtener un estimador que tenga todas estas cualidades. Por ejemplo, la desviación estándar muestral se usa generalmente como estimador de la desviación estándar poblacional sin

ser un estimador insesgado.

Un estimador de algún parámetro poblacional provee una indicación del valor verdadero del parámetro. Esta indicación puede ser o no significativa, ya que ninguna indicación del grado de seguridad del estimador es dada por éste.

Por lo anterior la estimación de parámetros poblacionales generalmente se da en forma de intervalos de confianza. Este intervalo se da en la forma

$$\bar{\theta} \pm \epsilon, \quad \epsilon > 0$$

El valor de ϵ está relacionado con el tamaño de muestra y la desviación estándar de $\bar{\theta}$.

Un intervalo de confianza generalmente lleva algún grado de certidumbre, de que el parámetro esté en el intervalo. Existen una buena variedad de técnicas para construir intervalos de confianza, la elección de la técnica depende de la distribución a la cual corresponde el estimador. Por esta razón, no damos ejemplos, en vez de ello, las técnicas requeridas se introducen en el momento en que se les necesite.

3.2 Prueba de Hipótesis

Una alternativa a la construcción de distribuciones empíricas es establecer que los puntos muestrales de una variable aleatoria X corresponden a alguna distribución conocida.

Entonces se usan métodos estadísticos para evaluar la validez de la hipótesis. En esta sección revisaremos algunos de los conceptos fundamentales que se encuentran en la construcción y prueba de hipótesis estadísticas. En secciones posteriores examinaremos pruebas particulares y sus aplicaciones en simulación.

Una hipótesis estadística es una suposición acerca de la población que ha sido muestreada. Podría establecerse que la población sigue cierta distribución dada o que un parámetro particular de la población tiene cierto valor. Una prueba de hipótesis es simplemente una regla por medio de la cual una hipótesis ya sea que se acepta o se rechaza. Esta decisión se basa generalmente en estadísticas obtenidas de explorar una muestra o un conjunto de muestras de la población. Estas estadísticas muestrales son conocidas como estadísticas de prueba. La región crítica de una estadística de prueba consiste de los valores de la estadística de prueba que indican el rechazo de la hipótesis.

Ejemplo 3.1 Se lleva a cabo una simulación para examinar la operación de un centro de servicio. Se estima que los clientes arriban al centro con una tasa promedio de 12 por hora. Para probar esta suposición, la hipótesis se establece como: $H_0: \mu = 12$.

El enunciado H_0 (llamada la hipótesis nula) implica otra hipótesis (hipótesis alternativa), la cual es verdad si se determina que H_0 es falsa. Para este caso la hipótesis alternativa se puede establecer como

$$A: \mu \neq 12.$$

Supongamos que el encargado de la simulación decide rechazar H_0 si una muestra produce un promedio de arribos de menos de 10 o más de 14 por hora. La estadística de prueba es entonces la media muestral \bar{X} , en la cual la región crítica es dada por $|\bar{X} - 12| > 2$

Una prueba de hipótesis se basa en estadísticas obtenidas a partir de una muestra. Cualquier cosa basada en estadísticas muestrales envuelve algún grado de probabilidad de cometer un error de decisión. La muestra recolectada podría no ser representativa de la población y si basamos una decisión en esta muestra podría resultar errónea. Existen dos tipos de errores que se pueden cometer. Si la estadística muestral nos dirige a rechazar la hipótesis nula que es cierta, se está cometiendo un error tipo I. La probabilidad de cometer este error se denota por " α " y se le llama el nivel de significancia. Si la estadística de prueba no rechaza una H_0 falsa se comete un error tipo II. La probabilidad de cometer este tipo de error se denota por " β ". La situación de la prueba y la posible decisión se resume en la figura 3.1.

Resultado de La Prueba	H_0 es falsa	H_0 es verdad
H_0 se rechaza	CORRECTO	Error I
H_0 se acepta	Error II	CORRECTO

Figura 3.1 Errores I, II

Obviamente uno de los objetivos en la prueba de hipótesis es minimizar α y β , las probabilidades de una decisión incorrecta. Desafortunadamente si una probabilidad se reduce, la otra aumenta. En efecto, la única manera de reducir simultáneamente ambos riesgos se basa en la decisión sobre una estadística muestral obtenida de una muestra de gran tamaño. Muchas veces α se establece con algún nivel de aceptación pre-determinado y la regla de decisión es formulada para minimizar β .

Los pasos a dar en una prueba de hipótesis pueden resumirse en:

1. Establecer H_0 y A .
2. Fijar α
3. Elegir la estadística de prueba apropiada por medio de la cual se pruebe H_0 .
4. Suponer que H_0 es cierta y determinar la distribución muestral de la estadística de prueba.
5. Indicar región crítica en la cual H_0 será rechazada,
- 6.- Recolectar una muestra, calcular la estadística de prueba y probar la hipótesis.

Ejemplo 3.2 Considérese el experimento descrito en el ejemplo 3.1. Una prueba de la hipótesis siguiendo el procedimiento descrito se haría como sigue:

1. $H_0: \mu = 12$, $A: \mu \neq 12$

2. $\alpha = 0.05$

3. La estadística de prueba es \bar{X} . Esta estadística de prueba puede ser estandarizada como

$$\bar{X}: Z = \frac{\bar{X} - \mu}{\sigma/\sqrt{n}}$$

donde σ es la desviación estándar poblacional y n el tamaño de muestra.

4. \bar{X} es distribuida normal con media μ y desviación estándar σ/\sqrt{n} . Este resultado se sigue del teorema Central del límite.

5. La región crítica sobre Z para un riesgo de $\alpha = 0.05$ es mostrada en la figura 3.2. Entonces H_0 se rechaza si

$$\left| \frac{\bar{X} - \mu}{\sigma/\sqrt{n}} \right| > 1.96$$

6. Supóngase que se toma una muestra de tamaño 16, que $\sigma = 2$, y que la muestra produce $\bar{X} = 14$. Entonces

$$Z = \frac{14 - 12}{1} = 2$$

Así, esta muestra dirige al rechazo de H_0 . La conclusión es entonces que $\mu \neq 12$.

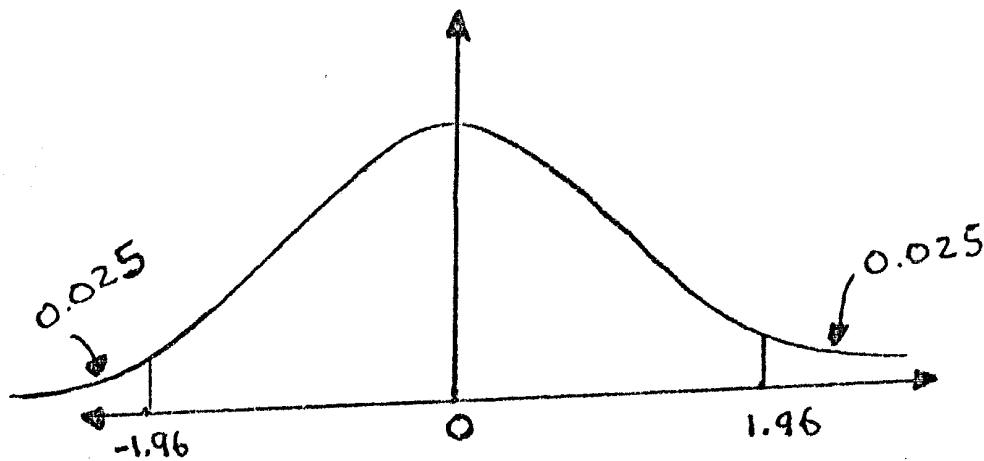


Figura 3.2 Región Crítica

El procedimiento descrito anteriormente no considera β , la probabilidad de error tipo II. Para calcular β , es necesario suponer que H_0 la hipótesis nula es falsa y una hipótesis alternativa específica es verdad. Por ejemplo, si H_0 en el ejemplo 3.1 se supone falsa y la hipótesis alternativa $A: \mu = 14$ se supone verdad, la situación es mostrada en la figura 3.3. El área sombreada es β y puede ser calculada de la función de distribución normal como:

$$\begin{aligned}
 \beta &= P(\bar{X} \leq 13.96) = \\
 &= P(\text{Aceptar } H_0 \text{ dado que es falsa}) \\
 &= P(\bar{X} \leq 12 + 1.96) \\
 &= P\left(Z \leq \frac{13.96 - 14}{1}\right) \\
 &= P(Z \leq -0.04) = .484
 \end{aligned}$$

Para calcular β para varias hipótesis alter-

nativas y representar β contra μ , se usa una curva conocida como la curva característica de operación (C.C.O.). Para los datos del ejemplo 3.1 se muestra la C.C.O. en la figura 3.4.

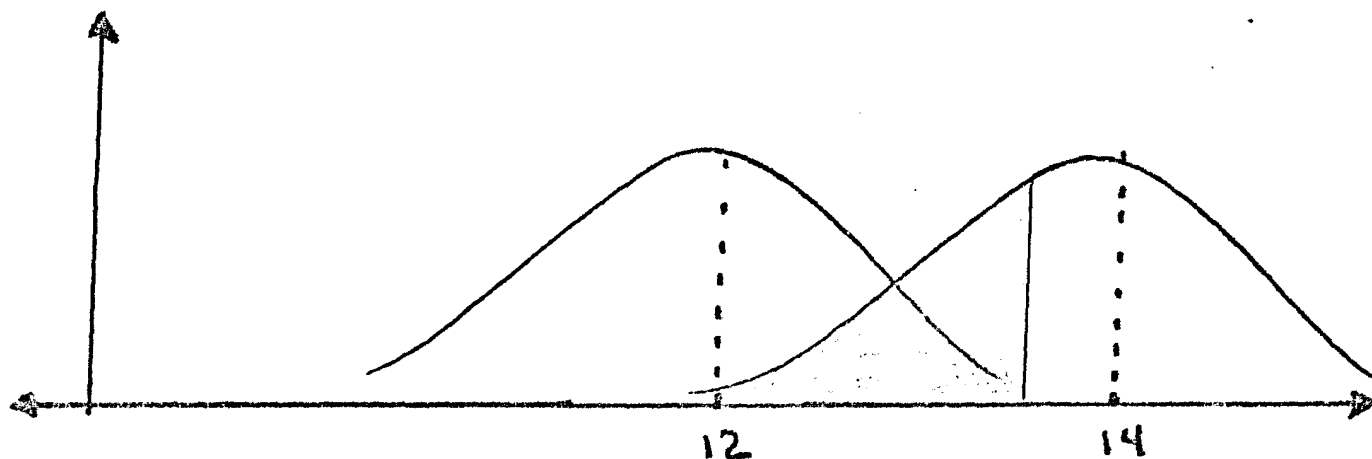


Figura 3.3 valor de β .

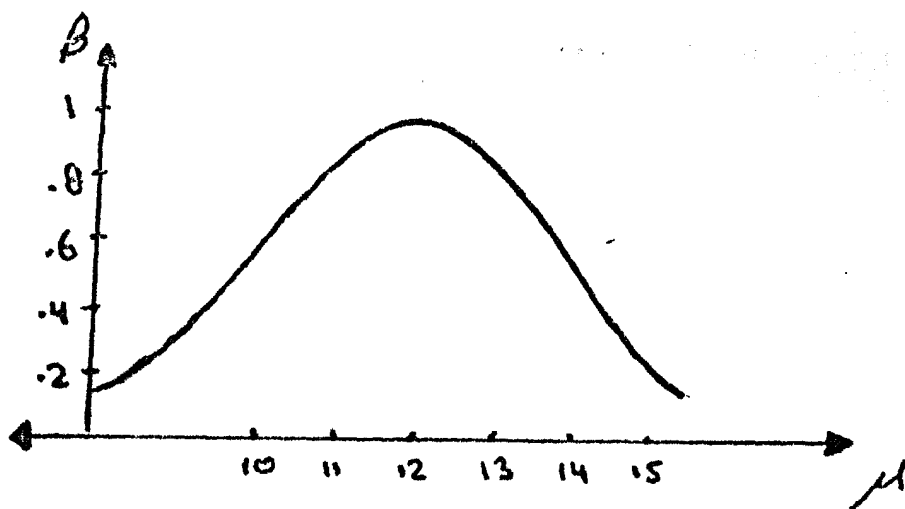


Figura 3.4 Curva Característica de operación (C.C.O.)

Frecuentemente, más que aparear β contra μ , se aparece el complemento $1-\beta$ contra μ . La cantidad $1-\beta$ se conoce como el poder de la prueba, una gráfica de la pareja $1-\beta$ y μ se conoce como curva de poder. La curva de poder para la prueba y datos del ejemplo 3.1 se muestra en la figura 3.5.

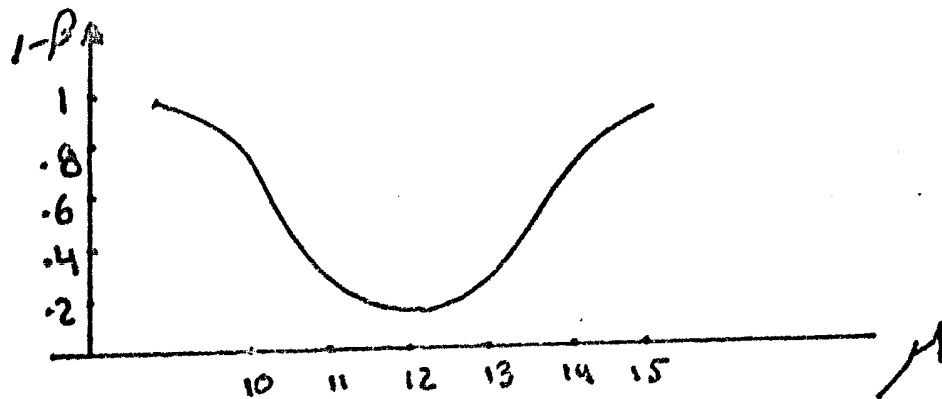


Figura 3.5 Una curva de poder.

Nótese que rechazar una hipótesis nula es una acción "positiva", mientras que aceptar ésta es una acción "nula". Esto puede ser determinado por un examen a la curva característica o a la de poder. En el ejemplo anterior, si la media poblacional fuera 12.5, la hipótesis nula $H_0: 12$ sería aceptada con tanta probabilidad como si la media poblacional fuera 12. Esto nos dice que la prueba no detecta pequeñas diferencias entre lo hipotético y lo real. Sin embargo, si detecta una diferencia un tanto considerable, con probabilidad alta. En este sentido es que se dice que rechazar la hipótesis nula es una acción "positiva". Aceptar la hipótesis nula puede entonces ser visto como una inconclusión. La hipótesis puede ser falsa pero no tenemos evidencia suficiente de esto.

Qué prueba usar depende de lo que se desea probar. En las siguientes cuatro secciones se discuten cuatro de las pruebas más usadas, indicando las suposiciones acerca de la población así como el uso y utilidad de estas pruebas.

3.3 La Prueba t

La prueba t se usa para probar la hipótesis de que dos poblaciones normales tienen media igual. Sean dos poblaciones normales, la primera con media μ_1 y varianza σ_1^2 y la segunda con media μ_2 y varianza σ_2^2 . En general no se conocen las varianzas, estas deben ser estimadas de los datos muestrales. Se toman dos muestras, la primera de tamaño n_1 denotada por x_1, \dots, x_{n_1} de la primera población y la segunda de tamaño n_2 denotada por y_1, \dots, y_{n_2} de la segunda población. Las hipótesis, nula y alternativa son

$$H_0 : \mu_1 = \mu_2 \quad A : \mu_1 \neq \mu_2 \quad \text{ó}$$

$$H_0 : \mu_1 - \mu_2 = 0 \quad A : \mu_1 - \mu_2 \neq 0$$

Si las variables x_i, y_i son independientes, debemos hacer una suposición extra sobre las varianzas desconocidas para tener una prueba exacta. Esta suposición es que las varianzas son iguales. Aunque existen pruebas aproximadas para el caso en que son varianzas diferentes, estas pruebas no se discuten aquí pero pueden disponerse de ellas en [3.1] ó [3.2]. Con la suposición de varianzas iguales, la prueba estadística para probar la igualdad de medias es:

$$t = \frac{(\bar{X} - \bar{Y})}{[s^2 (1/n_1 + 1/n_2)]^{1/2}}$$

Donde \bar{X} es la media muestral de la primer población y \bar{Y} para la segunda y S^2 es un estimado de la varianza dado por:

$$S^2 = \frac{[(n_1-1)S_1^2 + (n_2-1)S_2^2]}{(n_1 + n_2 - 2)}$$

S_1^2 es la varianza muestral para la primer población, y S_2^2 para la segunda. Si la hipótesis nula es verdad ($\mu_1 = \mu_2$), t se distribuye como t -de Student con $n_1 + n_2 - 2$ grados de libertad (vease sección 2.8.9). La región crítica para la prueba es: $|t| \geq t(1-\alpha/2), (n_1 + n_2 - 2)$

Ejemplo 3.3 Supóngase que se toman 10 observaciones de una variable aleatoria X y los diez resultados son $\{1, 2, -1, 3, 7, 8, 9, 4, 3, 2\}$. También se toman observaciones de una variable aleatoria Y , en este caso son 8 y los resultados son $\{0, 3, 6, -2, 4, 0, 7, 8\}$. Si X, Y son normales con igual varianza. Entonces

$$\bar{X} = \frac{\sum_{i=1}^{10} x_i}{10} = 3.8, \quad \bar{Y} = \frac{\sum_{i=1}^8 y_i}{8} = 3.25$$

$$S_1^2 = \frac{\sum_{i=1}^{10} (x_i - \bar{x})^2}{9} = 10.4, \quad S_2^2 = \frac{\sum_{i=1}^8 (y_i - \bar{y})^2}{7} = 13.36$$

$$S^2 = \frac{(9)(10.4) + 7(13.36)}{16} = 11.695$$

Las hipótesis son: $H_0: \mu_1 - \mu_2 = 0, A: \mu_1 - \mu_2 \neq 0$

La prueba estadística es:

$$t = \frac{(\bar{x} - \bar{y})}{(s^2 (1/n_1 + 1/n_2))^{1/2}} = \frac{(3.8 - 3.25)}{[11.695 (1/10 + 1/8)]^{1/2}} = .34$$

Si el nivel de significancia α es 0.05 (Con probabilidad 0.05 se rechaza H_0 que es cierta).

Entonces $t_{0.975, 16} = 2.12$, así $t = 0.34 < 2.12$ por tanto no podemos rechazar H_0 .

Supongamos ahora que no existe la independencia. Si éste es el caso, es necesario hacer otras suposiciones para poder tener una estadística de prueba. Supongamos que las muestras son tomadas en las mismas condiciones, además requerimos tamaños de muestra igual. Bajo estas suposiciones, en vez de considerar una variable aleatoria separada, consideramos las diferencias $D_i, i=1, \dots, n$, como una muestra aleatoria de tamaño n de una distribución normal con media $\mu = \mu_1 - \mu_2$ y varianza σ^2 . La prueba para igualdad de medias es:

$$H_0: \mu = 0, \quad A: \mu \neq 0$$

La estadística de prueba apropiada para probar esta hipótesis es:

$$t = n^{1/2} \bar{D} / S_D$$

Donde \bar{D} es el promedio de diferencias y S_D es

$$S_D = \left[\sum_{i=1}^n \frac{(D_i - \bar{D})^2}{n-1} \right]^{1/2}$$

Si la hipótesis nula es verdad, esta estadística es t -de Student con $n-1$ grados de libertad. La región crítica es: $|t| \geq t_{(1-\alpha/2), (n-1)}$.

Ejemplo 3.4 Supóngase que los siguientes pares son observados de las variables X y Y .

i	1	2	3	4	5	6	7	8
x_i	1	0	3	6	2	1	7	6
y_i	-2	4	-2	1	4	7	3	4
$D_i = x_i - y_i$	3	-4	5	5	-2	-6	4	2

Entonces $\bar{D} = \sum_{i=1}^8 D_i / 8 = .875$,

$$S_D^2 = \sum_{i=1}^8 \frac{(D_i - \bar{D})^2}{7} = 18.41$$

Y la estadística de prueba es $t = n^{1/2} \bar{D} / S_D$
 substituyendo tenemos $t = .577$

Si $\alpha = 0.05$, entonces $t_{0.975, 7} = 2.305$,
 otra vez no podemos rechazar la hipótesis H_0 .

Las dos pruebas examinadas en esta sección permiten rechazar para valores extremos, menores o mayores a los de hipótesis. Por ejemplo, para probar la hipótesis $H_0: \mu_1 \leq \mu_2$ contra $A: \mu_1 > \mu_2$ donde las muestras son independientes, se usa una prueba similar, solo que ahora el rechazo se hace posible sólo para valores grandes de la estadística de prueba. El rechazo

ocurre sólo en uno de los lados de los valores de la hipótesis. La única diferencia en el procedimiento es el cálculo de la región crítica. En este caso es:

$$t \geq t(1-\alpha), (n_1+n_2-2)$$

Algo similar existe para el caso en que no existe la independencia.

Para resumir, la prueba t se usa para probar la igualdad de medias cuando se muestrean dos poblaciones normales. Si las muestras son independientes, existe una prueba exacta con sólo suponer que tienen varianzas iguales. Para el caso de muestras no independientes se consideran las diferencias.

3.4 Prueba F

La prueba F se usa para probar igualdad entre varianzas de dos poblaciones normales. Supóngase que se toman dos muestras de las poblaciones con medias μ_1 y μ_2 y varianzas σ_1^2 y σ_2^2 respectivamente. Representamos por x_1, \dots, x_{n_1} la muestra tomada de la primera población y y_1, \dots, y_{n_2} la muestra de la segunda. Las hipótesis son: $H_0: \sigma_1^2 = \sigma_2^2, A: \sigma_1^2 \neq \sigma_2^2$

La prueba estadística usada es: $F = S_1^2 / S_2^2$.

Donde S_1^2 es la varianza muestral para la primera población y S_2^2 lo es de la segunda. Si H_0 es cierta entonces, F se distribuye F (vease sección 2.8.10) con $\nu_1 = n_1 - 1$ y $\nu_2 = n_2 - 1$. La región crítica para la

prueba es: $F \leq F_{\alpha/2, (n_1-1, n_2-1)}$ o

$F > F_{(1-\alpha/2), (n_1-1, n_2-1)}$

Ejemplo 3.5 Supóngase que se toman muestras de dos poblaciones normales y los resultados son

i	1	2	3	4	5	6	7	8	9	10
x _i	-5	4	-8	14	21	16	0	1		0
i	1	2	3	4	5	6	7	8		
y _i	0	1	-1	2	-2	0	1	2		

Entonces $S_1^2 = 73.73$ y $S_2^2 = 1.98$

La estadística de prueba es: $F = \frac{S_1^2}{S_2^2} = \frac{73.73}{1.98} = 37.2$

Si $\alpha = 0.05$, la región crítica es

$F \leq F_{0.025, (9, 7)} = 0.238$ o $F > F_{0.975, (9, 7)} = 4.8$

Entonces la hipótesis de igualdad de varianzas se rechaza. Nótese que con estas muestras, la hipótesis es rechazada aún si α se escoge inclusive tan pequeño como 0.00001. Entonces podemos decir con buen grado de confianza que las varianzas poblacionales son diferentes.

Recuérdese de la sección anterior que tenemos una prueba exacta para comparación de medias de dos poblaciones normales cuando las muestras son independientes, suponiendo la igualdad de varianzas, la

prueba F es usada para esto último. Justamente como ocurre con la prueba t, existe una prueba F con desigualdad. La prueba que se usa es la misma. La definición de la región crítica cambia. Las hipótesis apropiadas y regiones críticas son:

Hipótesis	Región crítica
$H_0: \sigma_1^2 \leq \sigma_2^2, A: \sigma_1^2 > \sigma_2^2$	$F > F_{(1-\alpha)}(n_1-1, n_2-1)$
$H_0: \sigma_1^2 > \sigma_2^2, A: \sigma_1^2 < \sigma_2^2$	$F \leq F_{(\alpha)}(n_1-1, n_2-1)$

Ejemplo 3.6 Con los datos del ejemplo 3.5 que conducen a rechazar la hipótesis de igualdad de varianzas, sean las hipótesis nuevas.

$$H_0: \sigma_1^2 \leq \sigma_2^2, A: \sigma_1^2 > \sigma_2^2$$

Si $\alpha = 0.05$, la región crítica es:

$$F > F_{0.95}(9, 7) = 3.68.$$

Así, con el cálculo de $F = 37.2$, también rechazamos esta hipótesis nula.

En resumen, la prueba F se usa para comparar la varianza de dos poblaciones normales. Existen pruebas para igualdad y desigualdad. La prueba es la misma únicamente cambiando la región crítica.

3.5 Prueba Chi-Cuadrada de Bondad de ajuste

Una de las mayores dificultades a las que se enfren-

ta un investigador al diseñar una simulación, es la caracterización de las variables aleatorias de interés. Antes de modelar la operación de un centro de servicio, por ejemplo, debemos caracterizar la distribución de los arribos. En muchos casos la variable aleatoria de interés se supone que tiene alguna distribución particular, naturalmente los resultados obtenidos por la simulación son en general muy sensibles a esta suposición. Necesitamos un método por medio del cual se pruebe una suposición acerca de la distribución de una variable aleatoria. La prueba Chi-Cuadrada de bondad de ajuste es usada en este sentido.

Esta prueba hace una comparación entre el valor esperado y el valor observado en un rango de la variable aleatoria. La hipótesis de que la distribución supuesta y lo observado son iguales es probada usando la estadística

$$\chi^2 = \sum_{i=1}^r \frac{(O_i - E_i)^2}{E_i}$$

Donde O_i es la frecuencia observada en el intervalo i y E_i es el valor esperado de observaciones en el mismo intervalo si suponemos cierta la distribución especificada. Damos a continuación un procedimiento para llevar a cabo esta prueba.

1.- Construir una tabla de frecuencias de los valores observados de la variable. El número de intervalos es arbitrario; sin embargo, la experiencia ha demostrado que intervalos con menos de tres a cinco

observaciones tiende a distorsionar los resultados.

2.- Calcular el valor esperado para cada intervalo bajo la suposición de que la distribución especificada como hipótesis es cierta. Los parámetros de la distribución supuesta son generalmente estimados muestralmente.

3.- Calcular la cantidad $(O_i - E_i)^2 / E_i$ $i=1..r$. donde O_i y E_i son los especificados con anterioridad.

4.- Calcular la estadística Chi-Cuadrada

$$\chi^2 = \sum_{i=1}^r \frac{(O_i - E_i)^2}{E_i}$$

Los grados de libertad para esta estadística son $r - p - 1$, donde r es el número de intervalos y p el número de parámetros estimados para la distribución establecida.

Como un ejemplo, supóngase que hay diez intervalos y que la distribución es supuestamente normal. Como estimamos tanto μ como σ^2 los grados de libertad para la estadística serían $10 - 2 - 1 = 7$. Si la distribución se supone Poisson se requiere un solo parámetro y los grados de libertad son $10 - 1 - 1 = 8$.

5.- Escoger un valor de α y probar la hipótesis rechazamos la hipótesis de igualdad de distribución si

$$\chi^2 \geq \chi^2_{(1-\alpha), (r-p-1)}$$

Ejemplo 3.7 Un investigador intenta caracterizar los arribos a un centro de servicio. Después de un período largo de tiempo, el investigador construye la siguiente tabla de frecuencias. La cantidad de interés es el número de arribos por hora.

CLIENTES	FRECUENCIA	CLIENTES	FRECUENCIA
0-2	13	12-14	16
3-5	17	15-17	11
6-8	21	18-20	5
9-11	42		

El investigador establece la hipótesis de que los arribos son Poisson. Un estimador para λ , el promedio de arribos por hora, se calcula de la tabla de frecuencias como $\bar{x} = 9.0$. Con $\lambda = 9$ se construye la tabla que indica los valores observados y esperados en cada intervalo (7).

#intervalo	Cotas	observado	esperado
1	0-2	13	0.75
2	3-5	17	13.75
3	6-8	21	42.5
4	9-11	42	43.375
5	12-14	16	19.5
6	15-17	11	4.0
7	18-20	5	1.125

Uniendo los dos primeros intervalos y los dos últimos tenemos

$$\chi^2 = \sum_{i=1}^5 \frac{(O_i - E_i)^2}{E_i} = 51.2$$

Entonces $\chi^2 = 52.2 > \chi^2_{0.99,3} = 11.2$

La hipótesis de que los arribos son Poisson puede ser rechazada con $\alpha = 0.05$

3.6 Prueba Kolmogorov - Smirnov

Una alternativa a la prueba Chi-cuadrada es la prueba de Kolmogorov-Smirnov. Como en la prueba Chi, ésta es usada para probar igualdad de distribuciones. En ocasiones esta prueba es más poderosa que la prueba Chi. Esta prueba es mejor al momento de detectar pequeñas diferencias entre las distribuciones real y supuesta. La descripción de la prueba se hace ahora.

1. Sea $S(x)$ la función de distribución empírica para una muestra de N observaciones.

2.- Sea $F(x)$ la función de distribución supuesta

3.- Para cada punto calcular $F(x_i) - S(x_i)$.
Sea $D = \max x_i |F(x_i) - S(x_i)|$

4.- Escoger un valor de d , y si el valor calculado D es más grande que el valor crítico tabulado al nivel de significancia α , se rechaza H_0 .

Ejemplo 3.8 Consideremos los datos del ejemplo 3.7 usando el lado derecho de cada intervalo como el valor muestral, podemos contruir la siguiente tabla.

x	$S(x)$	$F(x)$	$ F(x) - S(x) $
2	0.104	0.006	0.098
5	0.204	0.116	0.124
8	0.408	0.456	0.048
11	0.744	0.803	0.059
14	0.872	0.959	0.087
17	0.96	0.995	0.035
20	1.00	1.000	0

En esta tabla la hipótesis supuesta es que se trata de una Poisson con $\lambda = 9.0$. La media muestral es 9.02

Si $\alpha = 0.05$, el valor crítico para esta tabla es $(1.37) / (\sqrt{125}) = 0.122$. Como $D = 0.124 > 0.122$ la hipótesis se rechaza.

Referencias

[3.1] Introduction to mathematical statistics
Robert V. Hogg & Allen T. Craig
Macmillan P. Co. Inc. New York.

[3.2] Cramér, H., Mathematical Methods of statistics, Princeton University Press, Princeton, N.J., 1946.

CAPITULO IV

Introducción

En muchas simulaciones se tienen eventos cuya ocurrencia es aleatoria o envuelven atributos cuyos valores también tienen cierta carga aleatoria. Por ejemplo, considérese un sistema de cómputo de propósito general. Un evento a ser modelado puede ser el acceso a un registro que se encuentra en algún dispositivo de acceso directo (disco por ejemplo). La duración de este evento tiene un valor aleatorio que depende entre otras cosas de la posición relativa del registro con respecto a la cabeza lectora. Considerando el mismo sistema, para la ejecución de algún programa es necesario que sea cargado en memoria principal, así, un atributo asociado a este evento puede ser el tamaño del programa cuyo valor es aleatorio.

Por razones como éstas, uno de los requerimientos de todo modelo de simulación es la facilidad de generar números aleatorios. Necesitaremos asignar valores particulares para aparentar la ocurrencia de eventos aleatorios en una simulación. En el ejemplo anterior sobre la simulación de un sistema de cómputo, debemos asignar un valor particular al tiempo requerido para acceder un registro cada vez que este evento sea simulado. En el caso de las peticiones de memoria para cargar programas debemos asignar un valor de la petición cada que se simule una petición de memoria.

En este capítulo se discutirán los métodos para la generación de números aleatorios en computadoras digitales. En los últimos años se han sugerido, usado y probado muchas técnicas para la generación de números aleatorios. Algunas de éstas basadas en fenómenos aleatorios, otras en procedimientos recursivos.

Inicialmente se usaron métodos manuales, incluyendo técnicas tales como lanzar monedas, rodar dados, extraer cartas, y otras. Estos métodos por su naturaleza eran muy lentos además de no permitir una repetición idéntica que permitiera reproducir una simulación.

Con el advenimiento del computador fue posible obtener números aleatorios con la calidad de reproducibilidad, no acusada por los primeros métodos. Un método para la generación de números aleatorios sobre un computador consistía en la elaboración de una tabla que era almacenada en la memoria del computador. En 1955 la RAND Corporation [4.1] dio a conocer una tabla de un millón de dígitos aleatorios. La ventaja de este método era una fácil reproducibilidad, sus desventajas eran la lentitud de acceso y una posible cobertura de toda la tabla.

4.1 Números Pseudo aleatorios

En vista de los problemas mencionados anteriormente se comienzan a proponer otras técnicas

para la obtención de números aleatorios en el computador. Antes de considerar algunas de tales técnicas, discutiremos algunas de las características deseables en un generador de números aleatorios.

- 1.- Los números producidos deberán seguir una distribución uniforme, ésta sería la mejor garantía de que se trata de números en verdad aleatorios. La prueba de uniformidad se puede hacer aplicando lo del capítulo III.
- 2.- Los números producidos deberán ser estadísticamente independientes. Esto es, el valor de uno de los números en la secuencia no debe afectar el valor del siguiente número.
- 3.- La secuencia de números aleatorios debe ser reproducible. Esto permite la replicación del experimento simulado.
- 4.- La secuencia debe ser no repetitiva para cualquier longitud deseada. Esto no es posible teóricamente, pero para propósitos prácticos, una longitud del ciclo de repetición específica puede ser suficiente. El ciclo de repetición de un generador es conocido como su período.
- 5.- La generación misma de la secuencia debe ser muy rápida. Durante una simulación se usan muchos números aleatorios. Si el generador es lento, puede aumentar el tiempo de simulación y por tanto su costo.

6.- El método usado en la generación de números aleatorios debe usar poca memoria. Los modelos de simulación generalmente requieren de mucha memoria. Así, el generador debe usar poca.

Con estos requerimientos es casi imposible evaluar las propuestas dadas para compensar la falta de reproducibilidad de las secuencias aleatorias. La primer propuesta es generar la secuencia de alguna manera y almacenarla, digamos en cinta. Esta propuesta sería insatisfactoria por el tiempo ocupado en obtener un número ya que cada que se requiere uno hay que leer en la cinta. Esta técnica también tiene período corto a no ser que se almacene una gran cantidad de números. Una alternativa consiste en almacenar en memoria principal para tener un tiempo de respuesta menor sin embargo se tiene la desventaja que el tamaño de memoria ocupado sería directamente proporcional con el tamaño de la secuencia. La tercera técnica y una de las más usadas consiste en usar un valor inicial y generar algorítmicamente la secuencia. Esta técnica desaparece los problemas de rapidez y espacio pero podría sufrir problemas de independencia y reproducibilidad.

El uso de un algoritmo para generar números aleatorios parece violar los principios básicos de aleatoriedad. Por esta razón la generación de números algorítmicamente se le llama generación pseudo aleatoria.

Estos números satisfacen ciertos criterios de

aleatoriedad pero siempre empiezan con un cierto valor inicial llamado la raíz y proceden de una manera determinística, una forma repetitiva. Mucho cuidado debe tenerse al usar secuencias pseudoaleatorias para asegurar que un buen grado de aleatoriedad este presente (uniformidad) y que el ciclo de repetición o período sea suficientemente grande.

En la siguiente sección se estudiarán diferentes algoritmos para generar números pseudoaleatorios. En el apéndice II se encuentra la implementación en un computador de un generador de números pseudoaleatorios junto con las pruebas estadísticas de uniformidad.

4.2 Algoritmos para generar números Pseudoaleatorios

Una buena cuota de trabajo ha sido hecha en lo que se refiere al diseño y prueba de algoritmos que producen secuencias de números pseudoaleatorios. Los algoritmos difieren no sólo en la técnica usada sino también en la velocidad de generación, longitud del ciclo de repetición o período y facilidad de programación, discutimos aquí algunos de los algoritmos más comunes.

4.2.1 Método de los medios cuadrados

La técnica de los medios cuadrados fue desarrollada a mediados de los 40's por John Von Neumann [4.21]. La técnica usa un valor inicial o raíz. El número se eleva al cuadrado, y los dígitos del centro de este cuadrado son usados como segundo número

de la secuencia. Este segundo número nuevamente se eleva al cuadrado y los dígitos del centro se toman como tercer número (recursión). El algoritmo continúa de esta forma.

Ejemplo 4.1 Supóngase que se desea generar una secuencia de números aleatorios de cuatro dígitos usando el método de los cuadrados centrales. Sea 3187 la raíz. Entonces

$$x_0 = 3187$$

$$(3187)^2 = 10156969 \therefore x_1 = 1569$$

$$(1569)^2 = 02461761 \therefore x_2 = 4617$$

$$(4617)^2 = 21316689 \therefore x_3 = 3166$$

$$(3166)^2 = 10023556 \therefore x_4 = 0235$$

$$(0235)^2 = 00055225 \therefore x_5 = 0552$$

Este proceso continúa y produce:

$$x_6 = 3047, x_7 = 2842, x_8 = 0769, x_9 = 5913, x_{10} = 9635.$$

Esta técnica tiene algunas limitaciones. Primero, las secuencias generadas por este método tienen en general un período corto, segundo, si en algún momento se llega a obtener el cero en la secuencia entonces todos los siguientes elementos son cero.

Los problemas que tiene este método llevan a desarrollar algoritmos alternativos para la generación de secuencias aleatorias. Un método que aparece es el muy usado congruencial.

4.2.2 Método Congruencial lineal

Muchos de los generadores que se usan son modificaciones al esquema de congruencias dado por Lehmer [4-31]. En este algoritmo la secuencia de números es generada por la relación recursiva.

$$x_{n+1} = (ax_n + c) \text{ mód } m, \quad n \geq 0.$$

El valor inicial " x_0 " se le llama raíz, la constante " a " se le llama multiplicador, " c " el incremento y " m " el módulo. La selección de estos valores (fijos durante toda la generación) tiene un efecto determinante en la calidad de la secuencia.

Ejemplo 4.2 Sea $a=2, c=3, m=10, x_0=0$. Entonces

$$x_0 = 0$$

$$x_1 = (2x_0 + 3) \text{ mód } 10 = 3$$

$$x_2 = (2x_1 + 3) \text{ mód } 10 = 9$$

$$x_3 = (2x_2 + 3) \text{ mód } 10 = 1$$

$$x_4 = (2x_3 + 3) \text{ mód } 10 = 5$$

$$x_5 = (2x_4 + 3) \text{ mód } 10 = 3$$

$$x_6 = (2x_5 + 3) \text{ mód } 10 = 9$$

Ejemplo 4.3 Sea $a=2, c=0, m=10, x_0=1$. Entonces

$$x_0 = 1$$

$$x_1 = (2x_0) \text{ mód } 10 = 2$$

$$x_2 = (2x_1) \text{ mód } 10 = 4$$

$$x_3 = (2x_2) \text{ mód } 10 = 8$$

$$x_4 = (2x_3) \text{ mód } 10 = 6$$

$$x_5 = (2x_4) \text{ mód } 10 = 2$$

$$x_6 = (2x_5) \text{ mód } 10 = 4$$

El ejemplo 4.3 ilustra el caso en el cual $c=0$. Este algoritmo es llamado congruencial multiplicativo. Si $c \neq 0$, el esquema es llamado congruencial mixto.

Los ejemplos anteriores muestran el corto período que tienen (los números se repiten). Knuth [4.2] muestra que la elección de a, c, m y x_0 es determinante para una buena calidad en la secuencia. Las recomendaciones que se extraen de este trabajo son:

1.- Elección de m . Como el período siempre es menor que m , una elección de un valor grande para m es una petición pertinente, un valor de m que facilite la solución de la relación congruencial debe ser usado. Para máquinas que usan una representación binaria, un valor de $2^k - 1$, donde k es el tamaño de la palabra de la memoria es una excelente elección.

2.- Elección de a y c . Una secuencia generada por un esquema congruencial lineal tiene período m (toda la capacidad de la máquina).

si y sólo si

i) c es primo relativo de m

ii) $a \equiv 1 \pmod{g}$ para cualquier factor primo g de m

iii) $a \equiv 1 \pmod{4}$ si m es un múltiplo de 4.

La condición i) significa que el divisor común más grande de c y m es la unidad.

La condición ii) significa que $a = [a/g] + 1$. sea g un factor primo de m ; entonces denotando $k = [a/g]$, podemos poner

$$a = 1 + gk.$$

La condición iii) dice que $a = [4][a/4] + 1$, $m/4$ es

un entero.

Como muchos computadores emplean ya sea el sistema decimal o el sistema binario escogemos

$$m = 2^{\beta} - 1 \text{ ó } m = 10^{\beta} - 1$$

donde β denota la longitud de palabra. Discutiremos ambos casos.

Para un computador binario, con $m = 2^{\beta} - 1$, de la relación recursiva que define la secuencia y la condición i) se tiene que "C" debe ser impar y

$$a \equiv 1 \pmod{4},$$

$$\text{así } a = 2^r + 1, \quad r \geq 2.$$

Por ejemplo para un computador de 35 bits una elección óptima es: $m = 2^{35} - 1, a = 2^7 + 1, c = 1$.

En el apéndice II se encuentra el generador congruencial para una POP 1134 cuya longitud de palabra es de 15 bits (En realidad son 16 bits sólo que uno lo usa como bit de signo).

Para un computador decimal $m = 10^{\beta} - 1$. Una forma de generar un período máximo es la siguiente:

"C" debe ser positivo y no divisible por 2 ó 5,
"a" debe cumplir $a \equiv 1 \pmod{20}$ o alternativa-
mente $a = 10^r + 1, r > 1$. Una buena elección
es: $a = 101, c = 1$

La generación de números aleatorios que estén en el (0,1) a partir de los obtenidos por

$$x_{i+1} = (ax_i + c) \text{ mód } m \quad i=1,2,\dots$$

x_0 fijo.

Se obtiene haciendo $U_i = \frac{x_i}{m} \quad i=1,2,\dots$

Claramente tal secuencia se comporta como la original sólo que escalada al (0,1).

El procedimiento para generar números pseudo-aleatorios sobre un computador binario puede ser como sigue:

- 1.- Escoger cualquier número impar como x_0 .
- 2.- Escoger un entero $a = 8r \pm 3$, donde r es cualquier entero positivo. Escoger " a " cerca de $2^{\beta/2}$ (si $\beta=35$, $a = 2^{17} + 3$ es buena elección).
- 3.- Calcular x_1 usando aritmética entera de punto fijo. Este producto consiste de $2 \cdot \beta$ bits de los cuales los β bits más altos se descartan y los más bajos son x_1 .
- 4.- calcular U_1 como $U_1 = x_1 / 2^\beta$
- 5.- Cada número x_{i+1} se obtiene de los β bits más bajos del producto $a \cdot x_i$.

6.2.3 Generador congruencial aditivo

La técnica de este generador requiere de más de una raíz, se requieren n valores x_1, \dots, x_n . Esta secuencia puede ser obtenida usando alguna otra

técnica; La aplicación del algoritmo produce una extensión de la secuencia a x_{n+1}, x_{n+2}, \dots . El algoritmo es:

$$x_j = (x_{j-1} + x_{j-n}) \pmod{m}$$

La principal ventaja de este método es su velocidad; no se requieren multiplicaciones. Permite períodos mayores de "m". Como lo menciona Knuth [4.2] la conducta teórica de este método no tiene un estudio a fondo como la técnica congruencial.

Ejemplo 4.4 Sea $m=10$, la secuencia de entrada es 1, 2, 4, 8, 6 generada en el ejemplo 4.3. Se extiende la secuencia con el congruencial aditivo.

- $x_1 = 1$
- $x_2 = 2$
- $x_3 = 4$
- $x_4 = 8$
- $x_5 = 6$

$$x_6 = (x_5 + x_1) \pmod{10} = (6 + 1) \pmod{10} = 7$$

$$x_7 = (x_6 + x_2) \pmod{10} = (7 + 2) \pmod{10} = 9$$

$$x_8 = (x_7 + x_3) \pmod{10} = (9 + 4) \pmod{10} = 3$$

$$x_9 = (x_8 + x_4) \pmod{10} = (3 + 8) \pmod{10} = 1$$

$$x_{10} = (x_9 + x_5) \pmod{10} = (1 + 6) \pmod{10} = 7$$

4.3 Pruebas y validez de secuencias Pseudoaleatorias

La validez estadística de los resultados de un modelo de simulación depende del grado de aleatoriedad de los generadores empleados. A partir de esto, se han desarrollado muchos procedimientos estadísticos para la prueba de generadores de números aleatorios. Se han usado pruebas tanto analíticas como empíricas para investigar las propiedades aleatorias de estos generadores. En este trabajo se sigue un generador de números aleatorios al cual

se le hacen pruebas de uniformidad, esto es, probar que una secuencia aleatoria obtenida a partir de nuestro generador es originaria de una población uniforme en el $[0,1]$. El programa para un generador por uno de los métodos descritos y las pruebas estadísticas aparecen en el apéndice II. Las pruebas

son:

Estimación de parámetros

Prueba Chi-Cuadrada y Kolmogorov - Smirnov.

Prueba de independencia

Existen muchas más pruebas sobre generadores. Knuth [4.21] describe diez de tales pruebas.

4.4 Generación de variables aleatorias

Ya vimos cómo podemos generar una secuencia de números aleatorios donde todos los elementos tengan la misma probabilidad de aparecer. Ahora podremos generar una secuencia que cumpla con una distribución de probabilidad específica. Antes de esto describiremos dos métodos generales para generar números aleatorios con una distribución dada.

4.4.1 Método de inversión

Dada una función de distribución $F(x)$ siendo esta monótona creciente y contenida en el $[0,1]$. Como se muestra en la figura 4.1.

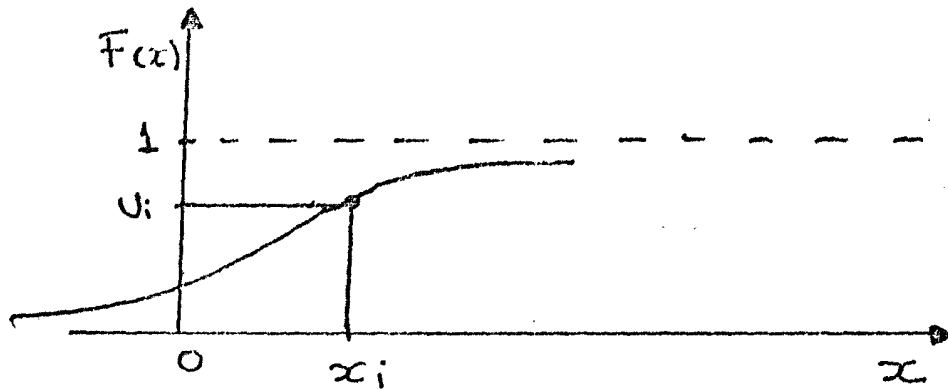


Figura 4.1 Método de Inversión

Entonces, si $\{U_i\}$ es una secuencia con distribución uniforme en el $[0,1]$. Definiendo a $\{x_i\}$ como:

$$F(x_i) = U_i \quad \text{o} \quad x_i = F^{-1}(U_i)$$

Entonces $\{x_i\}$ se espera que sea una secuencia con función de distribución $F(x)$ y por tanto con una densidad $f(x)$ dada. Nótese que es fundamental un generador uniforme en $[0,1]$ para la anterior construcción. Existen dos restricciones fuertes para la aplicación del método y es que se debe tener la distribución y su inversa.

4.4.2 Método del rechazo

Este método es aplicable cuando la densidad $f(x)$ tiene sus límites superior e inferior en un rango $[a,b]$ y un máximo en f_{\max} . Así, el método puede ser descrito como sigue.

Dada la función de densidad $f(x)$; donde $f(x)$ es mayor o igual a cero, para toda x , y además

$$f(x) = 0, \quad \text{para } x < a \quad \text{o} \quad x > b$$
$$\max_{x \in [a,b]} [f(x)] = f_{\max}$$

Como lo muestra la figura 4.2

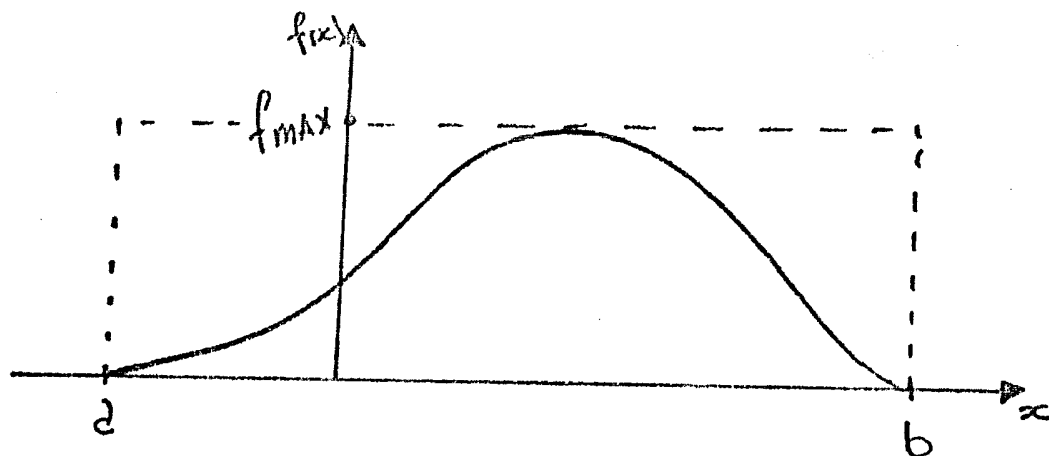


Figura 4.2 Método del Rechazo

Sean $\{U_i\}$ y $\{V_i\}$ secuencias aleatorias independientes cada una con distribución uniforme estándar.

Defínase $\{x_i\}$ como

$$\{x_i\} = a + b\{U_i\} \text{ si } \{V_i\} \text{ es menor que } \frac{f(a+b\{U_i\})}{f_{\max}}$$

y en caso contrario se rechaza.

Así, $\{x_i\}$ es una secuencia con densidad $f(x)$.

Restricciones para su aplicación:

* Se generan pares $\{a + b(U_i), f_{\max} \cdot V_i\}$ con distribución uniforme en el rectángulo $[a, b] \times [0, f_{\max}]$. Se aceptan $x_i = a + b \cdot U_i$ si el par cae bajo la gráfica de la función de densidad (debemos conocerla).

* Ya que se rechazan algunos valores, la eficiencia del método es:

$$E = \frac{1}{f_{\max} (b-a)}$$

4.5 Programoteca de generación de distribuciones

Ya que hemos visto la manera de generar variables aleatorias en general, ahora veremos cómo aplicar estos métodos a densidades particulares y cómo implantar la generación de estas distribuciones en el computador.

En la definición de las distribuciones se usarán las siguientes variables:

X : La variable aleatoria

$f(x)$: La función de densidad de X

$F(x)$: La función de distribución de X

a : mínimo

b : máximo.

μ : media

λ : media.

$(\sigma)^2$: Varianza.

σ : desviación estándar.

α : parámetro de la densidad.

En el apéndice III se encuentra la relación de los parámetros, rangos de valores y características para cada rutina de la programoteca.

4.5.1 Distribución Uniforme

La función de densidad de esta distribución es:

$$f(x) = \begin{cases} 1/(b-a) & , x \text{ en } [a, b] \\ 0 & , x \text{ fuera de } [a, b] \end{cases}$$

$$\text{La distribución es: } F(x) = \begin{cases} 0 & \text{si } x < a \\ \frac{x-a}{b-a} & \text{si } x \in [a, b] \\ 1 & \text{si } x > b \end{cases}$$

Para simular una distribución uniforme en $[a, b]$, usamos la transformación inversa de la función de distribución (método de la sección 4.4.1)

$$x = F^{-1}(U) = a + (b-a)U; \quad a < x < b$$

donde U es un aleatorio en $[0, 1]$. La densidad uniforme en $[a, b]$ se muestra en la figura 4.3.

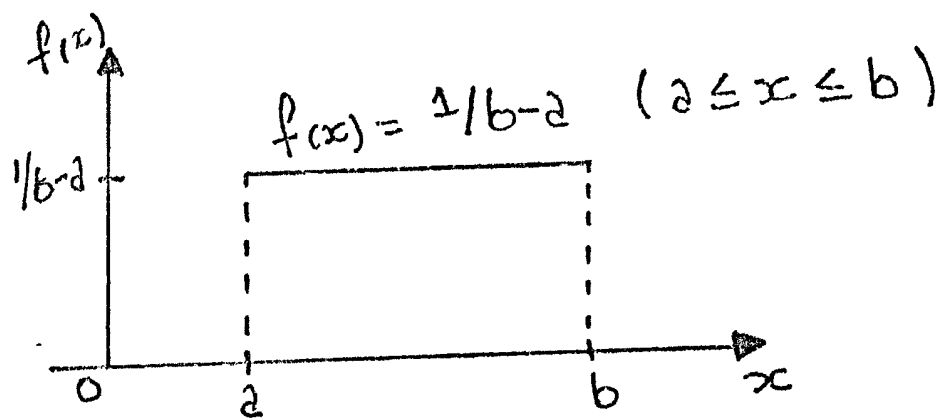


Figura 4.3 Distribución Uniforme en $[a, b]$

La implantación en lenguaje FORTRAN de esta distribución es:

```
Subroutine UNIFORM (NSEED, RAN, A, B)
CALL RAND (NSEED, RAN)
RAN = A + (B - A) * RAN
RETURN
END
```

4.5.2 Distribución de Bernoulli

Esta distribución está definida como:

$$P(X=x) = f(x) = \begin{cases} p & \text{si } x = 1 \\ 1-p & \text{si } x = 0 \end{cases}$$

Generándola desde una distribución uniforme en $[0,1]$:

si $0 \leq U < p$, entonces $x = 1$
 si $p \leq U \leq 1$, entonces $x = 0$

Donde U es una variable aleatoria uniforme en $[0,1]$ y p ($0 < p < 1$) es un parámetro de Bernoulli. La figura 4.4 muestra esta distribución.

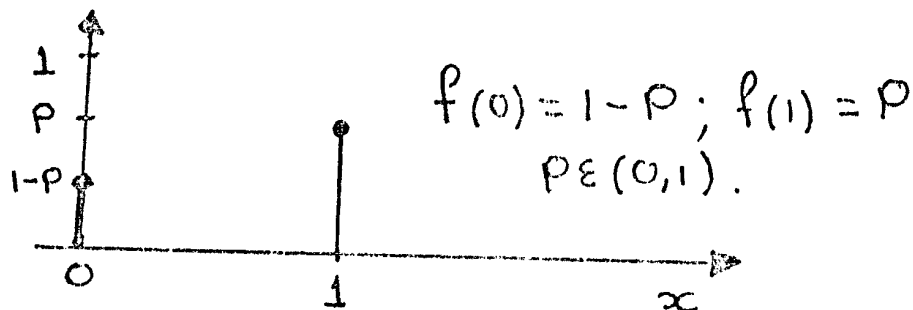


Figura 4.4 Distribución Bernoulli

La implantación en FORTRAN para la Bernoulli es:

```

SUBROUTINE BERN (USEED, NBERN, P)
C
NBERN SE DISTRIBUYE BERNOULLI, P es el parametro.
NBERN = 0
CALL RAND (USEED, P)
IF (R .LE. P) NBERN = 1
RETURN
END
  
```

4.5.3 Distribución Binomial

Esta densidad está definida como:

$$f(x) = \begin{cases} \frac{n!}{x!(n-x)!} p^x (1-p)^{n-x}, & x=0..n. \\ 0 & \text{en otro caso} \end{cases}$$

La variable aleatoria que da origen a esta densidad puede ser vista como el número de éxitos ("p" la probabilidad de éxito) dentro de N ensayos de Bernoulli. Considerando lo anterior implantamos la rutina generadora de binomiales. La distribución es mostrada en la figura 4.5

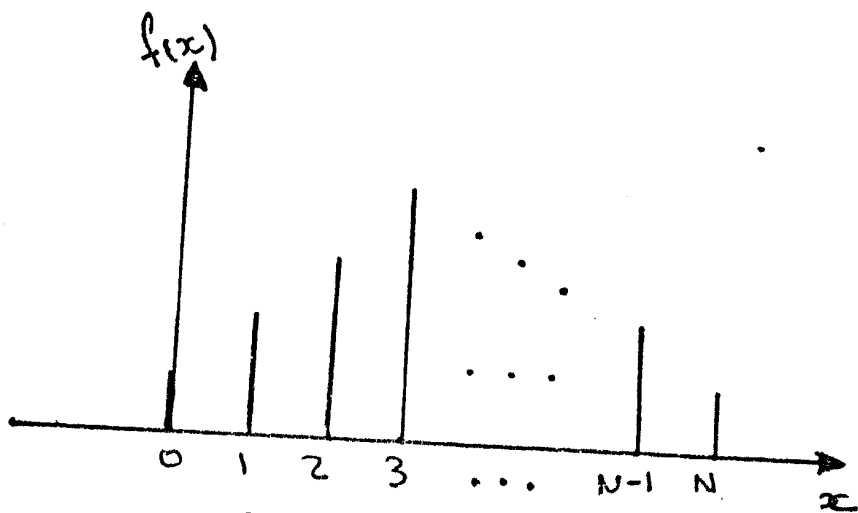


Figura 4.5 Distribución Binomial

La rutina en FORTRAN ES.

```
SUBROUTINE BINOM (NSEED, NRAN, N, P)
NRAN = 0
DO 6 I = 1, N
CALL RAND (NSEED, R)
IF (R .LE. P) NRAN = NRAN + 1
6 CONTINUE
RETURN
END
```

4.5.4 Distribución Exponencial

Esta distribución que es de las más usadas tiene la siguiente densidad.

$$f(x) = \begin{cases} \frac{1}{\text{LAMBDA}} e^{-\left(\frac{x}{\text{LAMBDA}}\right)}, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

Con función de distribución.

$$F(x) = \begin{cases} 1 - e^{-\left(\frac{x}{\text{LAMBDA}}\right)}, & x \geq 0 \\ 0, & x < 0 \end{cases}$$

Considerando $U = F(x)$, así
 $x = F^{-1}(U) = -(\text{LAMBDA}) \ln(1-U)$.
 donde $x > 0$ y LAMBDA es la media.

Para generar x con distribución exponencial hacemos el cambio $V = 1-U$, así

$$x = -(\text{LAMBDA}) \ln(V) = -(\text{LAMBDA}) \ln(1-U)$$

donde U es aleatorio en el (0,1) entonces $1-U$ también tiene distribución uniforme en el (0,1). La distribución exponencial es mostrada en la figura 4.6

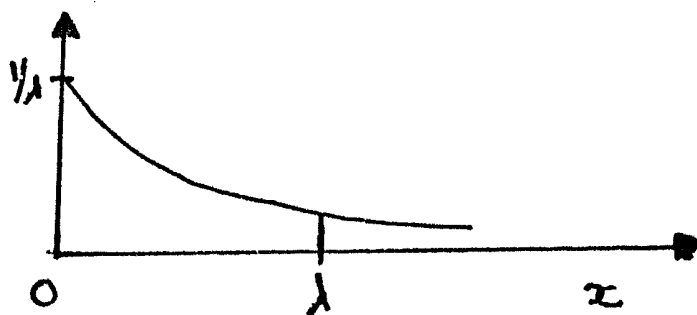


Figura 4.6 Distribución exponencial.

La rutina generadora para la distribución exponencial es dada en seguida.

```

SUBROUTINE EXPON (NSEED, RAN, RMEDIA)
CALL RAND (NSEED, RAN)
RAN = -RMEDIA * ALOG (RAN)
RETURN
END

```

5.5.5 Distribución Erlang

La función de densidad es:

$$f(x) = \begin{cases} \frac{(\text{ALFA})^k}{(k-1)!} x^{(k-1)} e^{-(\text{ALFA})x} & , x > 0 \\ 0 & , x < 0 \end{cases}$$

Se sabe que si X tiene distribución Erlang, puede verse como la suma de k variables aleatorias independientes con distribución exponencial y media $(\text{ALFA})^{-1}$. Esto es

$$X = \sum_{i=1}^k x_i = \sum_{i=1}^k \left(-\frac{1}{\text{ALFA}} \ln U_i \right) = -\frac{1}{\text{ALFA}} \ln \left(\prod_{i=1}^k U_i \right)$$

donde $U_i \ i=1..k$ son variables aleatorias independientes uniformes en $(0,1)$. La figura 4.7 muestra la distribución Erlang.

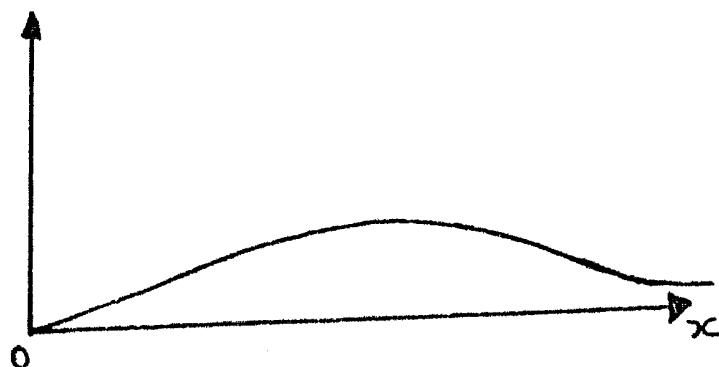


Figura 4.7 Distribución Erlang.

La implantación de esta distribución es:

```
SUBROUTINE ERLANG (NSEED, RAN, A, K)
PR = 1.0
DO 4 I = 1, K
CALL RAND (NSEED, RAN)
4 PR = PR * RAN
RAN = -A LOG (PR) / A
RETURN
END
```

4.5.6 Distribución Geométrica

La función de densidad es:

$$P(X=x) = f(x) = \begin{cases} p^x (1-p) & , x = 0, 1, 2, \dots \\ 0 & \text{en otro caso.} \end{cases}$$

La figura 4.8 muestra esta distribución. Una variable aleatoria X que genera una densidad geométrica puede ser vista como el número de fracasos ("p" la probabilidad de fracaso) antes del primer éxito ("1-p" la probabilidad de éxito) en ensayos sucesivos de Bernoulli. tomamos este hecho para implantar la subrutina que genera una distribución geométrica.

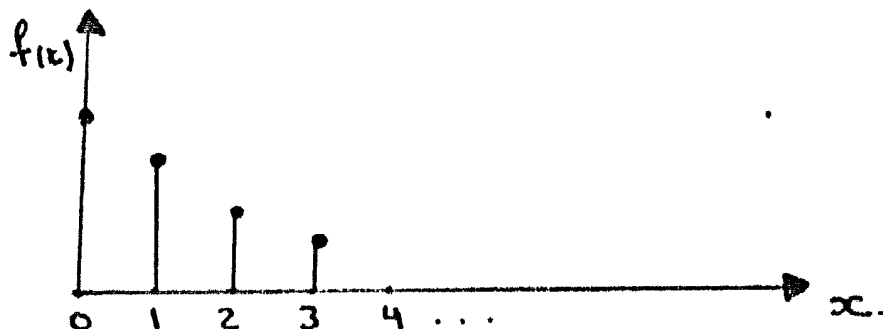


Figura 4.8 Distribución geométrica

```
SUBROUTINE GEO (NSEED, NRAM, P)
```

```
NRAM = 0
```

```
2 CALL RAND (NSEED, R)
```

```
IF (R .GT. P) RETURN
```

```
NRAM = NRAM + 1
```

```
GO TO 2
```

```
END
```

4.5.7 Distribución Normal

La función de densidad para esta distribución es:

$$f(x) = \frac{1}{\text{SIGMA} \sqrt{2\pi}} e^{-\frac{(x - \text{MU})^2}{2 \text{SIGMA}^2}} \quad x \in \mathbb{R}$$

Para generar esta distribución, basta generar una distribución normal estándar, esto es donde "MU" sea cero y "SIGMA" sea uno. Con lo anterior obtenido se tiene que si:

X es normal estándar entonces $Y = \text{MU} + X \cdot \text{SIGMA}$ es normal con media MU y desviación estándar SIGMA.

Haciendo uso del teorema central del límite:

Sean U_1, U_2, \dots, U_{12} variables aleatorias uniformes e independientes en $(0, 1)$. Entonces,

$$X = \sum_{i=1}^{12} U_i - 6$$

Se distribuye aproximadamente normal con media 0 y desviación estándar 1.

Una justificación de lo anterior puede ser vista en [4.5]

Este algoritmo es relativamente eficiente y barato sin embargo, podría no representar adecuadamente los extremos de la distribución (ver [4.5]). La gráfica normal se ilustra en la figura 4.9.

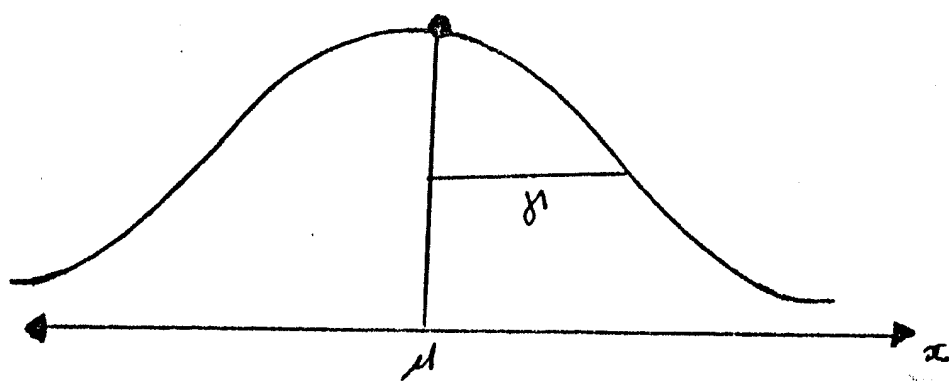


Figura 4.9 Distribución Normal

La implantación en FORTRAN es:

```

SUBROUTINE NORMAL (NSEED, RAN, RMEDIA, RDS)
SUM = 0.0
DO 4 I = 1, 12
CALL RAND (NSEED, RAN)
4 SUM = SUM + RAN
RAN = RDS * (SUM - 6.) + RMEDIA
RETURN
END

```

4.5.8 Distribución Poisson

La función de densidad para Poisson es:

$$P(X=x) = f(x) = \begin{cases} \frac{\text{ALFA}^x e^{-\text{ALFA}}}{x!} & , x = 0, 1, 2, \dots \\ 0 & , \text{en otro caso} \end{cases}$$

Si el tiempo entre eventos se distribuye exponencial, entonces el número de eventos ocurridos por unidad de tiempo se rige por esta distribución.

La idea para poder generar una variable Poisson consiste en generar tiempos entre arribos exponenciales e independientes; X_0, X_1, \dots tal que

$$\sum_{i=0}^x X_i \leq 1 < \sum_{i=0}^{x+1} X_i$$

Así entonces, X tendrá distribución Poisson.

Por otro lado, ya que

$$X_i = -\frac{1}{\text{ALFA}} \ln(U_i)$$

Donde U_i es una distribución uniforme en (0,1) obtenemos

$$\sum_{i=0}^x -\frac{1}{\text{ALFA}} \ln(U_i) \leq 1 < \sum_{i=0}^{x+1} -\frac{1}{\text{ALFA}} \ln(U_i) \quad \text{ó}$$

$$\ln \left(\prod_{i=0}^{x+1} U_i \right) \leq -\text{ALFA} < \ln \left(\prod_{i=0}^x U_i \right) \quad \text{ó}$$

$$\prod_{i=0}^{x+1} U_i \leq e^{-ALFA} < \prod_{i=0}^x U_i$$

Cuando la media ALFA es grande, entonces la cantidad e^{-ALFA} es pequeña y el procedimiento anterior consumirá mucho tiempo de procesador. Para evitar esto, se podría usar la siguiente aproximación para ALFA grande:

$$Z = \frac{X - ALFA}{\sqrt{ALFA}}, \text{ que es normal estándar}$$

Para obtener X, se genera Z normal estándar y se usa $X = ALFA + Z \sqrt{ALFA}$ redondeando al próximo entero.

La figura 4.10 muestra la densidad Poisson.

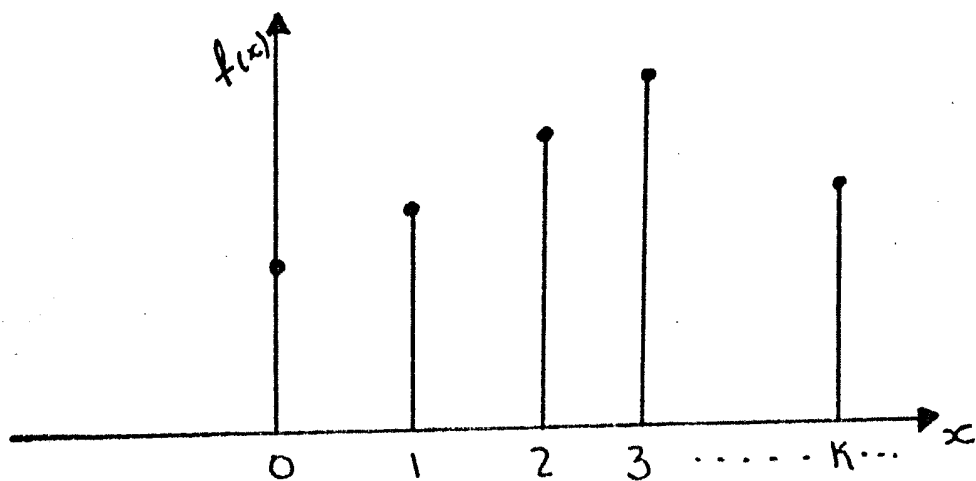


Figura 4.10 Distribución de Poisson

La subrutina de Poisson es:


```

SUBROUTINE POISSN (NSEED, NRAN, RMEDIA)
NRAN = 0
E = EXP (-RMEDIA)
PR = 1.0
4 CALL RAND (NSEED, R)
PR = PR * R
IF ((PR - E) .LT. 0.0) RETURN
NRAN = NRAN + 1
GO TO 4
END

```

4.6 Fórmulas Estadísticas

En estudios de simulación casi nunca podemos conocer las estadísticas poblacionales; generalmente necesitamos estimar las estadísticas poblacionales para producir los resultados de la simulación.

Algunas medidas usadas como estimaciones de las estadísticas de población son las siguientes:

Sean X_1, X_2, \dots, X_n datos simulados para una variable aleatoria X ; entonces,

$$\bar{X} = \sum_{i=1}^n X_i / n \quad \text{es la media muestral}$$

$$S = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X})^2 \quad \text{es la varianza muestral}$$

$$Y(k) = \frac{S^2(k)}{S^2} \quad \text{es la autocorrelación de retardo } k$$

$$S^2(k) = \frac{1}{n-1-k} \sum_{i=1}^{n-k} (X_i - \bar{X}) (X_{i+k} - \bar{X})$$

es la autocovarianza muestral.

Sean X_1, X_2, \dots, X_n y Y_1, Y_2, \dots, Y_n los datos simulados para variables aleatorias X, Y . Entonces,

$$r_{xy} = S_{xy} / S_x \cdot S_y$$

es la correlación entre X y Y .

$$S_{xy} = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X}) (Y_i - \bar{Y})$$

es la covarianza de muestra entre X y Y .

Las rutinas en FORTRAN de cálculo de estas estadísticas aparecen en el apéndice IV.

Referencias

4.1 RAND CORPORATION. A MILLION RANDOM DIGITS WITH 100,000 NORMAL DEVIATES
NEW YORK: FREE PRESS, 1955.

4.2 Knuth D.E. THE ART OF COMPUTER PROGRAMMING, VOL. 2: SEMI-NUMERICAL ALGORITHMS. READING, MASS: ADDISON-WESLEY

4.3 LEHMER D.H. PROCEEDING OF THE
SECOND SYMPOSIUM ON LARGE SCALE DIGITAL
COMPUTING MACHINERY.
CAMBRIDGE, MASS.: HARVARD UNIVERSITY
PRESS, 1951.

4.4 RUBINSTEIN Y. REUVEN
SIMULATION AND THE MONTE CARLO METHOD
JOHN WILEY & SONS 1981

4.5 FELLER, W., AN INTRODUCTION TO
PROBABILITY THEORY AND ITS APPLICATIONS,
WILEY, NEW YORK 1950.

CAPITULO V

Introducción

Los sistemas discretos fueron caracterizados en el capítulo I como sistemas en los que los cambios son predominantemente discontinuos, el término evento se usó para describir la ocurrencia de un cambio en un punto del tiempo. Un evento puede causar un cambio en el valor de algunos atributos de una entidad, puede crear o destruir una entidad o empezar o terminar una actividad.

Dado un modelo matemático dinámico de un sistema discreto, la simulación de sistemas discretos se usa para resolver el modelo. Una técnica que se puede usar en este sentido es seguir los cambios en el modelo del sistema, entonces la tarea de simulación de sistemas discretos requiere de la construcción de un programa en el cual sea posible seguir la secuencia de eventos. En las últimas secciones de este capítulo se discuten algunos de los problemas que se encuentran al hacer la programación de una simulación de este tipo.

El estado de un sistema - continuo o discreto - se expresa casi siempre como una función del tiempo. Dos tipos de "tiempos" se manejan en una simulación: el tiempo de simulación y el tiempo de corrido. El término "tiempo de simulación" se usa para referirnos al período de tiempo que se simulará - 30 minutos, 5 años, 10 años, esto es, el período deseado de investigación. El "tiempo de corrido" es el tiempo que toma el computador para simular el período de interés. Por lo general existe poca correlación entre estos dos tiempos o medidas. El tiempo de corrido es influido por factores tales como

la complejidad del modelo de simulación o el número de eventos simulados, el tiempo de corrido es por lo general menor que el tiempo de simulación. Por ejemplo, podemos simular un sistema de tráfico por un período de años en sólo unos minutos de tiempo de corrido. Sin embargo en algunos casos ocurre lo contrario. Por ejemplo, la simulación del funcionamiento de un sistema operativo podría llevar algunos minutos mientras que en el funcionamiento real de un sistema operativo las funciones de éste son llevadas a cabo en tiempos medidos en microsegundos.

En la siguiente sección se hace una descripción de los métodos usados para rastrear los cambios que ocurren en un modelo matemático dinámico, suponiendo que se habla únicamente de sistemas discretos.

5.1 Métodos que manejan el tiempo

Recuerdese del capítulo I, que dentro de la clasificación que se hace de los modelos matemáticos se tiene una segunda distinción entre modelos estáticos y modelos dinámicos (dependen del tiempo). Los modelos dinámicos permiten cambios en los valores de los atributos, derivados estos cambios como función del tiempo. La manera de seguir estos cambios es el centro de discusión de la sección siguiente.

5.1.1 Método Rastreo periódico

Una de las maneras más comunes para el manejo del tiempo con el fin de seguir la huella o con-

controlar la ocurrencia de eventos se conoce como técnica del rastreo periódico. El método consiste en ir incrementando de manera constante el reloj de la simulación y chequear la ocurrencia de algún evento en el intervalo formado al incrementar el reloj, en caso de que ocurra algún evento, se simula; en caso contrario se vuelve a incrementar el reloj sin simular evento alguno. Un ejemplo de este método se ilustra en la figura 5.1.

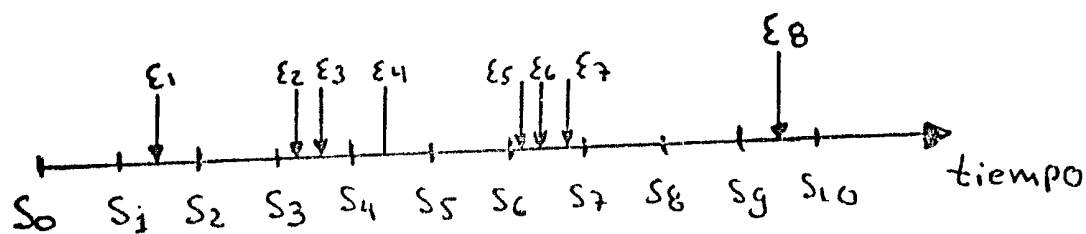


Figura 5.1 Método del rastreo periódico

En la figura 5.1 no ocurre evento alguno en la primer unidad de tiempo simulada (desde S_0 hasta S_1), así el reloj es incrementado (hasta S_2). El evento E_2 ocurre en el segundo intervalo de tiempo. Se simula este evento y se incrementa nuevamente el reloj. Ningun evento se detecta durante el tercer período. En el cuarto se simulan E_2 y E_3 , en el quinto se simula E_4 . Así continúa el proceso hasta que se simula E_8 .

Para ayudar en la comprensión de este método y al mismo tiempo notar algunas desventajas consideraremos un ejemplo. Pensemos en la simulación del funcionamiento de un centro de servicio con una cola de espera simple, durante un período de 10 minutos. Supóngase que el sistema está vacío al inicio

y al final de este período. Supóngase también que se atienden cuatro clientes, los arribos de los clientes ocurren en los tiempos 1.8, 3.2, 6.1 y 7.4 y la terminación de la atención ocurre en los tiempos 2.6, 4.8, 7.3 y 8.1. Si el incremento de tiempo usado es de un minuto, los arribos denotados por A_1, A_2, A_3 y A_4 y las terminaciones por C_1, C_2, C_3 y C_4 . El sistema se ilustra en la figura 5.2

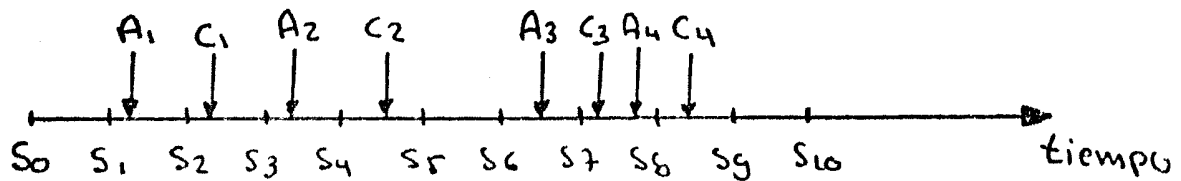


Figura 5.2 Arribo y Terminación (Eventos).

En este ejemplo C_3 y A_4 ocurren en un mismo intervalo, se tiene que determinar que evento se simulará primero, esto es una desventaja del método. Otra consideración para usar el método de rastreo periódico es la determinación del incremento del tiempo. En el ejemplo resultó ser muy grande pues C_3 y A_4 están en el mismo intervalo. Se debe escoger un incremento lo suficientemente pequeño de tal manera que la probabilidad de que ocurran más de un evento en un intervalo, sea pequeña. En el ejemplo se logra la separación de C_3 y A_4 si el incremento fuera de una décima de segundo. Es evidente que un incremento pequeño nos lleva a perder mucho tiempo al cuestionar sobre la existencia de eventos en algún intervalo (están vacíos). Otra desventaja de este

método es que para algunos modelos hay períodos de gran actividad así como períodos de poca actividad y el método no detecta esta conducta y aplica igual atención a todos los períodos. Una buena alternativa sería conocer los puntos de cambio del modelo y manipular el reloj con incrementos de tiempo que nos lleven de un cambio a otro, esta alternativa es discutida en seguida

5.1.2 Método rastreo de eventos

En este método el reloj se incrementa en la cantidad necesaria para localizar la ocurrencia del siguiente evento, no es un incremento fijo. Esta técnica requiere de algún esquema para determinar cuando ocurren los eventos. Cuando los eventos se localizan o generan, se insertan en una cola ordenada con respecto al tiempo. La magnitud del incremento es dada al rastrear esta lista de eventos ordenada y localizar el siguiente evento. Entonces el reloj es incrementado y el evento es simulado.

Nuevamente consideremos el modelo de simulación de un centro de servicio con cola de espera. El reloj de la simulación se pone inicialmente en 1.8 y el evento "A₁" es el que se simula. El reloj avanza hasta 2.6 y entonces se simula C₁. El proceso continúa hasta simular C₄, en 8.1.

Con el método de rastrear eventos, desaparecen algunos problemas que tiene el método de rastreo periódico. La principal ventaja que tiene el rastreo de los eventos sobre el rastreo periódico es que el reloj de la simulación se ve incrementado en la cantidad exacta

de tiempo para pasar de un evento a otro aunque para lograr esto se deba tener la estructura de cola ordenada de eventos.

5.2 Generación de entidades

En el capítulo I se define un sistema como una colección de objetos con un conjunto de interacciones bien definidas entre estos. A estos objetos los llamamos entidades. Es necesario tener alguna representación de las entidades al momento de una simulación.

Una entidad de un modelo de simulación debe ser identificada de manera única en el programa de simulación; en muchos casos esto se hace por la simple enumeración de las entidades. Por ejemplo en el modelo del centro de servicio podemos enumerar los arribos de clientes al sistema. Junto con la enumeración, las entidades son caracterizadas por un conjunto de atributos. El estado de una entidad en cualquier punto de tiempo es el conjunto corriente de valores de sus atributos. Las entidades que están presentes durante toda la simulación se llaman permanentes mientras que las que son introducidas por un tiempo se llaman transitorias.

Las entidades permanentes se dan generalmente como datos de entrada al modelo de simulación y son dadas durante la iniciación del modelo. Las entidades transitorias, por otro lado, se introducen en puntos particulares en el curso de la simulación y luego son removidas del modelo. Así, una generación

de una entidad transitoria envuelve la generacion de un arribo en un punto del tiempo asi como la asignacion de valores a varios atributos.

El arribo de una entidad transitoria al sistema es un evento. Este arribo altera el estado del sistema ya que aumenta el número de entidades. Los arribos pueden ocurrir a intervalos regulares de tiempo o bien el tiempo entre arribos puede obedecer al muestreo de una distribución de probabilidades específica. El tiempo de arribo para la primer entidad se genera por muestrear la distribución dada y poner entonces el evento en la lista ordenada. Cuando el reloj de la simulación avanza al punto en el cual ocurre el evento, se simula el evento y el tiempo de arribo para la segunda entidad se genera. Una alternativa a este esquema es generar todos los arribos de una sola vez. Este esquema requiere almacenar todos los arribos a la vez y aumenta el tamaño de memoria que ocupa el modelo.

Todo lo anterior visto en el sistema de comunicaciones del capítulo I seria como sigue. Las entidades transitorias en este sistema son los mensajes. Si el número de arribos es distribuido Poisson, entonces, el tiempo entre arribos es exponencial. Una vez que se determina la distribución del tiempo entre arribos, los valores de los atributos deberan asignarse. Los valores de los atributos pueden ser generados ya sea en el momento en que el tiempo del arribo es generado o al momento en

que ocurre el arribo. El factor que determina cual de los dos momentos escoger es la dependencia de estos valores con respecto al estado del sistema. Valores de atributos que dependen del estado del sistema se asignan al momento de la ocurrencia del arribo, como sucede con los tiempos entre arribos, los valores de los atributos por lo general seran obtenidos de muestrear alguna distribución dada.

Continuando con el ejemplo descrito anteriormente, una vez que han sido determinados los tiempos entre arribos para entidades, por muestrear la distribución exponencial, los valores de los atributos se deben asignar. Dos atributos de interés de las entidades (los mensajes) son la prioridad y la longitud del mensaje. Si la longitud del mensaje es distribuida normal con media μ y varianza σ^2 y la prioridad tiene la siguiente distribución; mensajes normales con probabilidad $1/2$, mensajes de mediana urgencia con probabilidad $3/8$ y urgentes con probabilidad $1/8$. La asignación de la longitud del mensaje se obtiene de muestrear la normal correspondiente mientras que la asignación de prioridad se puede hacer muestreando la distribución uniforme en el $(0,1)$ y si el valor de la muestra está en el intervalo $(0, 1/2)$ se le asigna prioridad normal, si está en $(1/2, 7/8)$ se le da la prioridad de mediana urgencia y si está en $(7/8, 1)$ se le asigna prioridad urgente. En el capítulo VI se aplicarán estos métodos en algunas simulaciones.

5.3 Eventos y su sincronización

Un evento es la ocurrencia de un cambio en el estado del sistema en algún punto del tiempo. En este sentido un evento no tiene duración, pero existe en un instante del tiempo. Un aspecto de importancia en la simulación de sistemas discretos es el manejo y sincronización de los eventos. Estos pueden clasificarse en dos tipos; los eventos del sistema y los del programa. Un evento del sistema es aquel que representa la simulación de algo en el sistema real. Mientras que un evento de programa no tiene que ver con el sistema real sino sólo en el programa de la simulación. Detalles de programa tales como obtener estadísticas de la simulación son eventos de programa. Aún cuando estos eventos son ajenos al sistema real necesitan un manejo igual que los eventos de sistema.

La secuencia de los eventos y su sincronización son aspectos importantes en la simulación de sistemas discretos. Los eventos son representados como instantes en el tiempo. Una vez que se maneja un evento y avanza el reloj de la simulación, se invoca una rutina que se encarga de la ejecución del evento particular, esta rutina simula el efecto que tiene el evento en el sistema real. Si la secuencia de eventos está separada por algún tiempo entonces no habrá problema en su ejecución, si por el contrario dos o más eventos ocurren simultáneamente, se debe decidir en base al sistema real cual evento se ejecuta primero.

Un ordenamiento adecuado de los eventos es importante ya que los efectos de ciertos eventos pueden tener consecuencias sobre otros eventos. Por ejemplo algunos eventos hacen que se manejen otros, mientras que otros eventos pueden causar la cancelación de algunos eventos ya manejados. Los eventos sin un orden conveniente tendrían efectos drásticos sobre la conducta del sistema.

Los eventos se manejan en la mayoría de los programas de simulación de sistemas discretos por un "CALENDARIO" de eventos. Los eventos generados por el sistema se unen para formar una lista ordenada con respecto al tiempo. Si se usa el método de rastreo periódico entonces el calendario de eventos es rastreado para identificar los eventos que ocurren en el período de rastreo y así invocar a las rutinas de los eventos y después continuar con la simulación. Si por el contrario se usa el rastreo de eventos el reloj de la simulación se controla por el calendario de eventos. Aquí el calendario de eventos se consulta para observar el próximo evento a ocurrir y así invocar la rutina que ejecuta dicho evento. El reloj de la simulación se adelanta hasta donde indica el calendario de eventos. Después que se ejecuta la rutina se repite el proceso

El número de eventos y la complejidad de la correspondiente rutina determina el tiempo de corrido y en consecuencia el costo de la simulación.

5.4 Manejo de colas y listas

Las listas de espera o colas las encontramos en muchos de los sistemas que se desean modelar ya que estos hacen uso de algún recurso limitado, es el caso por ejemplo de un sistema de cómputo donde los recursos son compartidos. Entonces las colas aparecen al competir los usuarios por estos recursos limitados. Es importante entonces un buen manejo de colas para el desarrollo de la simulación. Una de las disciplinas de manejo de colas más comunes es la disciplina "FIFO" (FIRST-IN-FIRST-OUT), en la cual el usuario que tiene más tiempo esperando el servicio es el que se atiende. Otras disciplinas incluyen la conocida como "LIFO" (LAST-IN-FIRST-OUT) en la cual el usuario que tiene menos tiempo esperando es el atendido; la disciplina PRI (DE PRIORIDAD) en la cual el usuario es seleccionado en base a un valor particular asociado con un atributo.

Cuando es generado un arribo de una entidad y es manejado se asignan valores de sus atributos, esto se hace ya sea por muestreo de alguna distribución dada o a través de algún otro procedimiento preestablecido. En muchos casos la entidad es enumerada para ser identificada. Una manera conveniente para representar una entidad es por medio de un registro como el que se muestra en la figura 5.3. Se tiene en el registro la identificación de la entidad y el valor de sus atributos.

# de entidad	tiempo de arribo	Atributo 1	...	Atributo n
--------------	------------------	------------	-----	------------

Figura 5.3 Registro para una entidad.

Así como un registro es conveniente para representar entidades que arriban al sistema, una manera conveniente de representar una cola es por medio de una lista. La organización de las colas como listas deberá soportar operaciones tales como insertar un registro (aumentar una entidad a la cola), borrar un registro (eliminar una entidad) y acceder un registro particular de la lista para examinar el valor de algún atributo. La manera como se maneja la lista depende mucho de la estructura de ésta.

Existen un buen número de consideraciones en la construcción de listas. Primero, la dimensión de la lista debe ser definida. Una lista con entradas simples es una lista unidimensional; una lista de sublistas es una lista bidimensional y así sucesivamente. Segundo, la densidad de la lista debe considerarse. Las entidades de la lista pueden estar almacenadas en localidades de memoria contiguas o separadas. Tercero, el sentido de la lista también se debe considerar. Por ejemplo una lista unidimensional podría tener sentido de arriba hacia abajo o viceversa.

Existen otros factores que influyen en la

construcción de una lista. Entre otros, uno de importancia es la técnica que se usa para realizar búsquedas sobre una lista. Las entidades pueden ser localizadas de varias formas, dependiendo esto de la construcción de la lista. Las entidades pueden ser localizadas por un rastreo secuencial de la lista hasta encontrar el registro deseado o pueden ser localizadas directamente por aplicar la técnica de hash. Una tercer técnica es por medio de apuntadores, esta técnica que es la más usada consiste en añadir una liga en el registro de la entidad que apunta al siguiente registro en la lista. Si se desea tener más apuntadores se aumentan en el registro de la entidad. La principal ventaja de las listas ligadas es que esta puede ser alterada con facilidad. Las cancelaciones o aumento de entidades pueden hacerse con un manejo sencillo de los apuntadores. Un ejemplo de una lista ligada simple es dado en la figura 5.4.

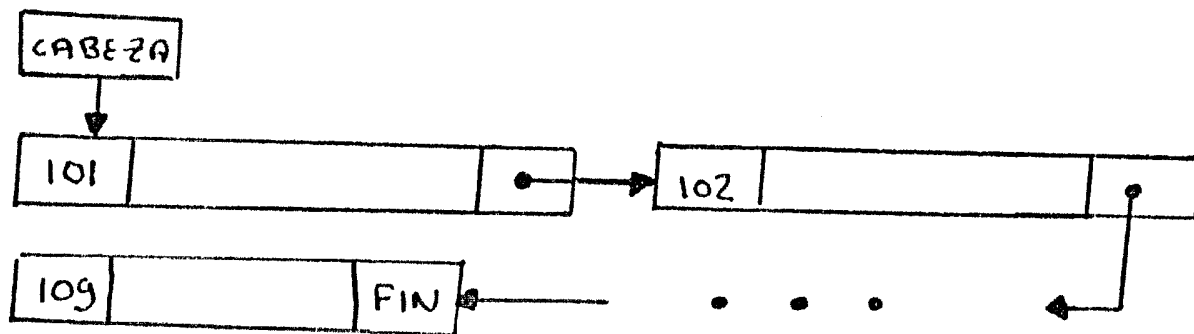


Figura 5.4 Lista ligada simple

Una lista doblemente ligada se obtiene de la lista anterior si se aumenta un apuntador a cada registro y el registro 109 apunta al 108, este al 107 y así hasta que el 102 apunta al 101. Una

lista circular se obtiene de la lista simple si el registro 109 apunta a la cabeza. La información necesaria para un buen manejo de estas estructuras se encuentra en [5.1].

Supongamos que el registro 102 se desea borrar de la lista. Esto se logra si el registro 101 se le cambia el apuntador hasta donde apunta el registro 102 es decir hasta 103. El registro 102 será reclamado desde alguna rutina recolectora de espacio desocupado. Supongamos que después de borrar el registro 102 deseamos añadir un registro 102a entre 101 y 103. Esto se logra apuntando desde el registro 101 al registro 102a y desde este se apunta a donde apuntaba 101 esto es al 103. Nótese que este esquema no requiere que el registro 102a se localice en lugar contiguo al registro 101; el registro 102a puede estar en algún otro lugar y ser localizado a través del apuntador del registro 101. Esto permite manejar listas de registros que no están necesariamente en un arreglo contiguo.

Muchas de las disciplinas de manejo de colas son de fácil implementación haciendo uso de listas ligadas. Para simular la operación de una cola "FIFO" es suficiente con una lista ligada simple. Los arribos de entidades son añadidos al final de la lista. Una cola "LIFO" también es simulada con una lista ligada simple. Aquí los arribos se hacen por la cabeza de la lista. Un ejemplo ayudará a la comprensión de estos conceptos.

Consideremos un sistema con cola de un centro de servicio. Al arribar los usuarios (entidades) se les asigna un valor numérico creciente, se les asocia el tiempo de arribo y el tiempo de servicio requerido. Supongamos que hay cinco entidades en la cola esperando servicio, con los atributos siguientes:

Número de entidad	101	102	103	104	105
Tiempo de arribo	1013	1009	1011	1015	1014
Tiempo demandado	2	4	6	1	5

Estas entidades pueden ser representadas por un registro como el que se muestra en la figura 5.5.

Número de usuario	Tiempo de arribos	tiempo requerido	Apuntador
-------------------	-------------------	------------------	-----------

Figura 5.5 Registro para usuarios

Si se usa la disciplina "FIFO" se usa el siguiente arreglo.

CABEZA
102

101	1013	2	105
102	1009	4	103
103	1011	6	101
104	1015	1	
105	1014	5	104

FIN
104

El proceso inicia con el registro 102 y continua por donde indican los apuntadores procesando el

103, 101, 105 y 104; cada registro debe ser borrado al ser procesado. Cuando un nuevo arribo ocurre (registro 106) se añade a la lista simplemente apuntando a este desde el último (registro 104).

Si se usa la disciplina "LIFO", entonces, el arreglo usado es:

CABEZA: 104

101	1013	2	103
102	1009	4	
103	1011	6	102
104	1015	1	105
105	1014	5	101

Aquí el proceso comienza con el registro 104, continúa con la secuencia: 105, 101, 103 y 102. Si llega una nueva entidad (registro 106) se suma a la cola por la cabeza es decir se apunta desde el registro 106 al 104 y la cabeza al 106.

Un esquema muy común de prioridad que se usa al trabajar con demanda de centros de servicio es la disciplina del tiempo requerido más corto. Aquí la prioridad más alta se le asigna al usuario que demanda menos tiempo. Si usamos esta disciplina el arreglo usado es como el mostrado en seguida.

CABEZA: 104

101	1013	2	102
102	1009	4	105
103	1011	6	
104	1015	1	101
105	1014	5	103

Entonces el proceso sigue la siguiente secuencia: 104, 101, 102, 105 y 103. Si ocurre un nuevo arribo hay que checar la demanda e insertar en el lugar correcto, para esto es necesario un rastreo sobre la lista checando el atributo de demanda de cada entidad.

Existen soluciones analíticas [5.3] que nos ayudan a establecer el comportamiento de colas de espera para ciertas características. El primer ejemplo del capítulo VI tiene una solución analítica de un sistema con un centro de servicio y se hace el programa que simula el sistema para comparar resultados y al mismo tiempo resaltar la importancia de la simulación como alternativa de solución ya que como se verá la solución analítica no siempre es fácil.

5.5 Resultados de una simulación

Un aspecto importante en el diseño, desarrollo, y uso de un modelo de simulación, es la manera como se recolecten y resuman los datos. Una de las principales ventajas de usar un lenguaje de programación con orientación a la simulación en vez de un lenguaje de propósito general es que los lenguajes orientados facilitan la recolección de datos. Dependiendo del sistema que se simule, diferente tipo de información acerca del comportamiento del sistema es lo que se desea recolectar. En esta sección se describen algunos tipos de datos que más comúnmente se desea obtener de una simulación.

El tipo de dato más común son contadores sobre las entidades - por ejemplo, el número de ocurrencias de un tipo de eventos, el número de entidades en cada cola del modelo y el tiempo total que una entidad está en el sistema. Este tipo de contadores se manejan al final de la simulación para obtener medidas que resuman el comportamiento del sistema. Este tipo de datos se colectan fácilmente usando variables contadores que son manipuladas cuando ocurre un evento que afecta los datos de interés. Se discute una situación para aclarar estos puntos.

Supongamos que se desarrolla un modelo de simulación para un centro de servicio con una cola de espera simple y que una de las variables de interés es el número de arribos durante la simulación. Esta información puede obtenerse iniciando en cero una variable contador "NUMARR" con cero al comienzo de la simulación y darle un incremento cada vez que ocurra un arribo. Otro dato de interés puede ser la cantidad de tiempo total que cada entidad permanece en el sistema.

Una atención particular se le da al lugar en el modelo en el cual se colectan datos. El lugar más obvio para esta colecta es durante el procesamiento de la ocurrencia de un evento que afecta las variables de interés. Por ejemplo, en la situación presentada anteriormente, la variable "NUMARR" debe ser incrementada y a sea cuando se invoca la rutina que indica el momento de

la ocurrencia de un arribo o al momento del arribo mismo.

La naturaleza de las estadísticas que se obtengan al final de la simulación dependen del tipo de datos colectados. Por ejemplo, si deseamos la media y la desviación estándar de alguna variable, se debe calcular la suma de las observaciones de la variable, la suma de los cuadrados de las observaciones y el número de observaciones en la cual se basa la acumulación. Si, por ejemplo, deseamos calcular la media y la desviación estándar de la longitud de la cola de espera del sistema con un centro de servicio y una cola de espera; Debemos tener tres variables contadores "SUM", "SUM CUAD" y "NVEC", las tres se ponen en cero. Cuando el reloj de la simulación se incrementa, la numeración de la cola, llamada "NUMCOL", se suma a la variable "SUM", el cuadrado de "NUMCOL" es sumado a "SUM CUAD" y "NVEC" se incrementa en uno. Al final de la simulación, la media y la desviación estándar de la longitud de la cola se calculan usando las formulas:

$$\text{MEDIA} = \text{SUM} / \text{NVEC}.$$

$$\text{DESU. ST.} = \frac{[(\text{SUM CUAD})^2 - (\text{NVEC})(\text{MEDIA})^2]^{1/2}}{(\text{NVEC} - 1)}$$

Además podemos agrupar los datos por medio de frecuencias. La recolección de estos datos hace posi-

ble resumir ciertos aspectos del sistema a través del uso de histogramas de frecuencias. Para visualizar lo anterior se discute la siguiente situación: Consideremos el sistema con un centro de servicio y una cola de espera simple. Se desea recolectar datos de tal manera que se pueda analizar la distribución del tiempo entre arribos. Esto se hace como sigue. Se hace una partición de clases en intervalos de tiempo. Si el tiempo entre arribos se maneja en la variable "TIMARR" y las clases se establecen como:

CLASE 1	$0 \leq \text{TIMARR} < 1$
CLASE 2	$1 \leq \text{TIMARR} < 2$
⋮	⋮
CLASE 10	$9 \leq \text{TIMARR} \leq 10$

Un arreglo de diez entradas, llamado "ARRDIST", se pone en cero. Cada vez que se tiene un arribo, se checa "TIMARR" para determinar que entrada de "ARRDIST" se incrementa. Cuando se termina la simulación, la distribución del tiempo entre arribos es resumida en el arreglo "ARRDIST" en forma de tabla de frecuencias. Esta puede ser sometida a la prueba de bondad de ajuste discutida en el capítulo III.

El tipo de datos que se colecta en una simulación es muy variado, aquí hemos discutido sólo algunos tipos de los más comunes.

Al hacer una simulación esta se hace por lo general con el fin de investigar varios aspectos

del sistema para ciertas condiciones de operación. Los aspectos del sistema que están bajo estudio se les conoce como variables de respuesta. Los aspectos del sistema que son controlados y que generalmente varían para observar los efectos sobre las variables de respuesta son llamadas variables de control.

En muchos casos las variables de respuesta pueden tomar cierto sentido estadístico, por ejemplo, podríamos considerarlas como parámetros de alguna distribución. Así, a las variables de respuesta se les puede aplicar lo que se discutió en el capítulo III con el fin de verificar los resultados de una simulación. En el capítulo VI se incluyen sistemas simulados, con su correspondiente verificación y análisis de resultados.

5.6 Lenguajes de programación para simulación

Una vez que se tienen identificadas las entidades, atributos y actividades para una simulación, el modelo se formula en términos de un diagrama de flujo, esto es el modelo es puesto en una forma adecuada para el computador. Es decir, el modelo debe ser codificado usando algún lenguaje. El rango de los lenguajes de programación que se usan en la simulación de sistemas discretos es bastante extendido, desde lenguajes de bajo nivel, hasta lenguajes especializados en simulación tales como GPSS (General Purpose Simulation Systems). La referencia más importante acerca de GPSS es E.5.31. Una buena cantidad de factores tienen influencia en la elección del

lenguaje, incluyendo la familiaridad del programador con el lenguaje; los lenguajes que soporta la instalación donde se va hacer la simulación; la complejidad del modelo; y la necesidad de recolectar información y desplegar los resultados de la simulación. En seguida hacemos una breve discusión acerca de los requerimientos que una simulación haría a los lenguajes y un repaso de las características de estos.

Muchos modelos de simulación realizan funciones similares. Algunas de estas son:

- 1.- Generación de variables aleatorias
- 2.- Manejo del reloj de la simulación
- 3.- Manejo de rutinas, para la simulación de la ejecución de eventos.
- 4.- Manejo de colas
- 5.- Recolectar datos
- 6.- Resumir y analizar datos
- 7.- Formular e imprimir salida

Muchos de los sistemas simulados tienen cierta conducta estocástica, por ejemplo los tiempos de servicio, tiempos entre arribos y tiempos de demanda en un sistema de un centro de servicios, son aleatorios. Para simular tales eventos estocásticos se requiere de las técnicas discutidas en el capítulo IV. Así cualquier lenguaje usado en la simulación debe permitir la elaboración de estas rutinas o tener rutinas propias para generar las variables aleatorias.

En cuanto al segundo punto, ya sea que se use alguno de los métodos discutidos en la sección 5.1 (rastreo periódico o rastreo de eventos) o algún otro método, el lenguaje debe permitir un adecuado manejo del tiempo.

Cuando se hace el manejo de eventos para su ocurrencia, el programa de simulación por lo general efectúa los cambios requeridos en el estado del sistema, por invocar subrutinas hechas con este propósito.

El manejo de colas es común en muchos modelos ya que algunos sistemas tienen incluida la competencia de recursos limitados. La representación y manejo de líneas de espera se hace de manera adecuada por medio del manejo de listas. Entonces un lenguaje con un manejo eficiente de listas ofrece una ventaja significativa para la simulación.

En muchas simulaciones los datos a recolectar no ofrecen mayor dificultad, tal es el caso de simulaciones que se interesan en calcular medias, varianzas o desviaciones estándar. Sin embargo algunos modelos de simulación más sofisticados exigen de análisis estadístico más elaborado, quizás análisis de regresión, análisis de varianza o alguna herramienta estadística sofisticada. Entonces en estos casos el lenguaje de programación a escoger debe ser orientado a facilitar estos cálculos.

En resumen, cada lenguaje candidato debe pasar

por una consideración de los puntos anteriores.

En la figura 5.6 se tiene una clasificación de los lenguajes y paquetes disponibles.

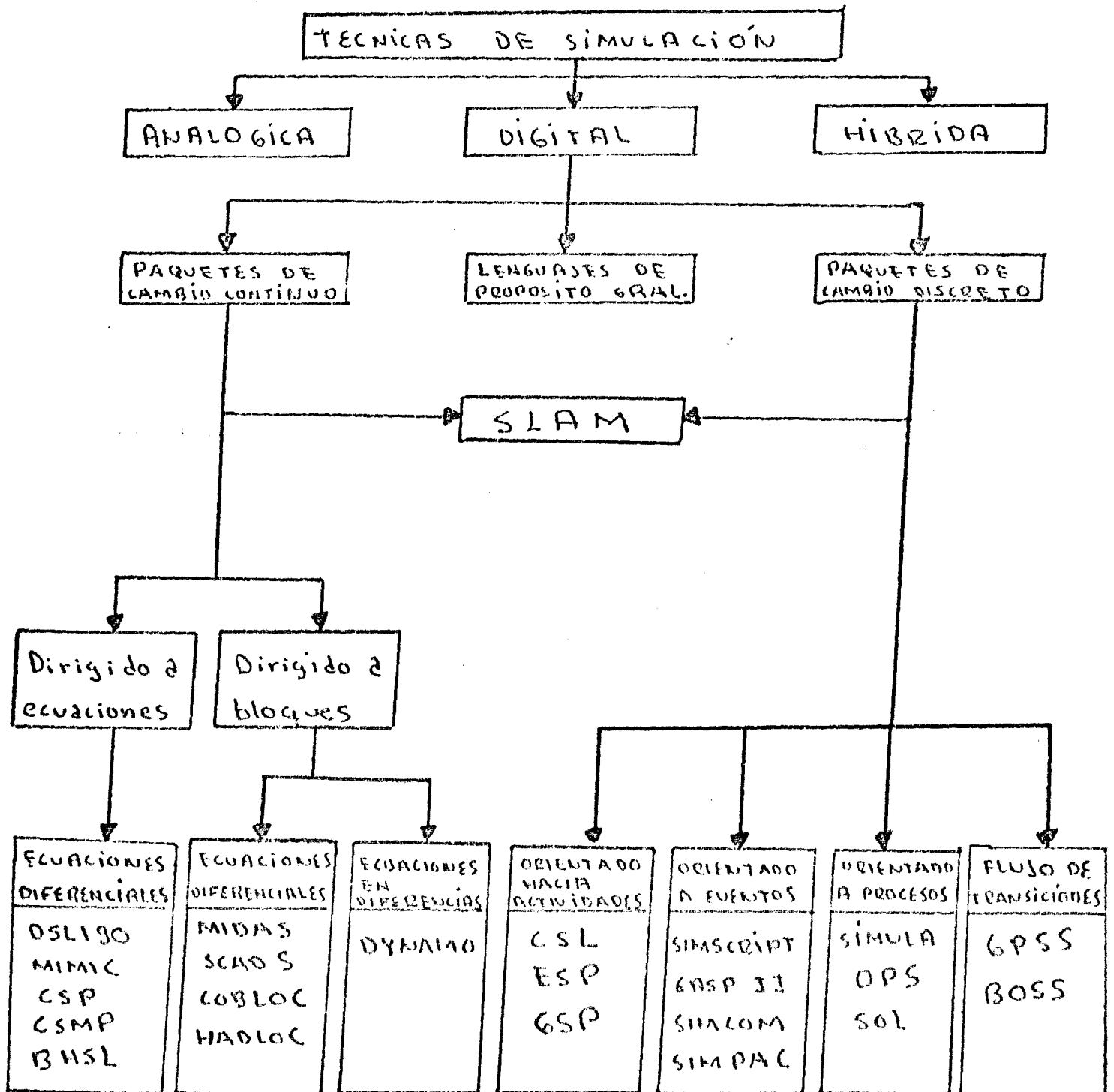


Figura 5.6 Clasificación de lenguajes

Las referencias para paquetes de cambio discreto son [5.3], [5.4], [5.5], [5.6] y [5.7].

Con el objeto de ilustrar el uso de un lenguaje de programación de propósito general (FORTRAN) en la elaboración de una simulación, en el capítulo siguiente se hace la simulación de algunos sistemas.

Referencias

- 5.1) WIRTH N. ALGORITHMS + DATA STRUCTURE = PROGRAMS. PRENTICE-HALL INC. ENGLEWOOD N.J.
- 5.2) GROSS, DONALD, AND HARRIS, CARL. FUNDAMENTAL OF QUEUEING THEORY. NEW YORK: JOHN WILEY AND SONS, 1974.
- 5.3) BOBILLIER, P. A., KAHAN, B. C. AND PROBST, A. R. SIMULATION WITH GPSS AND GPSS V. ENGLEWOOD CLIFFS, N.J: PRENTICE-HALL, 1976
- 5.4) CONSOLIDATED ANALYSIS CENTERS, INC. SIMSCRIPT II.5 REFERENCE HANDBOOK. SANTA MONICA, CA., 1971
- 5.5) PRITSKER, A. A. B. THE GASIO IV SIMULATION LANGUAGE. NEW YORK: JOHN WILEY AND SONS, 1974

CAPITULO VI

Introducción

Este capítulo tiene como finalidad, buscar solución a problemas que aparecen en sistemas operativos, a través del uso de lo desarrollado en los capítulos anteriores. La simulación puede ser vista como una alternativa a una posible solución analítica que como se ve en el primer problema, no siempre es fácil, y así, dejav a la simulación como alternativa factible.

Los problemas escogidos son:

- *) Modelo de un centro de servicio con cola simple manejada "FIFO"
- *) Modelo de un centro de servicio (un "CPU") con disciplina "ROUND ROBIN".
- *) Modelo de memoria virtual.

Cada problema trae consigo:

- *) Descripción del problema
- *) Solución analítica (solo el primero)
- *) Estructura y desarrollo del programa que simula.
- *) Resultados y su interpretación.

6.1 Modelo de un centro de servicio.

Al diseñar sistemas operativos, frecuentemente se esta interesado en conocer el comportamiento futuro de un centro de servicio compartido (CPU ó canal de I/O)

Debemos caracterizar las siguientes variables de entrada:

- i) tiempo entre arribos
- ii) tiempo demandado de servicio
- iii) disciplina de despacho de la cola

Con el objeto de acompañar esta simulación de una solución analítica se supone lo siguiente.

- i) la cola de servicio se despacha por "FIFO".
- ii) tiempos de arribo y demanda se distribuyen exponencial con parámetros $1/\lambda$ y $1/\mu$ respectivamente.
- iii) no hay restricción en la capacidad de la cola.

La figura 6.1 ilustra este sistema.

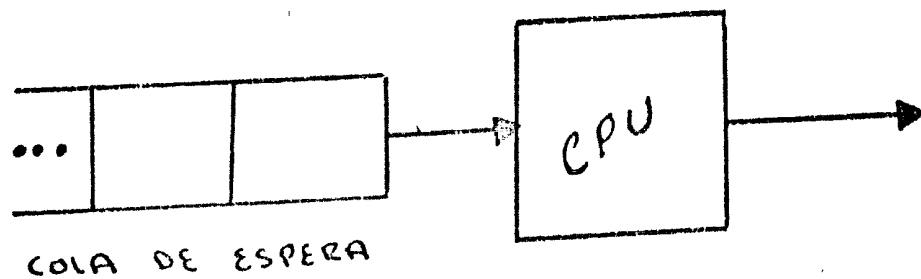


Figura 6.1 Sistema de un centro de Servicio

Los arribos o usuarios llegan al sistema a hacer cola (despacho "FIFO") hasta que se les concede el "CPU" y se les atiende totalmente para salir del sistema. Cada arribo demandará un tiempo de "CPU" aleatorio distribuido exponencial con parámetro $1/\mu$. Los arribos se distribuyen también exponencial con

parámetro λ .

Antes de simular para ver el comportamiento de este sistema, damos la solución analítica.

Es de interés en cualquier sistema con cola, el número de clientes o arribos en el sistema. Sea " S_j " el estado del sistema cuando hay " j " clientes. Sea $P_j(t)$ la probabilidad de estar en el estado " j " al tiempo t . Ahora bien, el sistema se encuentra en el estado " S_j " al tiempo " $t + \Delta t$ " si y sólo si uno de los eventos siguientes excluyentes ocurre.

*) El sistema estaba en " S_{j-1} " al tiempo t y ocurre un arribo únicamente en $(t, t + \Delta t)$ y además no se atiende a ningún cliente.

*) El sistema está en " S_j " al tiempo t y no ocurre arribo ni despacho en $(t, t + \Delta t)$.

*) El sistema está en S_{j+1} en el tiempo t y no hay arribo y sí se despacha un cliente.

Después de un poco de teoría sobre la distribución exponencial [6.1] se tiene que la probabilidad de un arribo en $[t, t + \Delta t]$ es

$$(1) \quad \lambda \Delta t + O(\Delta t)$$

donde $O(\Delta t)$ es tal que $\lim_{\Delta t \rightarrow 0} \left[\frac{O(\Delta t)}{\Delta t} \right] = 0$

Mientras que la probabilidad de un despacho en $[t, t+\Delta t]$ es:

$$(2) \mu \Delta t + O(\Delta t)$$

Entonces si despreciamos la probabilidad de más de un arribo o despacho en $[t, t+\Delta t]$ tenemos.

$$\begin{aligned}
 P_j(t+\Delta t) = & P_{j-1}(t) (\lambda \Delta t) (1 - \mu \Delta t) \\
 (3a) \quad & + P_j(t) (1 - \lambda \Delta t) (1 - \mu \Delta t) \\
 & + P_{j+1}(t) (1 - \lambda \Delta t) (\mu \Delta t), \quad j=1,2,\dots
 \end{aligned}$$

Simplificando y reordenando

$$\begin{aligned}
 (3b) \quad \lambda P_{j-1}(t) - (\lambda + \mu) P_j(t) + \mu P_{j+1}(t) = & \frac{P_j(t+\Delta t) - P_j(t)}{\Delta t} \\
 j = 1, 2, \dots
 \end{aligned}$$

Si $\Delta t \rightarrow 0$ tenemos

$$\begin{aligned}
 (3c) \quad P_j'(t) = & \lambda P_{j-1}(t) - (\lambda + \mu) P_j(t) + \mu P_{j+1}(t) \\
 j = 1, 2, \dots
 \end{aligned}$$

El caso en que $j=0$ debe tratarse por separado ya que S_{-1} no existe. Con el mismo procedimiento se obtiene

$$(4) \quad P_0'(t) = -\lambda P_0(t) + \mu P_1(t)$$

Estas ecuaciones pueden verse como ecuaciones diferenciales en diferencias, cuya solución nos da la distribución de probabilidades del número de clientes en el sistema

$$(5) \quad \begin{aligned} P_0'(t) &= -\lambda P_0(t) + \mu P_1(t) \\ P_j'(t) &= \lambda P_{j-1}(t) - (\lambda + \mu) P_j(t) + \mu P_{j+1}(t) \\ & \quad j > 1 \end{aligned}$$

En el momento que el sistema se estabilize, la probabilidad " P_j " toma un valor constante y P_j' es cero por lo tanto lo anterior se puede simplificar a ecuaciones en diferencias simples.

$$(6) \quad P_1 = \frac{\lambda}{\mu} P_0; \quad P_{j+1} = \frac{\lambda + \mu}{\mu} P_j - \frac{\lambda}{\mu} P_{j-1} \quad j > 1$$

Buscamos ahora solución a este nuevo sistema de ecuaciones en diferencias. tenemos que

$$(7a) \quad P_1 = \frac{\lambda}{\mu} P_0$$

$$(7b) \quad P_2 = \frac{\lambda + \mu}{\mu} P_1 - \frac{\lambda}{\mu} P_0 = \left(\frac{\lambda}{\mu}\right)^2 P_0$$

y en general

$$(7c) \quad P_j = \left(\frac{\lambda}{\mu}\right)^j P_0 \quad j=1, 2, \dots$$

$$\text{Como } \sum_{j=0}^{\infty} P_j = \sum_{j=0}^{\infty} \left(\frac{\lambda}{\mu}\right)^j P_0 = 1 \quad \text{y } \frac{\lambda}{\mu} \in [0, 1]$$

y si $\mu > \lambda$ (mas despachos que arribos)

la serie $\sum_{j=0}^{\infty} (\frac{\lambda}{\mu})^j$ converge y en este caso

$$(7f) \quad P_j = \left(\frac{\lambda}{\mu}\right)^j \left(1 - \frac{\lambda}{\mu}\right) \quad j=0,1,2,\dots$$

El caso en que $\lambda \geq \mu$ (mas arribos que despachos) no tiene esta solución analítica.

Sea X que denota la variable aleatoria que cuenta el número de clientes en el sistema, entonces

$$(8) \quad P(X=x) = P_x = \left(1 - \frac{\lambda}{\mu}\right) \left(\frac{\lambda}{\mu}\right)^x \quad x=0,1,\dots$$

Entonces

$$E(X) = \sum_{x=0}^{\infty} x P_x = \left(1 - \frac{\lambda}{\mu}\right) \sum_{x=0}^{\infty} x \left(\frac{\lambda}{\mu}\right)^x$$

Despues de desarrollar la sumatoria tenemos

$$(9) \quad E(X) = \left(1 - \frac{\lambda}{\mu}\right) \left(\frac{\lambda/\mu}{(1 - \lambda/\mu)^2}\right) = \frac{\lambda/\mu}{1 - \lambda/\mu}$$

Es el número esperado de clientes en el sistema.

Ahora calculamos el número esperado de clientes en

la cola.

Sea Q la variable aleatoria que mide la longitud de la cola, entonces

$$\begin{aligned}
 (10) \quad E(Q) &= 0 + \sum_{j=1}^{\infty} (j-1) p_j = \sum_{j=1}^{\infty} j p_j - \sum_{j=1}^{\infty} p_j = \\
 &= E(x) - \sum_{j=1}^{\infty} p_j = E(x) - (1 - p_0) = \\
 &= E(x) - (1 - (1 - \frac{\lambda}{\mu})) = \frac{(\lambda/\mu)^2}{1 - (\lambda/\mu)}
 \end{aligned}$$

$$E(Q) = \frac{(\lambda/\mu)^2}{1 - (\lambda/\mu)}$$

Otra medida importante es el tiempo que un cliente esta en el sistema. Sea Y la variable aleatoria que mide el tiempo de espera en el sistema. Sea W su esperanza. Usando la formula de Little [6.2] se relaciona $E(x)$ con $E(Y)$. A saber

$$(11) \quad E(x) = \lambda E(Y) \quad \text{de donde}$$

$$(12) \quad E(Y) = \frac{1}{\mu - \lambda}$$

Ademas. Si Z es la variable aleatoria del tiempo de espera en la cola, tenemos

$$(13) \quad E(Z) = E(Y) - \frac{1}{\mu} = \frac{\lambda}{\mu(\mu - \lambda)}$$

Podemos considerar las formulas 7f, 9, 10, 12 y 13 como la solución analítica del sistema. En seguida se da la solución por simulación y se comparan resultados.

Notese que a pesar de lo simple del problema, la solución analítica tiene cierta complejidad, y se espera que en problemas más complejos, la solución analítica sea casi imposible y dejar a la simulación como única alternativa factible.

En la solución por simulación suponemos que:

- * La distribución del tiempo entre arribos es exponencial con media de arribos λ .
- * La distribución del tiempo de servicio es exponencial con media de despacho μ .
- * Se despacha "FIFO"
- * El manejo de eventos se hace por medio de rastreo periódico (ver sección 5.3)

El diagrama de flujo usado es mostrado en la figura 6.2.

Las estadísticas de interés son, el tiempo promedio de espera en la cola, el promedio de la longitud de cola, el promedio del

número de clientes en el sistema y el tiempo total de ocupación del sistema.

Después del diagrama de flujo se da el programa encargado de la simulación el cual nos da también solución analítica en base a lo obtenido en la primer sección.

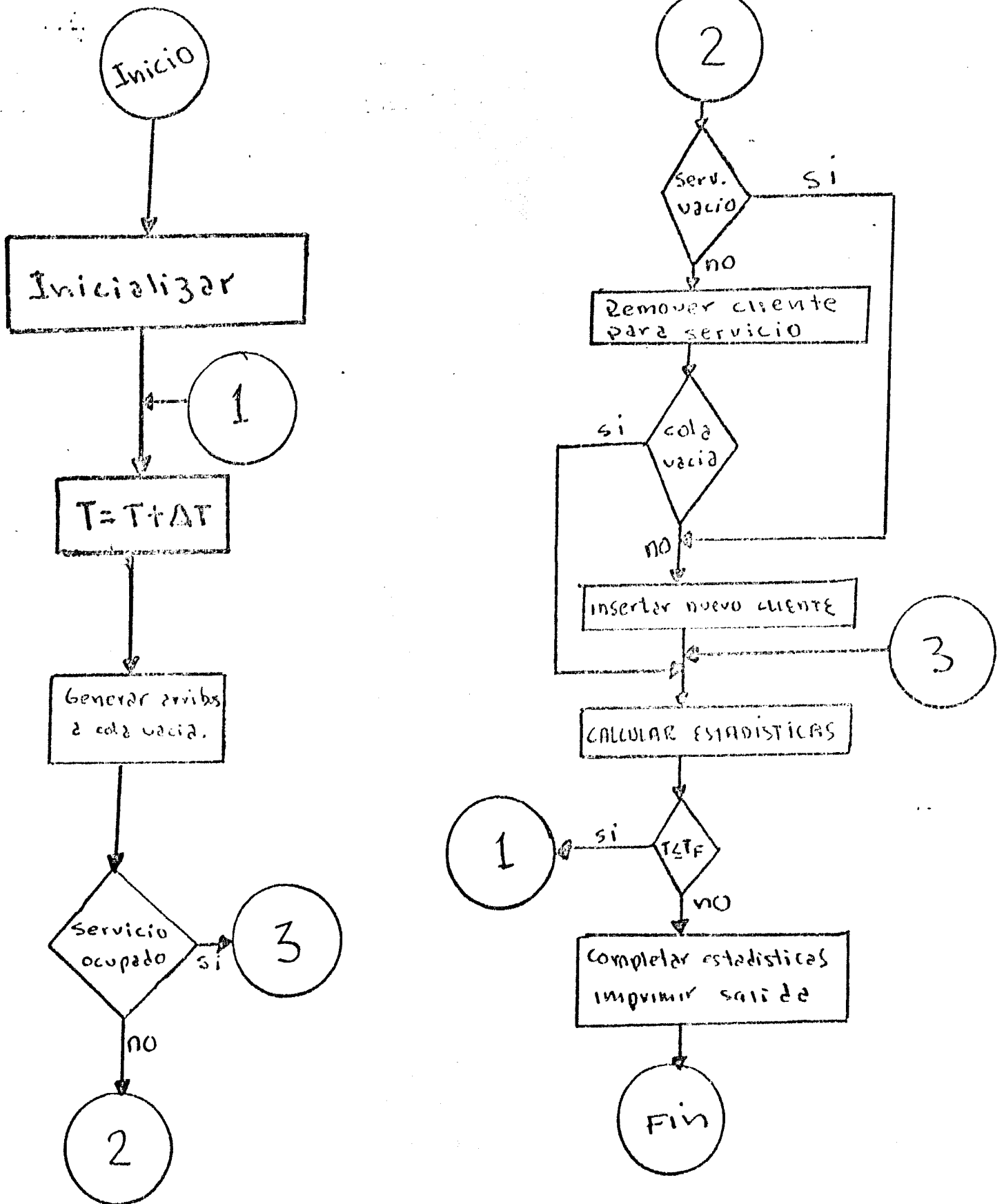


Figura 6.2 Diagrama de flujo del sistema de un centro de servicio.

C
C ESTE PROGRAMA SIMULA LA OPERACION DE UN SISTEMA CON UN CENTRO
C DE SERVICIO CON COLA SE ESPERA SIMPLE, LOS ARRIBOS A ESTA COLA SON
C EXPONENCIALES CON PARAMETRO (1/LAMBDA), LOS TIEMPOS DE SERVICIO SON
C TAMBIEN EXPONENCIALES CON PARAMETRO (1/MU).
C SE HACE UNA COMPARACION CON LA SOLUCION ANALITICA OBTENIDA EN ESTE TRA-
C BAJO.

C
C
C LOS DATOS DE ENTRADA SON:

C DELT,- INCREMENTO DEL TIEMPO
C SLEN,- LONGITUD DE LA SIMULACION
C LAMBDA,- TAZA PROMEDIO DE ARRIBOS
C MU,- TAZA PROMEDIO DE SERVICIO
C

C
C INTEGER QUEUE
C LOGICAL SFULL
C REAL LAMBDA,MU

C
C COMMON NSEED,AMBDA1,RMU1

C
C
C INICIALIZACION

C T=0.
C TIDLE=0.
C AVEQ=0.
C AVES=0.
C NSER=0.
C WTIME=0.
C ARTIM=0.
C TSER=0.
C QUEUE=0.
C SFULL= .FALSE.
C NSEED=5555
C WRITE(5,55)
55 FORMAT(' DAR INCREMENTO,TIEMPO DE SIMULACION,TAZA DE ARRIBO,','/,',
1 'Y TAZA DE SERVICIO')
C READ(5,500) DELT,SLEN,LAMBDA,MU
500 FORMAT(4F10.5)
C AMBDA1=LAMBDA
C RMU1=MU
C LAMBDA=1./LAMBDA
C MU=1./MU

C
C
C SUBROUTINA DE SIMULACION

C
C GENERA PRIMER ARRIBO
C
C CALL EXPON(NSEED,ARTIM,LAMBDA)
C
C CHECA SI HA OCURRIDO ARRIBO
C

```

1     IF (ARRTIM .LE. 0. ) CALL ARRVL(ARRTIM,QUEUE,LAMBDA)
C
C     CHECA SI EXPIRO EL TIEMPO DE SERVICIO
C
        IF (TSER .LE. 0. ) CALL SERCOM(TSER,QUEUE,NSER,MU,SFULL)
        IF (QUEUE .NE. 0) GOTO 40
        IF (SFULL) GOTO 40
        TIDLE=TIDLE+DELT
40    AVEQ=AVEQ+QUEUE
        IF (SFULL) AVES= AVES+1
        AVES=AVES+QUEUE
        WTIME=WTIME+QUEUE*DELT
C
C     CHECA SI EXPIRO PERIODO DE SIMULACION
C
        T=T+DELT
        ARRTIM=ARRTIM-DELT
        TSER=TSER-DELT
        IF (T .LT. SLEN) GOTO 1
C
C     CALCULA E IMPRIME ESTADISTICAS FINALES
C
        AVEQ=AVEQ/(SLEN/DELT)
        AVES=AVES/(SLEN/DELT)
        IF (SFULL) I=1
        WTIME=WTIME/(NSER+I+QUEUE)
        WRITE(5,610)
610    FORMAT(1X)
        WRITE(5,603) DELT,SLEN
603    FORMAT(1X,'PARA UN INCREMENTO DE',5X,F10.5,5X,'
1 / MIN.',/,/, ' Y UN TIEMPO DE SIMULACION DE ',5X,F6.2,
1 / MIN.',/,/,5X,' LOS RESULTADOS SON:')
        WRITE(5,600)NSER,QUEUE,I
600    FORMAT(1X,'FUERON',5X,I3,5X,' CLIENTES ATENDIDOS',/,5X,I3,5X,'
1 / CLIENTES QUEDADOS EN COLA DE ESPERA Y ',/,5X,I1,5X,' CLIENTES'
1 ,3X,' QUEDADOS EN SERVICIO.')
```

```

        WRITE(5,601) AVEQ,AVES,WTIME
601    FORMAT(1X,'EL PROMEDIO DE LONGITUD DE COLA FUE',5X,F5.2,5X,/,
1 / PROMEDIO DE CLIENTES EN EL SISTEMA FUE',5X,F5.2,5X,/,
1 / PROMEDIO DE TIEMPO DE ESPERA POR CLIENTE FUE ',5X,F5.2,5X,
1 / MIN.')
```

```

        WRITE(5,602) TIDLE
602    FORMAT(1X,'EL CENTRO DE SERVICIO ESTUVO DESOCUPADO UN TOTAL DE',
1 5X,F5.2,5X,' MIN.')
```

```

        WRITE(5,608)
608    FORMAT(///,'*****'////)
        WRITE(5,607)AMBDAI
607    FORMAT(15X,'LA TAZA DE ARRIBO ES:',F5.2,/)
        WRITE(5,1234)RMUI
1234    FORMAT(15X,'LA TAZA DE SERVICIO ES:',F5.2,/)
        CALL COMPARA
        STOP
        END
C
C     SUBROUTINE ARRVL(ARRTIM,QUEUE,LAMBDA)
```

INTEGER QUEUE
REAL LAMBDA

COMMON NSEED, AMBDA1, RMU1
QUEUE=QUEUE +1
CALL EXPON(NSEED, ARRTIM, LAMBDA)
RETURN
END

SUBROUTINE SERCOM(TSER, QUEUE, NSER, MU, SFULL)
REAL MU
INTEGER QUEUE
LOGICAL SFULL

IF (SFULL) NSER=NSER+1
COMMON NSEED, AMBDA1, RMU1
IF (QUEUE .NE. 0) GOTO 10
SFULL= .FALSE.
RETURN
QUEUE=QUEUE-1
CALL EXPON(NSEED, TSER, MU)
SFULL= .TRUE.
RETURN
END

SUBROUTINE COMPARA

COMMON NSEED, AMBDA1, RMU1

COMPARACION CON LOS RESULTADOS ANALITICOS, LA NUMERACION REFERIDA
ES LA QUE SE DIO EN LA SECCION 6.1

PROMEDIO DE LONGITUD DE LA COLA (FORMULA 10)
 $RPLC = ((AMBDA1 / RMU1) * 2) / (1 - (AMBDA1 / RMU1))$

PROMEDIO DE CLIENTES EN EL SISTEMA (FORMULA 9)
 $RPCS = (AMBDA1 / RMU1) / (1 - (AMBDA1 / RMU1))$

PROMEDIO DE TIEMPO DE ESPERA EN COLA (FORMULA 12)
 $RPTEC = (1.) / (RMU1 - AMBDA1)$

WRITE(5, 700) RPLC
WRITE(5, 800) RPCS
WRITE(5, 900) RPTEC
FORMAT(1X, 'PROMEDIO DE LONGITUD DE COLA (ANALITICO)', 5X, F5.2)
FORMAT(1X, 'PROMEDIO DE CLIENTES EN EL SISTEMA (ANALITICO)', 5X,
1 F5.2)
FORMAT(1X, 'PROMEDIO DE ESPERA EN COLA (ANALITICO)', 5X, F5.2)
RETURN
END

700

800

900

DAR INCREMENTO, TIEMPO DE SIMULACION, TAZA DE ARRIBO,
TAZA DE SERVICIO

PARA UN INCREMENTO DE 1.00000 MIN.
Y UN TIEMPO DE SIMULACION DE 300.00MIN.

LOS RESULTADOS SON:

CULCOR	773	CLIENTES ATENDIDOS		
4		CLIENTES QUEDADOS EN COLA DE ESPERA Y		
1		CLIENTES QUEDADOS EN SERVICIO		
		EL PROMEDIO DE LONGITUD DE COLA FUE	1.43	
		PROMEDIO DE CLIENTES EN EL SISTEMA FUE	1.90	
		PROMEDIO DE TIEMPO DE ESPERA POR CLIENTE FUE	0.42	MIN.
		EL ULTIMO DE SERVICIO ESTUVO DESOCUPADO UN TOTAL DE	26.00	MIN.

LA TAZA DE ARRIBO ES: 4.00

LA TAZA DE SERVICIO ES: 6.00

PROMEDIO DE LONGITUD DE COLA (ANALITICO)	1.33	
PROMEDIO DE CLIENTES EN EL SISTEMA (ANALITICO)		2.00
PROMEDIO DE ESPERA EN COLA (ANALITICO)	0.50	

PARA UN INCREMENTO DE TIEMPO DE SIMULACION, TAZA DE ARRIBO,
TAZA DE SERVICIO

PARA UN INCREMENTO DE 2.00000 MIN.

Y UN TIEMPO DE SIMULACION DE 970.00MIN.

LOS RESULTADOS SON:

POBACION 495 CLIENTES ATENDIDOS

7 CLIENTES QUEDADOS EN COLA DE ESPERA Y

1 CLIENTES QUEDADOS EN SERVICIO.

EL PROMEDIO DE LONGITUD DE COLA FUE 0.60

PROMEDIO DE CLIENTES EN EL SISTEMA FUE 1.20

PROMEDIO DE TIEMPO DE ESPERA POR CLIENTE FUE 0.21 MIN.

EL CANTO DE SERVICIO ESTABA DESOCUPADO UN TOTAL DE 2.00 MIN.

LA TAZA DE ARRIBO ES: 4.00

LA TAZA DE SERVICIO ES: 11.00

PROMEDIO DE LONGITUD DE COLA (ANALITICO) 0.65

PROMEDIO DE CLIENTES EN EL SISTEMA (ANALITICO) 1.20

PROMEDIO DE ESPERA EN COLA (ANALITICO) 0.20

Interpretación de resultados

En este caso no tiene sentido hacer pruebas estadísticas para probar la validez de los resultados pues conocemos la solución analítica. Nótese la gran semejanza entre las soluciones por simulación y analítica.

6.2 Sistema Round Robin

Una de las situaciones que se presentan en el ambiente de sistemas operativos es el hecho de usar una computadora que trabaja en base a procesos en lote o "batch", que se procesan concurrentemente en el sistema. La eficiencia del sistema depende de parámetros tales como el "QUANTUM" de CPU, o cantidad de tiempo de CPU destinada a cada trabajo; el tiempo de CPU total requerido por un trabajo; y el tiempo de llegada entre arribos.

El siguiente modelo se realiza con el objeto de obtener la simulación de varios procesos teniendo en cuenta los parámetros iniciales del sistema y así encontrar una mejor combinación de parámetros o variables de control, que den una mejor eficiencia al sistema.

Las variables de control son:

1. QCPU: El "Quantum" de CPU para cada proceso
2. TMED: Medio de arribos
3. TPMED: Medio de demanda
4. OVERHD: tiempo de "overhead"

Así, nuestro problema se define de la siguiente manera:

En un computador orientado a procesos en lote, los trabajos son procesados por la "CPU" usando un plan de trabajo en "ROUND ROBIN", basado

en la regla de que el primero que llega, primero que sale. Si no se agota el tiempo de demanda, se regresa formandose de nuevo en la cola.

Los procesos llegan de acuerdo a un proceso Poisson con una media de arribo de 1.0 seg. Los requerimientos de procesador para los trabajos son independientes e idénticamente distribuidos según una densidad exponencial con media de demanda de 0.8 seg. (se demanda menos que lo que se arriba). Existe un "quantum" de procesamiento máximo por el "cpu" de 0.1 seg., sin incluir el tiempo entre cambio de procesos ("overhead"). El "overhead" es de 0.015 seg.

El plan de trabajo "Round Robin" significa que si se requiere más tiempo para procesar un trabajo, después de que ha recibido el quantum máximo de procesamiento, entonces, tal trabajo regresa al final de la cola (puede regresar a otro lugar) de espera para "cpu".

El sistema es ilustrado en la figura 6.5.

Se realiza una simulación para estimar el tiempo de respuesta desde que un trabajo llega para procesarse hasta que termina.

La técnica que se usa para el manejo de eventos (arribos y atención de procesos) es la de rastreo de eventos discutida en la sección 5.2.1.

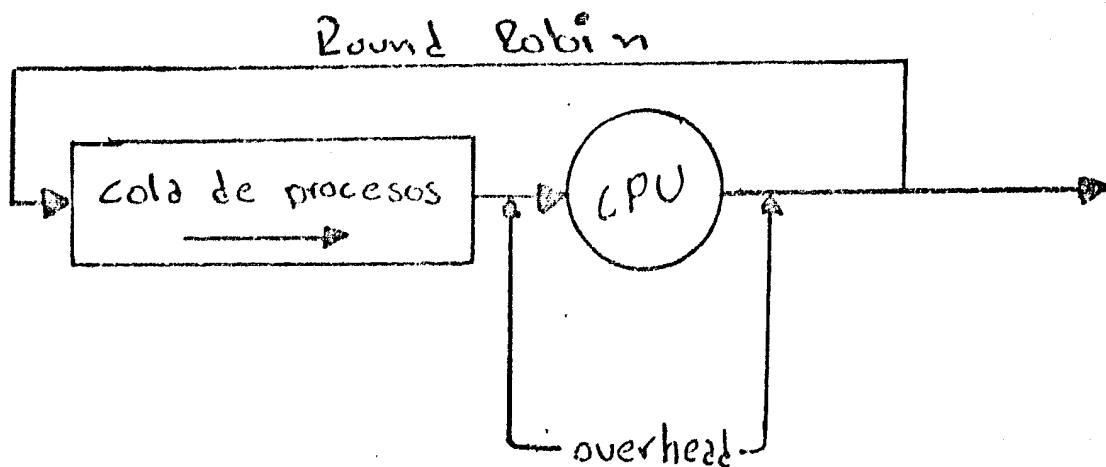


Figura 6.5 Sistema Round Robin.

El programa que hace la simulación no pierde generalidad al fijar variables de control, como las distribuciones de arribo y demanda, ya que se pueden cambiar fácilmente sin afectar el funcionamiento general del programa.

En cuanto a una posible solución analítica, es oportuno mencionar que este sistema es más complejo, y una solución analítica general parece casi imposible. Podría encontrarse una solución analítica si se le hacen simplificaciones al sistema, tales como; quitar el "overhead"; fijar las distribuciones de arribo y demanda; y otras que se podría tener la necesidad de simplificar.

La figura 6.6 muestra el diagrama de bloques del algoritmo en uso.

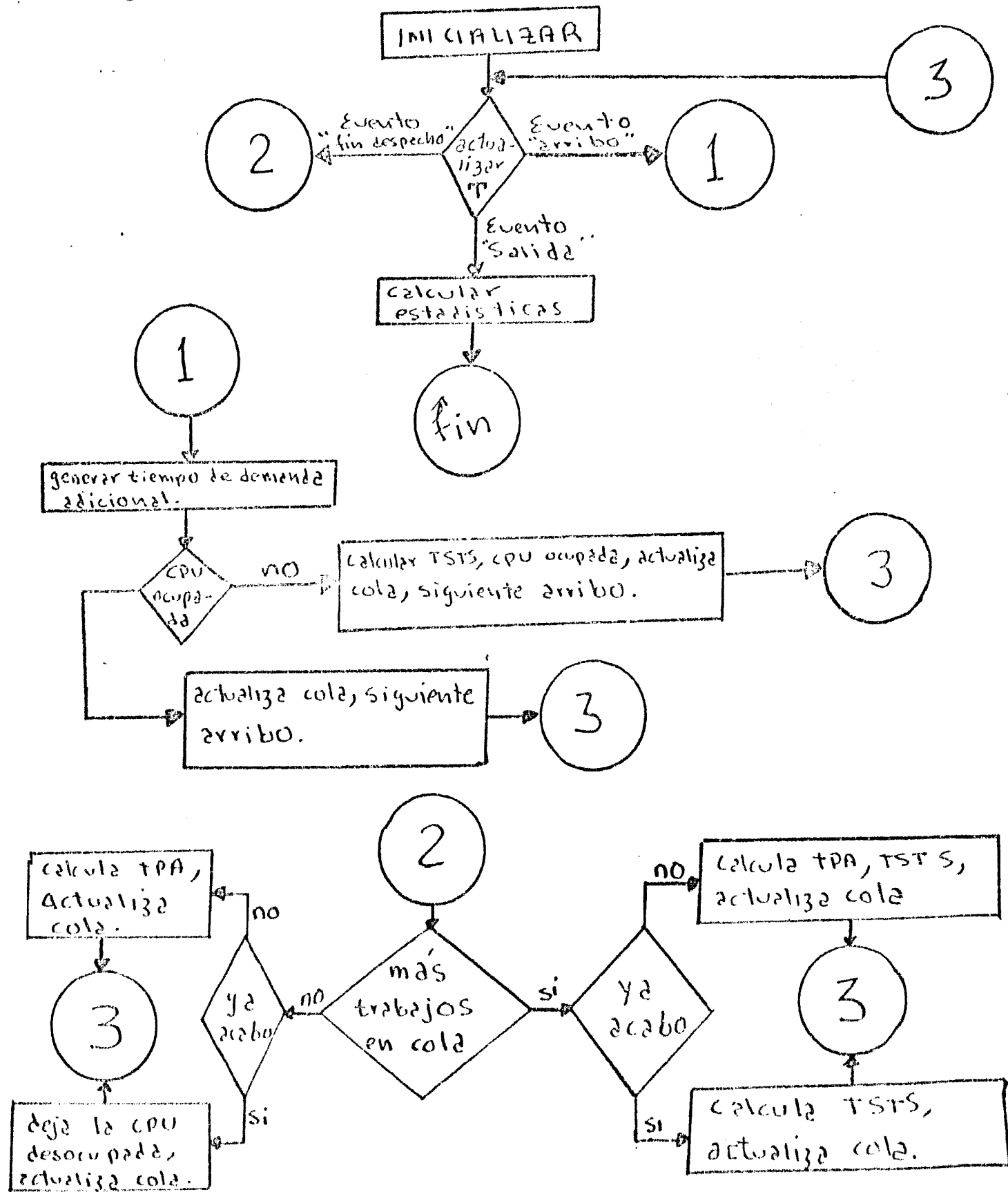


Figura 6.6 Algoritmo del sistema Round Robin

PROGRAMA BATCH

MODELO DE ENTRADA EN LOTE A UNA COMPUTADORA

LISTA DE VARIABLES:

- IARR = ARREGLO PARA ALMACENAR LOS TRABAJOS EN LA FILA:
IARR(1) ES EL NUMERO DE PROCESO EN EJECUCION
IARR(ITF) ES EL ULTIMO TRABAJO EN LA FILA.
- IDLE = STATUS DEL CPU IDLE=1 ES CPU DESOCUPADA
- ITF = TAMANO DE LA FILA DE ESPERA, INCLUYENDO EL TRABAJO
QUE ACTUALMENTE SE ESTA PROCESANDO
- QCPU = CUANTUM DE CPU
- TA(J) = TIEMPO DE ARRIBO DEL J-ESIMO TRABAJO
- TALTO = TIEMPO DE ALTO DE LA SIMULACION
- TAMED = TIEMPO DE ARRIBO MEDIO
- TIM = TIEMPO DE SIMULACION
- TP = TIEMPO DE PROCESAMIENTO ; MENOR O IGUAL A 0.1 SEG
- TPA(J) = TIEMPO ADICIONAL DE PROCESAMIENTO ADICIONAL REQUERIDO
POR EL J-ESIMO TRABAJO
- TFR(J) = TIEMPO DE PROCESAMIENTO DE CPU REQUERIDO POR EL
J-ESIMO TRABAJO
- TPRT(J) = TIEMPO DE PROCESAMIENTO DE CPU REQUERIDO POR EL
J-ESIMO TRABAJO TERMINADO
- TSA = TIEMPO DEL SIGUIENTE ARRIBO DE UN TRABAJO
- TSTS = TIEMPO AL CUAL EL SIGUIENTE TRABAJO SALE DEL CPU
- TTP(J) = TIEMPO DE TERMINACION DE PROCESAMIENTO PARA EL J-ESIMO
TRABAJO

REAL TA(1000), TTP(1000), TPA(1000), TFR(1000)

HASTA 25 TRABAJOS EN LA FILA DE BATCH

```

INTEGER IARR(25)
COMMON NSEED, TALTO, TSA, TSTS, TP, TIM, ITF, IDLE, NTMR, QCPU, TAMED,
1 TPMED, OVERHD
DO 33 I=1, 1000
TA(I)=0.0
TTP(I)=0.0
TPA(I)=0.0
TFR(I)=0.0
CONTINUE
33 DO 25 I=1, 25
IARR(I)=0
CONTINUE
25 TALTO=240.0
NSEED=5555

OVERHD=0.015
QCPU=0.1
TAMED=1.0
TPMED=0.8

```

C
C
C INICIALIZACION CON CUATRO TRABAJOS ESPERANDO ENTRADA A LA CPU
C PARA EVITAR TRANSITORIOS
C

TA(1)=0.0
TPA(1)=0.2
TPR(1)=TPMED
IARR(1)=1
TIM=TPMED
TP=QCPU
ITF=1
IDLE=0
TSTS=TIM +TP + OVERHD
NTMR=1
CALL EXPON(NSEED,TBA,TAMED)
TSA=TIM+TBA

C
C REALIZA EL SIGUIENTE EVENTO
C

50 CONTINUE
IF ((TSA .LT. TSTS) .AND. (TSA .LT. TALTO)) GO TO 11
21 CONTINUE
IF ((TSTS .LE. TSA) .AND. (TSTS .LE. TALTO)) GOTO 12
22 GOTO 133
11 CALL ARRIBO(TA,TTP,TPA,TPR,IARR)
GOTO 21
12 CALL PROC(TA,TTP,TPA,IARR)
GOTO 22
133 IF((TSA .LT. TALTO) .OR. (TSTS .LE. TALTO)) GOTO 50
CALL SALIDA(TA,TTP,TPR)
STOP
END

C
C
C SUBROUTINA QUE GENERA EL TIEMPO DE ARRIBO DE LOS PROCESOS
C

SUBROUTINE ARRIBO(TA,TTP,TPA,TPR,IARR)

C
C
C IARR = ARREGLO PARA ALMACENAR LOS TRABAJOS EN LA FILA ;
C IARR(1) ES EL NUMERO DE PROCESO EN EJECUCION
C IARR(ITF) ES EL ULTIMO TRABAJO EN LA FILA
C TA(J) = TIEMPO DE ARRIBO DEL J-ESIMO TRABAJO
C TPA(J) = TIEMPO DE PROCESAMIENTO ADICIONAL REQUERIDO
C POR EL J-ESIMO TRABAJO
C TPR(J) = TIEMPO DE PROCESAMIENTO DE CPU REQUERIDO POR
C EL J-ESIMO TRABAJO
C TTP(J) = TIEMPO DE TERMINACION DE PROCESAMIENTO PARA EL
C J-ESIMO TRABAJO
C

REAL TA(1000),TTP(1000),TPA(1000),TPR(1000)
INTEGER IARR(250)
COMMON NSEED,TALTO,TSA,TSTS,TP,TIM,ITF,IDLE,NTMR,QCPU,TAMED,
1 TPMED,OVERHD

```
TIM=TSA
NTMR=NTMR+1
TA(NTMR)=TSA
CALL EXPON(NSEED,TREQ,TPMED)
TFA(NTMR)=TREQ
TPR(NTMR)=TREQ
IF(IDLE .EQ. 0) GOTO 299
```

C
C
C

LA CPU ESTA DESOCUPADA

```
IDLE=0
TP=TFA(NRMA)
IF(TP .GT. QCPU) TP=QCPU
TSTS=TIM+TP+OVERHD
```

C
C
C

LA CPU ESTA OCUPADA

299

```
ITF=ITF+1
IARR(ITF)=NTMR
```

C
C
C

GENERA TIEMPO DE ARRIBO DEL SIGUIENTE TRABAJO

```
CALL EXPON(NSEED,TBA,TAMED)
TSA=TIM+TBA
RETURN
END
```

C
C
C

SUBROUTINA QUE CONTROLA LOS PROCESOS DE LA CPU

SUBROUTINE PROC(TA,TTP,TPA,IARR)

C
C
C
C
C
C
C
C
C

```
IARR = ALMACEN DE TRABAJOS EN LA FILA;
      IARR(1) ES EL NUMERO DEL PROCESO EN EJECUCION
      IARR(ITF) ES EL ULTIMO EN LA FILA.
TA(J) = TIEMPO DE ARRIBO EN EL J-ESIMO TRABAJO
TPA(J) = TIEMPO DE PROCESAMIENTO ADICIONAL REQUERIDO
        POR EL J-ESIMO TRABAJO
TTP(J) = TIEMPO DE TERMINACION DE PROCESAMIENTO
        PARA EL J-ESIMO TRABAJO
```

```
REAL TA(1000), TTP(1000), TPA(1000)
INTEGER IARR(250)
COMMON NSEED, TALTO, TSA, TSTS, TP, TIM, ITF, IDLE, NTMR, QCPU,
1 TAMED, TPMED, OVERHD
```

C

```
TIM = TSTS
TPA(IARR(1)) = TPA(IARR(1))-TP
```

C
C
C

CHECA SI OTROS PROCESOS ESPERAN ENTRAR A CPU

```
IF (ITF .GT. 1) GOTO 350
```

C
C
C

NO HAY MAS PROCESOS ESPERANDO ENTRAR A CPU

C

```
IF (TPA(IARR(1)) .LE. 0.0001) GOTO 340
```

C
C
TODAVIA NO TERMINA EL PROCESO

TP = TPA(IARR(1))
IF (TP .GT. QCPU) TP = QCPU
TSTS = TIM + TP + OVERHD
RETURN

C
C
C
340
TERMINO EL PROCESO

TTP(IARR(1)) = TIM
TSTS = TALTO + 1.0
IDLE = 1
ITF = 0
RETURN

C
C
C
350
OTROS TRABAJOS ESPERAN PARA ENTRAR A CPU

IF (TPA(IARR(1)) .LE. 0.0001) GOTO 380

C
C
C
TODAVIA NO TERMINA EL PROCESO

C
C
C
PONE EL TRABAJO FINAL EN LA COLA

IARR1 = IARR(1)
ITF1 = ITF - 1
DO 360 I=1, ITF1
I1 = I + 1
IARR(I) = IARR(I1)
CONTINUE
IARR(ITF) = IARR1

360
C
TP = TPA(IARR(1))
IF (TP .GT. QCPU) TP = QCPU
TSTS = TIM + TP + OVERHD
RETURN

C
C
C
EL PROCESO TERMINO

C
C
C
ACTUALIZA LA COLA

380
C
TTP(IARR(1)) = TIM
ITF = ITF - 1
DO 390 I = 1, ITF
I1 = I + 1
IARR(I) = IARR(I1)
CONTINUE
ITF1 = ITF + 1
IARR(ITF1) = 0

390
C
TP = TPA(IARR(I))
IF (TP .GT. QCPU) TP = QCPU
TSTS = TIM + TP + OVERHD
RETURN

C
C
END

```
C
C      SUBROUTINA QUE OBTIENE LAS ESTADISTICAS Y LA SALIDA DE LOS
C PROCESOS EN LOTE,
C
C      SUBROUTINE SALIDA(TA,TTP,TFR)
C
C      TA(J)   = TIEMPO DE ARRIBO DEL J-ESIMO TRABAJO
C      TFR(J)  = TIEMPO DE PROCESAMIENTO DE CPU REQUERIDO POR
C                EL J-ESIMO TRABAJO
C      TTP(J)  = TIEMPO DE TERMINACION DE PROCESAMIENTO PARA EL
C                J-ESIMO TRABAJO
C
C
C      REAL TA(1000),TTP(1000),TFR(1000)
C      REAL TR(1000),TPRT(1000)
C      COMMON NSEED,TALTO, TSA, TSTS, TP, TIM, ITF, IDLE, NTMR, QCPU, TAMED,
C      1 TPRED, OVERHD
C
C      DETERMINA PROCESOS TERMINADOS E IMPRIME
C LA SALIDA DE CADA PROCESO
C
C      WRITE(5,123)
123  FORMAT(/,20X,'JOB      TIEMPO DE      TIEMTO      TIEMPO DE',/,20X,
1'      ARRIBO      DE CPU      RESPUESTA',/,20X,
1'      (SEGS.)    (SEGS.)    (SEGS.)',/)
      NJC=0
      DO 480 II=1,NTMR
      IF(TTP(II) .LE. 0.0001) GOTO 400
C
C      EL PROCESO TERMINA EN EL TIEMPO "TALTO"
C
      NJC=NJC+1
      TR(NJC)=TTP(II)-TA(II)
      TPRT(NJC)=TFR(II)
      WRITE(5,1001) II,TA(II),TFR(II),TR(NJC)
1001  FORMAT(22X,I3,3X,F7.3,4X,F5.3,5X,F7.3)
      GOTO 480
C
C      EL TRABAJO NO FUE TERMINADO EN EL TIEMPO "TALTO"
C
400  WRITE(5,1002) II,TA(II),TFR(II)
1002  FORMAT(22X,I3,3X,F7.3,4X,F5.3,'NO FUE TERMINADO')
480  CONTINUE
C
C      CALCULOS ESTADISTICOS
C
      CALL STAT(NJC,TR,TRM,TRDE)
      CALL STAT(NJC,TPRT,TPRM,TPRDE)
      CALL CORR(NJC,TR,TPRT,TRM,TRDE,TPRM,TPRDE,CC)
      WRITE(5,431)
431  FORMAT(///,' LOS PARAMETROS DE LA SIMULACION SON LOS'
1' y' SIGUIENTES:(')
      WRITE(5,144) QCPU
144  FORMAT(/,' EL QUANTUM DE CPU PARA ESTA CORRIDA FUE DE: ',F6.3)
      WRITE(5,154) TAMED
154  FORMAT(/,' EL TIEMPO DE ARRIBO MEDIO REQUERIDO PARA ESTA CORRIDA
```

```
1 FUE DE : 'F6.3)
WRITE(5,1540) TPMED
1540 FORMAT(/, ' EL TIEMPO DE PROCESAMIENTO MEDIO PARA ESTA CORRIDA
1 FUE DE : 'F6.3)
WRITE(5,1448) OVERHD
1448 FORMAT(/, ' EL OVERHEAD DE CPU PARA ESTA CORRIDA FUE DE : ',
1 F6.3)
C
C GENERA LA SALIDA ESTADISTICA
C
WRITE(5,4310)
4310 FORMAT(/, ' LAS ESTADISTICAS DE LA SIMULACION SON: ')
WRITE(5,1006) TRM,TRDE
1006 FORMAT(/, ' EL TIEMPO MEDIO DE RESPUESTA ES: 'F7.4, //,
1 ' LA DESVIACION ESTANDAR DEL TIEMPO DE RESPUESTA ES 'F7.4)
WRITE(5,1007) CC
1007 FORMAT(/, ' LA CORRELACION ENTRE EL TIEMPO REQUERIDO DE CPU Y '
1 /, ' EL TIEMPO DE RESPUESTA ES 'F6.4)
CALL HISTO(NJC,TR,0.0,60.0,30)
RETURN
END
```


JOB	TIEMPO DE ARRIBO (SEGS.)	TIEMPO DE CPU (SEGS.)	TIEMPO DE RESPUESTA (SEGS.)
1	0.000	0.800	1.030
2	1.525	0.605	0.725
3	2.459	3.576	7.147
4	5.320	1.001	4.180
5	6.109	0.920	4.675
6	6.609	1.818	11.378
7	7.273	0.380	2.196
8	8.992	0.087	0.732
9	9.543	0.061	0.271
10	9.860	0.270	1.255
11	10.311	1.091	5.498
12	11.510	0.508	4.336
13	11.869	0.463	3.588
14	12.031	0.846	6.033
15	12.307	0.115	1.268
16	13.166	1.446	7.909
17	13.364	1.310	7.405
18	13.392	0.035	0.708
19	13.518	0.555	5.206
20	14.124	0.205	3.125
21	14.184	0.174	1.766
22	16.319	0.420	2.801
23	19.942	1.611	3.751
24	20.346	0.717	1.926
25	22.779	0.659	1.233
26	24.205	1.193	1.388
27	26.110	0.237	0.412
28	26.346	2.537	5.146
29	27.908	2.100	6.840
30	28.923	0.144	0.548
31	30.229	0.779	3.549
32	30.321	0.342	2.048
33	31.424	0.305	1.555
34	32.149	0.605	2.470
35	33.083	3.576	14.251
36	35.944	1.001	4.235
37	36.734	0.920	4.509
38	37.234	1.818	13.945
39	37.897	0.380	2.251
40	39.616	0.087	0.911
41	40.168	0.061	0.910
42	40.484	0.270	1.318
43	40.935	1.091	9.263
44	42.134	0.508	4.645
45	42.494	0.463	4.934
46	42.655	0.846	7.964
47	42.931	0.115	1.447
48	43.790	1.446	10.770
49	43.989	1.310	9.299
50	44.016	0.035	0.886
51	44.143	0.555	5.359
52	44.749	0.205	2.828
53	44.808	0.174	2.190
54	46.943	0.420	3.842

55	50.566	1.611	6.425
56	50.970	0.717	3.285
57	53.403	0.659	2.166
58	54.829	1.193	3.156
59	56.734	0.237	0.668
60	56.971	2.537	8.271
61	58.532	2.100	7.529
62	59.547	0.144	0.582
63	60.853	0.779	4.322
64	60.946	0.342	1.785
65	62.049	0.305	1.982
66	62.773	0.605	3.323
67	63.707	3.576	20.429
68	66.568	1.001	4.155
69	67.358	0.920	4.313
70	67.858	1.818	12.596
71	68.522	0.380	2.170
72	70.240	0.087	0.830
73	70.792	0.061	0.369
74	71.108	0.270	1.123
75	71.559	1.091	8.301
76	72.759	0.508	5.566
77	73.118	0.463	4.260
78	73.280	0.846	8.566
79	73.555	0.115	1.481
80	74.415	1.446	11.103
81	74.613	1.310	9.908
82	74.641	0.035	0.805
83	74.767	0.555	5.638
84	75.373	0.205	3.906
85	75.432	0.174	2.280
86	77.567	0.420	4.087
87	81.190	1.611	7.676
88	81.594	0.717	4.660
89	84.028	0.659	2.661
90	85.453	1.193	4.110
91	87.359	0.237	1.121
92	87.595	2.537	10.885
93	89.156	2.100	8.222
94	90.171	0.144	0.616
95	91.478	0.779	3.666
96	91.570	0.342	1.819
97	92.673	0.305	2.017
98	93.398	0.605	3.276
99	94.332	3.576	21.883
100	97.193	1.001	5.662
101	97.982	0.920	5.612
102	98.482	1.818	13.320
103	99.146	0.380	2.204
104	100.865	0.087	1.063
105	101.416	0.061	1.177
106	101.733	0.270	2.192
107	102.184	1.091	8.450
108	103.383	0.508	5.600
109	103.742	0.463	4.295
110	103.904	0.846	8.715
111	104.180	0.115	1.515
112	105.039	1.446	12.632

113	105.237	1.310	10.871
114	105.265	0.035	0.840
115	105.391	0.555	5.673
116	105.997	0.205	3.940
117	106.057	0.174	2.314
118	108.192	0.420	4.351
119	111.815	1.611	8.975
120	112.219	0.717	5.844
121	114.652	0.659	3.500
122	116.078	1.193	4.950
123	117.983	0.237	1.385
124	118.219	2.537	11.725
125	119.781	2.100	9.406
126	120.796	0.144	0.880
127	122.102	0.779	4.275
128	122.194	0.342	1.854
129	123.297	0.305	1.936
130	124.022	0.605	3.425
131	124.956	3.576	23.835
132	127.817	1.001	6.946
133	128.607	0.920	6.322
134	129.107	1.818	13.924
135	129.770	0.380	2.124
136	131.489	0.087	0.522
137	132.041	0.061	0.521
138	132.357	0.270	1.685
139	132.908	1.091	10.509
140	134.007	0.508	6.015
141	134.367	0.463	5.238
142	134.528	0.846	9.325
143	134.804	0.115	1.435
144	135.663	1.446	12.101
145	135.862	1.310	11.712
146	135.889	0.035	0.874
147	136.016	0.555	6.736
148	136.622	0.205	3.363
149	136.681	0.174	2.371
150	138.816	0.420	4.265
151	142.439	1.611	10.004
152	142.843	0.717	6.339
153	145.276	0.659	5.902
154	146.702	1.193	6.324
155	148.607	0.237	1.562
156	148.844	2.537	15.069
157	150.405	2.100	10.362
158	151.420	0.144	0.982
159	152.726	0.779	4.886
160	152.819	0.342	1.889
161	153.922	0.305	1.972
162	154.646	0.605	3.461
163	155.580	3.576	27.639
164	158.441	1.001	6.523
165	159.231	0.920	7.898
166	159.731	1.818	15.588
167	160.395	0.380	2.671
168	162.113	0.087	0.727
169	162.665	0.061	0.606
170	162.981	0.270	1.951

171	163.432	1.091	9.326
172	164.632	0.508	5.233
173	164.991	0.463	6.611
174	165.153	0.846	9.543
175	165.428	0.115	1.651
176	166.288	1.446	12.673
177	166.486	1.310	12.629
178	166.514	0.035	0.795
179	166.640	0.555	6.664
180	167.246	0.205	3.229
181	167.305	0.174	2.407
182	169.440	0.420	4.374
183	173.063	1.611	10.312
184	173.467	0.717	5.694
185	175.901	0.659	4.960
186	177.326	1.193	7.970
187	179.232	0.237	1.696
188	179.468	2.537	16.700
189	181.029	2.100	13.597
190	182.044	0.144	1.519
191	183.351	0.779	4.347
192	183.443	0.342	2.156
193	184.546	0.305	2.813
194	185.271	0.605	3.498
195	186.205	3.576	28.899
196	189.066	1.001	10.169
197	189.855	0.920	9.003
198	190.355	1.818	17.579
199	191.019	0.380	4.062
200	192.738	0.087	0.863
201	193.289	0.061	0.362
202	193.606	0.270	2.381
203	194.057	1.091	11.759
204	195.256	0.508	6.892
205	195.615	0.463	5.080
206	195.777	0.846	8.487
207	196.053	0.115	2.001
208	196.912	1.446	14.865
209	197.110	1.310	12.704
210	197.138	0.035	1.095
211	197.264	0.555	6.119
212	197.870	0.205	4.240
213	197.930	0.174	2.328
214	200.065	0.420	5.285
215	203.688	1.611	11.917
216	204.092	0.717	5.769
217	206.525	0.659	6.673
218	207.951	1.193	9.803
219	209.856	0.237	1.988
220	210.092	2.537	20.493
221	211.654	2.100	14.548
222	212.669	0.144	1.293
223	213.975	0.779	5.533
224	214.067	0.342	2.759
225	215.170	0.305	3.194
226	215.895	0.605	4.569
227	216.829	3.576	NO FUE TERMINADO
228	219.690	1.001	9.583

229	220.480	0.920	9.763
230	220.980	1.818	NO FUE TERMINADO
231	221.643	0.380	3.164
232	223.362	0.087	0.899
233	223.913	0.061	0.783
234	224.230	0.270	2.417
235	224.681	1.091	11.094
236	225.880	0.508	8.423
237	226.240	0.463	6.531
238	226.401	0.846	10.880
239	226.677	0.115	3.726
240	227.536	1.446	NO FUE TERMINADO
241	227.735	1.310	NO FUE TERMINADO
242	227.762	0.035	1.365
243	227.889	0.555	7.535
244	228.495	0.205	3.493
245	228.554	0.174	2.479
246	230.689	0.420	5.711
247	234.312	1.611	NO FUE TERMINADO
248	234.716	0.717	NO FUE TERMINADO
249	237.149	0.659	NO FUE TERMINADO
250	238.575	1.193	NO FUE TERMINADO

LOS PARAMETROS DE LA SIMULACION SON LOS SIGUIENTES:

EL QUANTUM DE CPU PARA ESTA CORRIDA FUE DE: 0.100

EL TIEMPO DE ARRIBO MEDIO REQUERIDO PARA ESTA CORRIDA FUE DE : 1.000

EL TIEMPO DE PROCESAMIENTO MEDIO PARA ESTA CORRIDA FUE DE : 0.800

EL OVERHEAD DE CPU PARA ESTA CORRIDA FUE DE : 0.015

LAS ESTADISTICAS DE LA SIMULACION SON:

EL TIEMPO MEDIO DE RESPUESTA ES: 8.0827

LA DESVIACION ESTANDAR DEL TIEMPO DE RESPUESTA ES 9.6087

LA CORRELACION ENTRE EL TIEMPO REQUERIDO DE CPU Y
EL TIEMPO DE RESPUESTA ES 0.8968

DATOS DEL HISTOGRAMA

RANGO		FREC.	FREC. RELATIVA
DE	A		
0.00	2.00	63	0.26033 *****
2.00	4.00	49	0.20248 *****
4.00	6.00	46	0.19008 *****
6.00	8.00	25	0.10331 *****
8.00	10.00	20	0.08264 *****
10.00	12.00	16	0.06612 *****
12.00	14.00	10	0.04132 *****
14.00	16.00	5	0.02066 *****
16.00	18.00	2	0.00826 !
18.00	20.00	0	0.00000 !
20.00	22.00	3	0.01240 !*
22.00	24.00	1	0.00413 !
24.00	26.00	0	0.00000 !
26.00	28.00	1	0.00413 !
28.00	30.00	1	0.00413 !
30.00	32.00	0	0.00000 !
32.00	34.00	0	0.00000 !
34.00	36.00	0	0.00000 !
36.00	38.00	0	0.00000 !
38.00	40.00	0	0.00000 !
40.00	42.00	0	0.00000 !
42.00	44.00	0	0.00000 !
44.00	46.00	0	0.00000 !
46.00	48.00	0	0.00000 !
48.00	50.00	0	0.00000 !
50.00	52.00	0	0.00000 !
52.00	54.00	0	0.00000 !
54.00	56.00	0	0.00000 !
56.00	58.00	0	0.00000 !
58.00	60.00	0	0.00000 !

TOTAL DE REPLICACIONES : 242

Interpretación de los resultados

El parámetro que se varió en las simulaciones fue el "quantum" de "CPU", ya que es una variable perfectamente controlable en este sistema.

Los resultados de la simulación se muestran en la siguiente tabla:

Resultados del programa.				
Variables de control: Quantum				
Variables de respuesta: tiempos de respuesta, número de trabajos terminados, correlación req./resp.				
Quantum	media de respuesta.	desv. estándar de respuesta	correlación req./resp.	# trabajos terminados
0.05	19.4141	22.1499	0.9097	208
0.06	15.6845	17.2597	0.9230	222
0.08	10.3309	12.274	0.8913	224
0.1	8.0827	9.6087	0.8968	225
0.2	5.9301	6.9332	0.8658	225
0.4	5.2563	6.1644	0.8176	224
0.7	4.5327	5.7126	0.7213	226
0.8	4.9607	5.8718	0.6901	226

Según la tabla anterior, el mejor "quantum" de "CPU" está en la vecindad de 0.2, ya que a partir del siguiente paso (quantum = 0.4) no se logra una gran mejora en el tiempo

6.3 Sistema de memoria virtual

En algunas computadoras, la memoria principal se organiza en pequeñas unidades llamadas "páginas". Cualquier programa (instrucciones y datos) de un usuario puede ocupar una o más páginas. El manejo de estas es ajeno al usuario y se hace por medio del sistema operativo. Asociado con una memoria paginada, se encuentra un dispositivo periférico de memoria secundaria. Todo programa que entra al sistema es dividido en páginas de igual tamaño que las de memoria. Cuando un programa se carga en memoria, todas sus páginas pueden estar en memoria principal, o estar repartidas entre la memoria principal (páginas activas) y la memoria secundaria (páginas pasivas).

El programa se empieza a ejecutar por alguna instrucción contenida en alguna página activa. Si en el curso de la ejecución toca turno a una instrucción que se halla en una página pasiva entonces el programa se interrumpe (lo hace el sistema operativo) mientras se localiza la página demandada y es cargada en memoria principal. Generalmente la página cargada en memoria tomará el lugar de otra página activa que a la vez es actualizada en memoria secundaria. Al proceso de remover páginas activas se le llama "swapping".

Un sistema de memoria paginada ofrece varias ventajas a los programadores. El beneficio más importante es el incremento de memoria disponible. Un programa no debe estar acotado por el tamaño físico de la memoria principal; si un programa es

más grande que la memoria principal, este es dividido en paginas, algunas de las cuales se almacenan en memoria secundaria hasta su demanda. Así, un sistema de memoria paginada permite al programador actuar como si tuviera disponible un gran espacio de memoria. A esta facilidad de almacenamiento, que se le da a los programadores generalmente se le llama memoria "virtual" para distinguir de la memoria principal real. La figura 6.7 ilustra un sistema de memoria virtual.

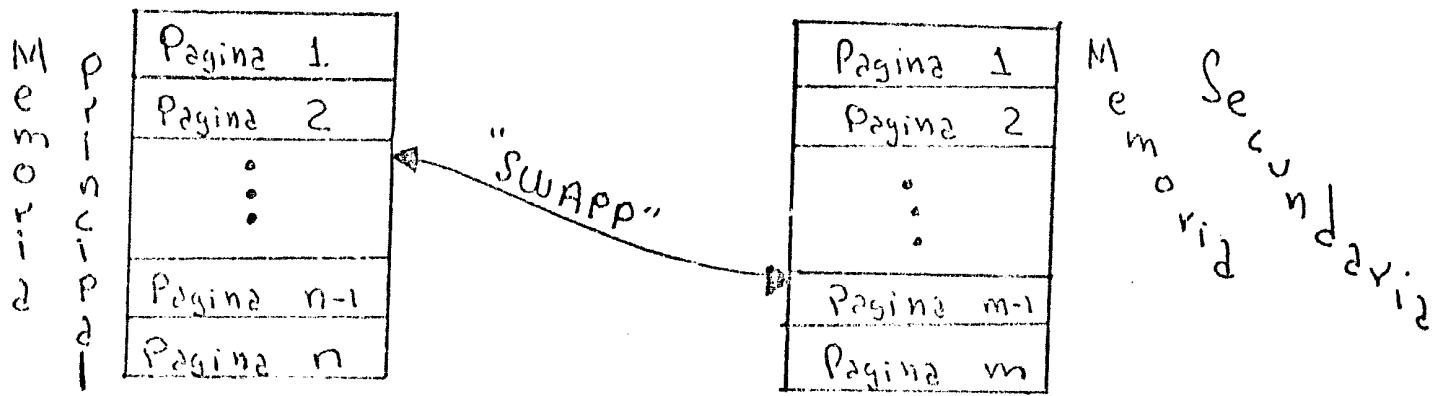


Figura 6.7 Sistema de memoria virtual

Una decisión importante al momento de diseñar el sistema de memoria virtual es la elección de la disciplina de "swapp" o movimiento de paginas.

Una ayuda para esta decisión la encontramos en la simulación. Se llevara a cabo una simulación donde las variables de control (ver capítulo V) son: máximo de paginas activas por programa, número de paginas por programa, tiempo de "swapp", tiempo de procesador por instrucción, comportamiento de la secuencia de instrucciones en su ejecución y algoritmo de "swapp". Mientras que las variables de respuesta que nos

ayudan a elegir el mejor algoritmo son: tiempo medio de ejecución por instrucción y la desviación estándar del tiempo de ejecución por instrucción.

El tiempo de procesador por instrucción puede obtenerse por estadísticas del procesador en estudio. El tiempo de "swapping" por lo general es una constante necesaria para hacer la entrada y salida (cambio de páginas). El comportamiento de las instrucciones en su ejecución también se obtiene por medio de estadísticas. Se analizarán varios algoritmos de "swapping", para observar el comportamiento del sistema se analizan las variables de respuesta.

El tiempo de procesador lo obtenemos de muestrear alguna distribución de probabilidad dada, por la naturaleza de los procesadores esta distribución generalmente será una normal con parámetros dados.

El orden que siguen las instrucciones a veces puede ser caracterizado como una cadena de Markov, esto es; Si denotamos por $P(1), P(2), \dots$ el orden en que son referenciadas las páginas (hay "NPP" páginas). Entonces las probabilidades de transición

$$\text{Prob}[(P(k+1))=j : P(k)=i]_{ij} \equiv P_{ij}$$

(Probabilidad de que: dado que la k -ésima instrucción está en la página "i" entonces la $(k+1)$ -ésima esté en la página "j".)

están dadas por la matriz estocástica P_{ij} , de entradas p_{ij} ; $1 \leq i, j \leq NPP$

Para la simulación también debemos determinar el máximo de páginas activas por programa y el número de páginas que tienen los programas.

Algunas alternativas para seleccionar el algoritmo de "swapp" son:

*) "Swapp" aleatorio, donde la página activa para hacer el cambio es tomada aleatoriamente.

*) "Swapp" "fifo", donde la página activa más antigua es la que se selecciona.

*) "Swapp" de frecuencia, donde la página activa menos frecuentada es la seleccionada.

Así, con la ayuda de la simulación se elige el algoritmo de "swapp" óptimo, para programas caracterizados por las variables de control que se deseen.

Un tratado completo sobre estos sistemas de memoria virtual puede ser visto en [6.3]

En nuestra simulación suponemos lo siguiente sobre las variables de control.

T tiempo de ejecución de las instrucciones. Se

supone que el tiempo de ejecución de las instrucciones se distribuye normal con media 0.02 seg. y varianza de 0.005 seg. cuadrados.

Número de paginas activas. Se toma un maximo de 3 paginas activas por programa.

Numero de paginas por programa. Se consideran programas de 2 a 5 paginas.

Tiempo de "swapp". Una constante de .5 seg.

Flujo de las instrucciones. Gobernado por una cadena de "MARKOV" con matriz de transicion

$$P_{ij} = \begin{bmatrix} .65 & .10 & .05 & .05 & .05 & .1 \\ .10 & .65 & .10 & .05 & .05 & .05 \\ .05 & .10 & .65 & .10 & .05 & .05 \\ .05 & .05 & .10 & .65 & .10 & .05 \\ .05 & .05 & .05 & .10 & .65 & .10 \\ .10 & .05 & .05 & .10 & .10 & .65 \end{bmatrix}$$

Algoritmos a probar. Se analizan los algoritmos "random" y "fifo".

La figura 6.8 muestra el diagrama de bloques usado en la simulación.

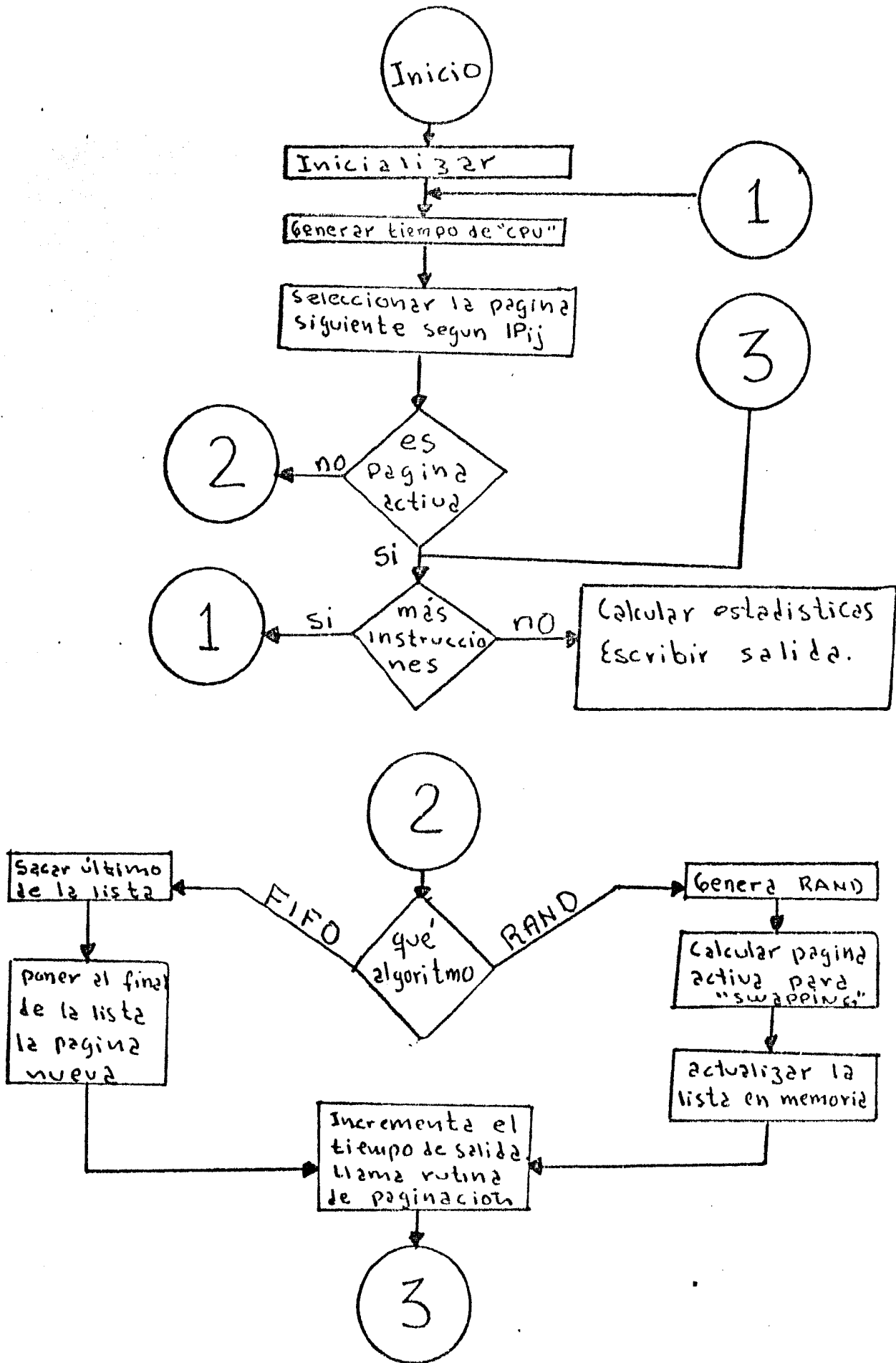


Figura 6.8 Diagrama de bloques, memoria virtual.

PROGRAMA 'VIRT'

```

DIMENSION INMED(3),TSAL(500),PRBMAT(6,5)
DATA INMED /1,2,3/
DATA PRBMAT /0.65,0.1,0.05,0.05,0.05,0.1,
1      0.75,0.75,0.15,0.1,0.1,0.15,
1      0.8,0.85,0.8,0.20,0.15,0.20,
1      0.85,0.9,0.9,0.85,0.25,0.25,
1      0.9,0.95,0.95,0.95,0.9,0.35/

```

EL ALGORITMO DE "SWAPP" ES DADO POR :

```

J=1 ES ALEATORIO
J=2 ES FIFO

```

DO 20 J=1,2

VALORES INICIALES

```

NSEED2=5555
NSEED=5555
NINSTR=500      !NUMERO DE INSTRUCCIONES
IPAGAC=1       !PAGINA INICIAL EN MEMORIA
KPROG=3
TEMED=0.02     !TIEMPO DE EJECUCION MEDIO
TEVAR=0.005    !VARIANZA DEL TIEMPO DE EJECUCION
NUMPAG=4       !NUMERO DE PAGINAS DE QUE CONSISTE EL
                !PROGRAMA ( MENOR QUE 5)
TCARME=0.5     !TIEMPO DE CARGA EN MEMORIA

```

DO 10 I=1,NINSTR

CALCULA EL TIEMPO DE EJECUCION

CALL NORMAL(NSEED,TSAL(I),TEMED,TEVAR)

CALCULA LA PAGINA DE LA SIGUIENTE INSTRUCCION

CALL RAND(NSEED2,RAN)

DO 31 K=1,NUMPAG

IF (PRBMAT(IPAGAC,K) .GT. RAN) GOTO 34

CONTINUE

K=K+1

NVAPAG=K

CHECA SI LA SIGUIENTE PAGINA ESTA EN MEMORIA

```

IF(INMED(1) .EQ. NVAPAG .OR. INMED (2) .EQ. NVAPAG
1 .OR. INMED(3) .EQ. NVAPAG) GOTO 10

```

CALL SWPAG(NSEED,INMED,KPROG,NVAPAG,J)

TSAL(I)=TSAL(I)+TCARME

CONTINUE

10

```
C
C      CALCULA ESTADISTICAS
C
C      CALL STAT(NINSTR,TSAL,TMEDIO,TDESV)
C
C      GENERA SALIDA
C
C      IF(J.EQ.1) WRITE (5,100)
C      IF (J.EQ.2) WRITE (5,200)
100   FORMAT(1H1,15X,'LOS RESULTADOS DEL PROGRAMA "VIRT" SON ',//,15X,
C      1 'ALGORITMO DE "SWAPP" ALEATORIO')
200   FORMAT(//,15X,'ALGORITMO DE "SWAPP" FIFO "')
C      WRITE(5,300)NINSTR,NVAPAG,TMEDIO,TDESV
300   FORMAT(//,5X,'SIMULACION CON ',14,' INSTRUCCIONES',//
C      1 ,5X,' PAGINAS DE INSTRUCCIONES DEL PROGRAMA ',11,
C      1 //,5X,' TIEMPO MEDIO DE EJECUCION',F10.4//,
C      1 5X,' DESVIACION ESTANDAR DEL TIEMPO DE EJECUCION'
C      1 ,F10.4,///,8X,
C      1 'DISTRIBUCION DE LA EJECUCION TIEMPO/INSTRUCCION')
C      CALL HISTO(NINSTR,TSAL,0.0,0.6,40)
20   CONTINUE
C      END
C
C      "SWAPP"
C
C      SUBROUTINE SWPAG(NSEED,INMED,KPROG,NVAPAG,J)
C
C      DIMENSION INMED(3)
C      IF(J.EQ.2) GOTO 1
C      CALL RAND(NSEED,RAN)
C      I=3*RAN+1
C      INMED(I)=NVAPAG
C      GOTO 2
1     KPROG=KPROG+1
C      IF(KPROG.EQ.4) KPROG=1
C      I=KPROG
C      INMED(I)=NVAPAG
2     RETURN
C      END
```

LOS RESULTADOS DEL PROGRAMA "VIRT" SON
ALGORITMO DE "SWAPP" ALEATORIO

SIMULACION CON 500 INSTRUCCIONES

PAGINAS DE INSTRUCCIONES DEL PROGRAMA : 4

TIEMPO MEDIO DE EJECUCION 0.0928

DESVIACION ESTANDAR DEL TIEMPO DE EJECUCION 0.1768

DISTRIBUCION DE LA EJECUCION TIEMPO/INSTRUCCION

DATOS DEL HISTOGRAMA

RANGO DE	A	FREC.	FREC. RELATIVA	
0.00	0.02	123	0.24600	!*****
0.02	0.03	260	0.52000	!*****
0.03	0.05	10	0.02000	!**
0.05	0.06	0	0.00000	!
0.06	0.08	0	0.00000	!
0.08	0.09	0	0.00000	!
0.09	0.11	0	0.00000	!
0.11	0.12	0	0.00000	!
0.12	0.14	0	0.00000	!
0.14	0.15	0	0.00000	!
0.15	0.17	0	0.00000	!
0.17	0.18	0	0.00000	!
0.18	0.20	0	0.00000	!
0.20	0.21	0	0.00000	!
0.21	0.23	0	0.00000	!
0.23	0.24	0	0.00000	!
0.24	0.26	0	0.00000	!
0.26	0.27	0	0.00000	!
0.27	0.29	0	0.00000	!
0.29	0.30	0	0.00000	!
0.30	0.32	0	0.00000	!
0.32	0.33	0	0.00000	!
0.33	0.35	0	0.00000	!
0.35	0.36	0	0.00000	!
0.36	0.38	0	0.00000	!
0.38	0.39	0	0.00000	!
0.39	0.41	0	0.00000	!
0.41	0.42	0	0.00000	!
0.42	0.44	0	0.00000	!
0.44	0.45	0	0.00000	!
0.45	0.47	0	0.00000	!
0.47	0.48	0	0.00000	!
0.48	0.50	0	0.00000	!
0.50	0.51	3	0.00600	!
0.51	0.53	73	0.14600	!*****
0.53	0.54	31	0.06200	!*****
0.54	0.56	0	0.00000	!
0.56	0.57	0	0.00000	!
0.57	0.59	0	0.00000	!
0.59	0.60	0	0.00000	!

TOTAL DE REPLICACIONES : 500

ALGORITMO DE "SWAPP" FIFO "

SIMULACION CON 500 INSTRUCCIONES

PAGINAS DE INSTRUCCIONES DEL PROGRAMA : 4

TIEMPO MEDIO DE EJECUCION 0.0928

DESVIACION ESTANDAR DEL TIEMPO DE EJECUCION 0.1768

DISTRIBUCION DE LA EJECUCION TIEMPO/INSTRUCCION

DATOS DEL HISTOGRAMA

RANGO DE	A	FREC.	FREC. RELATIVA
0.00	0.02	98	0.19600 !*****
0.02	0.03	306	0.61200 !*****
0.03	0.05	0	0.00000 !
0.05	0.06	0	0.00000 !
0.06	0.08	0	0.00000 !
0.08	0.09	0	0.00000 !
0.09	0.11	0	0.00000 !
0.11	0.12	0	0.00000 !
0.12	0.14	0	0.00000 !
0.14	0.15	0	0.00000 !
0.15	0.17	0	0.00000 !
0.17	0.18	0	0.00000 !
0.18	0.20	0	0.00000 !
0.20	0.21	0	0.00000 !
0.21	0.23	0	0.00000 !
0.23	0.24	0	0.00000 !
0.24	0.26	0	0.00000 !
0.26	0.27	0	0.00000 !
0.27	0.29	0	0.00000 !
0.29	0.30	0	0.00000 !
0.30	0.32	0	0.00000 !
0.32	0.33	0	0.00000 !
0.33	0.35	0	0.00000 !
0.35	0.36	0	0.00000 !
0.36	0.38	0	0.00000 !
0.38	0.39	0	0.00000 !
0.39	0.41	0	0.00000 !
0.41	0.42	0	0.00000 !
0.42	0.44	0	0.00000 !
0.44	0.45	0	0.00000 !
0.45	0.47	0	0.00000 !
0.47	0.48	0	0.00000 !
0.48	0.50	0	0.00000 !
0.50	0.51	11	0.02200 !**
0.51	0.53	49	0.09800 !*****
0.53	0.54	36	0.07200 !*****
0.54	0.56	0	0.00000 !
0.56	0.57	0	0.00000 !
0.57	0.59	0	0.00000 !
0.59	0.60	0	0.00000 !

TOTAL DE REPLICACIONES : 500

Interpretación de los resultados

Se corrió el programa variando el número de páginas por programa. Los resultados de la simulación fueron los siguientes:

Número de Páginas	ALEATORIO		FIFO	
	tiempo medio	desviación estándar	tiempo medio	desviación estándar
2	0.026	0.0546	0.022	0.0317
3	0.082	0.1653	0.076	0.1570
4	0.092	0.1766	0.092	0.1768
5	0.149	0.2191	0.1533	0.2221

Aparentemente el algoritmo aleatorio es mejor que el FIFO para programas de no más de cuatro páginas.

Por otra parte, parece que en un programa de más de 5 páginas el FIFO es más adecuado.

Haciendo una prueba estadística de los resultados comprobaremos los resultados obtenidos.

Para el programa de paginación aleatoria:

$$\bar{T}_A = 0.149 \text{ seg. (5 pags.)}; = 0.0261 \text{ seg. (2 pags)}$$

$$S_A = 0.2191 \text{ seg. (5 pags)}; = 0.0546 \text{ seg. (2 pags)}$$

Para el algoritmo de paginación FIFO:

$$\bar{T}_F = 0.1533 \text{ seg. (5 pags.)}; = 0.0220 \text{ seg. (2 pags.)}$$

$$S_F = 0.2221 \text{ seg. (5 pags.)}; = 0.0317 \text{ seg. (2 pags.)}$$

De estas estimaciones, aparentemente el algoritmo aleatorio es el más adecuado para programas de 5 páginas, y lo contrario para el caso en que son 2 páginas. Sin embargo tendremos que llevar a cabo una prueba de hipótesis para verificar si esta diferencia es significativa.

Para programas de 5 páginas:

$$\text{Hipótesis } H_0: \mu_F > \mu_A \quad ; \quad H_1: \mu_F < \mu_A$$

Criterio de decisión: Se rechaza H_0 si y solo si $Z > 1.96$ (5%).

Cálculos:

$$\text{Sigma}_{AX} = \sqrt{\frac{S_F}{N_A} + \frac{S_A}{N_A}} = \sqrt{\frac{0.2221}{500} + \frac{0.219}{500}} = 0.0297$$

$$Z = \frac{0.1533 - 0.149}{0.0297} = 0.145$$

Por tanto, no podemos aceptar H_0 , quizás tendríamos que tomar una muestra más grande pa-

ra demostrar que no la podemos rechazar.

Para programas de 2 páginas:

Hipótesis $H_0: \mu_A \geq \mu_F$; $H_1: \mu_A < \mu_F$

Criterio de decisión: Se rechaza H_0 si y solo si $Z > 1.96$ (5%).

Cálculos:

$$\sigma_{Ax} = \sqrt{\frac{SF}{NR} + \frac{SA}{NR}} = \sqrt{\frac{0.0317}{500} + \frac{0.054}{500}} = 0.0131$$

$$Z = \frac{0.0261 - 0.022}{0.0131} = 0.312.$$

Por tanto, tampoco podemos aceptar H_0 , quizás tendríamos que tomar de nuevo una muestra más grande para demostrar que no la podemos rechazar.

de respuesta.

Ahora, queremos desarrollar una relación entre el tiempo de respuesta para el j -ésimo trabajo terminado (TR) y el tiempo de "CPU" para este trabajo (TP) usando análisis de regresión.

Así, de los datos de la simulación tenemos:

$$\bar{T}_1 = 0.8 \quad (\text{Media del tiempo requerido})$$

$$S_1 = 0.8 \quad (\text{Desviación estándar del tiempo requerido})$$

$$\bar{T}_2 = 8.0827 \quad (\text{Media del tiempo de respuesta})$$

$$S_2 = 9.6087 \quad (\text{Desviación estándar del tiempo de respuesta})$$

Así, la relación de la regresión es:

$$\frac{TR - 8.0827}{9.6087} = (0.8968) \frac{TP - 0.8}{0.8}$$

o bien:

$$TR = 10.77 (TP) - 0.534$$

Esta ecuación expresa el tiempo de respuesta de un trabajo en función de TP (tipo exponencial).

Referencias

- 6.1 Gross, Donald, and Harris, Carl.
Fundamentals of Queueing theory.
New York : John Wiley and Sons, 1974.
- 6.2 Little, J.D.C. "A proof the queueing
formula $L = \lambda W$."
Oper. Res. 16, (1961) : 651-665
- 6.3 Hansen, P. Operating system principles
Prentice-hall, Inc., Englewood Cliffs, New Jersey.

CAPITULO VII

7.1 Conclusiones

Con la terminación de esta tesis tenemos una serie de conclusiones las cuales citamos a continuación.

La simulación requiere de un gran acervo de conocimientos sobre probabilidad, estadística, diseño de modelos y sobre todo del área de aplicación, para poder ser utilizada con éxito.

La simulación puede llegar a resolver problemas muy complejos que no podrían resolverse analíticamente. La computadora nos sirve para hacer grandes simulaciones que podrían ser muy tediosas e incosteables si se realizan manualmente.

La simulación puede dar a conocer soluciones no exactas, pero si muy aproximadas a la realidad, o en su caso, lo necesariamente aproximadas, de acuerdo a la exactitud requerida por el analista.

La simulación es un medio de resolver problemas que puede llegar a ser muy costoso, debido a la necesidad de contar con un computador en muchos casos y realizar varias o inclusive muchas corridas para poder obtener los resultados deseados.

Siempre que vaya a ser utilizada la simulación como alternativa para la solución de problemas, es necesario conocer la mejor

manera de resolverlos antes de llevar a cabo cualquier acción; esto es, es conveniente checar si ya existen los programas o paquetes que puedan resolver nuestro problema y si se puede tener disposición a ellos. Lo mismo podemos decir de los diversos y numerosos paquetes de simulación de uso general que ya existen. Esto último nos podría evitar el realizar un paquete de simulación para nuestro propósito particular, pero por otra parte, nos podría limitar el tipo de aplicaciones a las que estén enfocados.

Las subrutinas integradas en la programoteca de esta tesis pueden utilizarse en otro tipo de aplicaciones, ya que son de tipo general.

Para adoptar los programas de simulación de este documento, debe realizarse una evaluación técnica y económica para poder determinar si es conveniente adaptar estos programas de acuerdo a los objetivos de su proyecto.

Relacionado con estas conclusiones tenemos una serie de posibles extensiones o complementos al trabajo de esta tesis. En la sección siguiente citamos lo anterior.

7.2 Extensiones, complementos y trabajos relacionados

Surgen varias extensiones a este trabajo entre otras tenemos:

* Buscar soluciones analíticas para problemas típicos del área de computación. El tipo de problemas que aparecen por ejemplo en sistemas operativos, transmisión de datos, diseño de hardware y otras partes de la computación y que tienen cierta carga aleatoria, generalmente aparecen sin solución analítica.

* Usar lo desarrollado en el presente trabajo para simular otros sistemas de mayor complejidad.

* Un trabajo más ambicioso consistiría en el diseño y construcción de un lenguaje con orientación a la simulación de sistemas discretos.

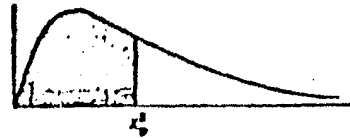
Cada una de estas extensiones tiene su grado de complejidad además de exigir una formación sólida en el área respectiva.

Termino este trabajo esperando que algún trabajo nuevo de tesis se encamine en alguno de los sentidos señalados anteriormente.

APENDICES

APENDICE I

PERCENTILES (χ^2_p)
DE LA
DISTRIBUCION CHI-CUADRADO
CON ν GRADOS DE LIBERTAD
(AREA SOMBREADA = p)



ν	$\chi^2_{0.995}$	$\chi^2_{0.99}$	$\chi^2_{0.975}$	$\chi^2_{0.95}$	$\chi^2_{0.90}$	$\chi^2_{0.85}$	$\chi^2_{0.80}$	$\chi^2_{0.75}$	$\chi^2_{0.70}$	$\chi^2_{0.65}$	$\chi^2_{0.60}$	$\chi^2_{0.55}$	$\chi^2_{0.50}$	$\chi^2_{0.45}$	$\chi^2_{0.40}$
1	7.88	6.63	5.02	3.84	2.71	1.32	0.455	0.102	0.0158	0.0039	0.0010	0.0002	0.0000		
2	10.6	9.21	7.38	5.99	4.61	2.77	1.39	0.575	0.211	0.103	0.0546	0.0201	0.0100		
3	12.8	11.3	9.35	7.81	6.25	4.11	2.37	1.21	0.554	0.352	0.216	0.115	0.072		
4	14.9	13.3	11.1	9.49	7.78	5.39	3.36	1.92	1.06	0.711	0.484	0.297	0.207		
5	16.7	15.1	12.8	11.1	9.24	6.63	4.35	2.67	1.61	1.15	0.831	0.554	0.412		
6	18.5	16.8	14.4	12.6	10.6	7.84	5.35	3.45	2.29	1.64	1.24	0.872	0.676		
7	20.3	18.5	16.0	14.1	12.0	9.04	6.35	4.25	2.83	2.17	1.69	1.24	0.989		
8	22.0	20.1	17.5	15.5	13.4	10.2	7.34	5.07	3.49	2.73	2.18	1.65	1.24		
9	23.6	21.7	19.0	16.9	14.7	11.4	8.34	5.90	4.17	3.33	2.70	2.09	1.73		
10	25.2	23.2	20.5	18.3	16.0	12.5	9.34	6.74	4.87	3.94	3.25	2.56	2.16		
11	26.8	24.7	21.9	19.7	17.3	13.7	10.3	7.58	5.58	4.57	3.82	3.05	2.60		
12	28.3	26.2	23.3	21.0	18.5	14.8	11.3	8.44	6.39	5.23	4.40	3.57	3.07		
13	29.8	27.7	24.7	22.4	19.8	16.0	12.3	9.30	7.04	5.89	5.01	4.11	3.57		
14	31.3	29.1	26.1	23.7	21.1	17.1	13.3	10.2	7.79	6.57	5.63	4.66	4.07		
15	32.8	30.6	27.5	25.0	22.3	18.2	14.3	11.0	8.55	7.26	6.26	5.23	4.60		
16	34.3	32.0	28.8	26.3	23.5	19.4	15.3	11.9	9.31	7.96	6.91	5.81	5.14		
17	35.7	33.4	30.2	27.6	24.8	20.5	16.3	12.8	10.1	8.67	7.56	6.41	5.70		
18	37.2	34.8	31.5	28.9	26.0	21.6	17.3	13.7	10.9	9.39	8.21	7.01	6.26		
19	38.6	36.2	32.9	30.1	27.2	22.7	18.3	14.6	11.7	10.1	8.91	7.63	6.84		
20	40.0	37.6	34.2	31.4	28.4	23.8	19.3	15.5	12.4	10.9	9.59	8.26	7.43		
21	41.4	38.9	35.5	32.7	29.6	24.9	20.3	16.3	13.2	11.6	10.3	8.90	8.03		
22	42.8	40.3	36.8	33.9	30.8	26.0	21.3	17.2	14.0	12.3	11.0	9.54	8.64		
23	44.2	41.6	38.1	35.2	32.0	27.1	22.3	18.1	14.8	13.1	11.7	10.2	9.26		
24	45.6	43.0	39.4	36.4	33.2	28.2	23.3	19.0	15.7	13.8	12.4	10.9	9.89		
25	46.9	44.3	40.6	37.7	34.4	29.3	24.3	19.9	16.5	14.6	13.1	11.5	10.5		
26	48.3	45.6	41.9	38.9	35.6	30.4	25.3	20.8	17.3	15.4	13.8	12.2	11.2		
27	49.6	47.0	43.2	40.1	36.7	31.5	26.3	21.7	18.1	16.2	14.6	12.9	11.8		
28	51.0	48.3	44.5	41.3	37.9	32.6	27.3	22.7	18.9	16.9	15.3	13.6	12.5		
29	52.3	49.6	45.7	42.6	39.1	33.7	28.3	23.6	19.8	17.7	16.0	14.3	13.1		
30	53.7	50.9	47.0	43.8	40.3	34.8	29.3	24.5	20.6	18.5	16.8	15.0	13.8		
40	66.8	63.7	59.3	55.8	51.8	45.6	39.2	33.7	29.1	26.5	24.4	22.2	20.7		
50	79.5	76.2	71.4	67.5	63.2	56.3	49.3	42.9	37.7	34.8	32.4	29.7	28.0		
60	92.0	88.4	83.3	79.1	74.4	67.0	59.3	52.3	46.5	43.2	40.5	37.5	35.5		
70	104.2	100.4	95.0	90.5	85.5	77.6	69.3	61.7	55.3	51.7	48.8	45.4	43.3		
80	166.3	112.3	106.6	101.9	96.6	83.1	79.3	71.1	64.3	60.4	57.2	53.5	51.2		
90	128.3	124.1	118.1	113.1	107.6	98.6	89.3	80.6	73.3	69.1	65.6	61.8	59.2		
100	140.2	135.8	129.6	124.3	118.5	109.1	99.3	90.1	82.4	77.9	74.2	70.1	67.3		

Apéndice II

```
C
C
C          SUBROUTINA "RAND"
C
C          ESTA SUBROUTINA GENERA UN NUMERO ALEATORIO (RAN) UNIFORMEMENTE
C  DISTRIBUIDO ENTRE "0" Y "1" USANDO EL METODO CONGRUENCIAL MULTIPLI-
C  CATIVO
C
C          SUBROUTINE RAND(NSEED,RAN)
C
C          EL OVERFLOW HACE EL MODULO,
C          DEBE COMPILARSE CON OPCION /NOCHECK PARA QUE NO CHEQUE
C          DESBORDAMIENTO.
C
C          NSEED=IABS(5*NSEED)
C          RAN=NSEED
C          RAN=RAN/32767.
C          RETURN
C          END
C
C
C          EL PROPOSITO DEL CODIGO SIGUIENTE ES GENERAR DATOS USANDO EL GE-
C  NERADOR DE NUMEROS ALEATORIOS "RAND" Y APLICAR PRUEBAS ESTADISTICAS ,
C  PARA TENER CONFIANZA AL USARLO DURANTE EL PRESENTE TRABAJO.
C
C          DIMENSION X(10000),K(20)
C          N=10000
C          NSEED=5555
C          NSAVE=NSEED
C
C          "GENERACION DE LA SECUENCIA"
C
C          DO 20 I=1,N
C          CALL RAND(NSEED,R)
20      X(I)=R
C
C          GENERACION DE ESTADISTICAS USANDO LAS SUBROUTINAS DEL APENDICE
C  IV.
C
C          CALL STAT(N,X,XM,XDS)
C          CALL AUTO(N,1,X,XM,XDS,XAC1)
C          CALL AUTO(N,2,X,XM,XDS,XAC2)
C          CALL AUTO(N,3,X,XM,XDS,XAC3)
C
C          PARTE EN 20 EL INTERVALO UNITARIO Y CALCULA LA ESTADISTICA
C  CHI-CUADRADA
C
C          DO 30 I=1,20
30      K(I)=0
C          DO 50 J=1,N
C          I=20.0*X(J)+1.0
50      K(I)=K(I)+1
```

```
SUM=0.0
RN=N
DO 60 I=1,20
RK=K(I)
SUM=SUM+20.0*(RK-0.05*RN)*(RK-0.05*RN)/RN
60
C
E
70
WRITE(5,70) NSAVE
FORMAT(/,5X,'PROGRAMA DE PRUEBA PARA LA SUBROUTINA RAND',/,
1 ' CON NSEED = ',I8,/)
WRITE(5,75) N
75
FORMAT(/,2X,I7,' NUMEROS PSEUDOALEATORIOS FUERON',1X,
1 'GENERADOS POR "RAND" ',/,11X,' CON LOS SIGUIENTES
1 RESULTADOS :',/)
WRITE(5,80) XM,XDS
80
FORMAT(' LA MEDIA ES ',F7.5,/,/, ' LA DESVIACION ESTANDAR ES '
1 ,F7.5,/)
WRITE(5,85) XAC1,XAC2,XAC3
85
FORMAT(' LA CORRELACION DE RETARDO 1 ES ',F7.5,/,/,21X,
1 'RETARDO 2 ES ',F7.5,/,/,21X,'RETARDO 3 ES ',F7.5,/)
CALL HISTO(N,X,0.0,1.0,20)
WRITE(5,90) SUM
90
FORMAT(/,/, ' LA ESTADISTICA CHI CUADRADA, ASUMIENDO
1 DISTRIBUCION UNIFORME',/,/, ' EN EL INTERVALO UNITARIO',
1 ' ES ',F8.4)
END
C
```


PROGRAMA DE PRUEBA PARA LA SUBROUTINA RAND
CON NSEED = 5555

10000 NUMEROS PSEUDOALEATORIOS FUERON GENERADOS POR "RAND"
CON LOS SIGUIENTES RESULTADOS :

LA MEDIA ES 0.49199

LA DESVIACION ESTANDAR ES 0.28187

LA CORRELACION DE RETARDO 1 ES 0.18110

RETARDO 2 ES 0.09214

RETARDO 3 ES 0.14601

DATOS DEL HISTOGRAMA

RANGO		FREC.	FREC. RELATIVA
DE	A		
0.00	0.05	469	0.04690 !****
0.05	0.10	469	0.04690 !****
0.10	0.15	625	0.06250 !*****
0.15	0.20	469	0.04690 !****
0.20	0.25	468	0.04680 !****
0.25	0.30	469	0.04690 !****
0.30	0.35	625	0.06250 !*****
0.35	0.40	469	0.04690 !****
0.40	0.45	468	0.04680 !****
0.45	0.50	627	0.06270 !*****
0.50	0.55	469	0.04690 !****
0.55	0.60	469	0.04690 !****
0.60	0.65	470	0.04700 !****
0.65	0.70	624	0.06240 !*****
0.70	0.75	468	0.04680 !****
0.75	0.80	468	0.04680 !****
0.80	0.85	468	0.04680 !****
0.85	0.90	625	0.06250 !*****
0.90	0.95	469	0.04690 !****
0.95	1.00	312	0.03120 !***

TOTAL DE REPLICACIONES :10000

LA ESTADISTICA CHI CUADRADA, ASUMIENDO DISTRIBUCION UNIFORME
EN EL INTERVALO UNITARIO ES 13.7720

Interpretación de los resultados

La media de los resultados es muy aproximada a $\frac{1}{2}$, lo cual nos indica que es muy cercana a la real. Lo mismo ocurre con la desviación estándar.

La autocorrelación en cada uno de los tres primeros retardos es prácticamente cero, lo cual nos indica que existe una regla o una tendencia o sesgo de generación, lo cual como ya hemos visto es una cualidad deseable y necesaria.

Como la Chi-Cuadrada es menor que Chi-Cuadrada $(0.95) = 30.1$ (ver apéndice I), entonces no podemos afirmar que lo observado no pertenezca a la población de lo esperado, esto con una probabilidad de equivocarnos del 5%. Por tanto, aceptamos los resultados como verdaderos.

ANEXO DE LA LEY DE ENFERMERIA Y CALIFICACIONES
DE LOS ENFERMOS EN ESPAÑA

Los parámetros de entrada están señalados con una anotación entre paréntesis. Se lo controla en la lista de parámetros de entrada.

La forma de entrada de los parámetros es libre con la única restricción de diferenciar los parámetros enteros de los decimales. La clave para poder diferenciarlos está en la primera letra de la literal que representa el valor de dicho parámetro. Si la letra comienza con las letras I, J, K, L, M o N entonces se trata de parámetros enteros incluidos dentro del rango especificado. De lo contrario, se trata de un parámetro decimal.

RUTINAS QUE PRODUCEN DISTRIBUCIONES ESTADÍSTICAS

BERN (BERN, N, P)

BERN = Señala. Su valor puede ser 5015 o alguno otro como se especifica en el código de las señillas a que se refiere a continuación y a las mismas restricciones.
N = Número aleatorio entero (0 a 1) obtenido en distribución de Bernoulli (válida)
P = Parámetro de Bernoulli

Características:

Tipo de Distribución: Discreta
Rango: 0, 1
Parámetro: P, $0 < P < 1$
Media: P
Varianza: $P(1-P)$

Uso: Determina la ocurrencia o no ocurrencia de un evento distribuido de acuerdo a la distribución de probabilidad de Bernoulli.

BINOM (BINOM, N, P)

BINOM = Señala
N = Número aleatorio obtenido en distribución Binomial (válida)
P = Parámetro Binomial

Características:

Tipo de Distribución: Discreta, con rango $0, 1, 2, \dots, n$
Parámetros: n > entero
Rango: $0 < P < 1$
Media: NP
Varianza: $NP(1-P)$

Uso: Se utiliza en base al número de éxitos realizados en n intentos independientes de éxito o falla con la misma

ERLANG (NSEED, RAN, A, K)

ERLANG (NSEED, RAN, A, K)

NSEED = Semilla
RAN = Número aleatorio producido en distribución Erlang (valido)
A y K = Parámetros ERLANG

Características:

Tipo de Distribución: Continua
Rango: De cero a oo (infinito)
Parámetros: A y K, A > 0
K > 1, K es entero
Varianza: $K(K-1/2)$

Uso: Esta distribución se deriva de la suma de variables aleatorias independientes e idénticamente distribuidas con "media" uno. Es usada extensamente en la teoría de colas cuando una actividad o tiempo de espera se considera que ocurre en "K" etapas cada fase que es igualmente distribuida. También es utilizada para cierto tipo de tráfico telefónico en tiempos de espera.

EXFON (NSEED, RAN, RNEDIA)

NSEED = Semilla
RAN = Número aleatorio producido en distribución exponencial (valido)
RNEDIA = Media

Características:

Tipo de Distribución: Continua
Rango: De cero a oo
Varianza: Media al cuadrado.

Uso: La actividad caracterizada por esta distribución pide que la probabilidad que tiene un evento de ocurrir después de otro que haya ocurrido en un intervalo de tiempo $DELTA$ t es la misma. Así, si una actividad ha estado presente durante t unidades de tiempo, la probabilidad de que termine en el siguiente intervalo $DELTA$ t es la misma que si tal actividad hubiera de comenzar.

Esta distribución posee una gran versatilidad.

GEO (NSEED, NRAN, P)

NSEED = Semilla
NRAN = Número aleatorio en distribución Geométrica (valido)
P = Parámetro Geométrico

Características:

Tipo de Distribución: Discreta
Rango: 0, 1, 2, ... oo

Parámetros: 5 P K 1
Media: $(1 - P) / P$
Varianza: $(1 + P) / P^2$

NORMAL (NSEED, RAN, RMEJIA, RDS)

NSEED = Semilla
RAN = Número aleatorio producido en distribución Normal (salida)
RMEJIA = Media
RDS = Desviación estándar

Características:

Tipo de Distribución: Continua
Rango: $-\infty < X < \infty$
Parámetros: RMEJIA, RDS sin restricciones

Uso: Esta distribución describe a la mayoría de los fenómenos de medición, tales como las alturas de hombres, mujeres y niños, el diámetro de un tubo, o la longitud de una línea, etc.

POISSON (NSEED, RAN, RMEJIA)

NSEED = Semilla
RAN = Número aleatorio entero producido en distribución de Poisson (salida)
RMEJIA = Media

Características:

Tipo de Distribución: Discreta
Rango: $0, 1, 2, \dots, \infty$
Parámetro: RMEJIA: media
Varianza: RMEJIA

Uso: Esta distribución se usa generalmente para generar el número de resultados ocurridos en un periodo de tiempo específico. Si la duración del tiempo entre resultados está exponencialmente distribuida y ellos ocurren uno a la vez, entonces el número de ocurrencias en un intervalo de tiempo fijo puede ser modelado por la distribución de Poisson. Así, si la distribución entre eventos es exponencial, el número de eventos estará distribuido según esta distribución.

RAND (NSEED, RAN)

NSEED = Semilla para generación del número aleatorio
RAN = Número aleatorio producido en una selección blanca en un intervalo de 0 a 1 (salida)

UNIFAN (NSEED, RAN, A, B)

NSEED = Semilla
RAN = Número aleatorio en distribución Uniforme (salida)
A = Límite inferior del rango aleatorio deseado
B = Límite superior del rango aleatorio deseado

Características:

Tipo de Distribución: Continua
Rango: de A a B
Parámetros: A sin restricción
B > A
Media: $(A + B)/2$
Varianza: $((B-A)^2)/12$

Uso: El uso de esta distribución a menudo implica una completa falta de conocimiento de la variable aleatoria concerniente, únicamente que se encuentra entre un valor mínimo y otro máximo.

APUNTA DE PROGRAMAS ESTADÍSTICOS

AUTO (N, NR, X, XM, XMS, XAC)

N = Tamaño del arreglo de datos de los valores de una variable física medible en su computador.
 NR = Fecha del auto en la edición (entero)
 - NR = 1 es el primer auto, NR = 2 es el segundo, etc.
 X = Nombre del arreglo de datos que se va a calcular su autocorrelación.
 XM = Media
 XMS = Desviación estándar
 XAC = Coeficiente de auto correlación (salida)

COVAR (N, X1, X2, XM1, XMS1, XM2, XMS2, XCC)

N = Tamaño del arreglo de los variables a calcular su correlación
 X = Nombre del arreglo de la variable 1
 X2 = Nombre del arreglo de la variable 2
 XM1 = Media del arreglo de la variable 1
 XM2 = Media del arreglo de la variable 2
 XMS1 = Desviación estándar del arreglo de la variable 1
 XMS2 = Desviación estándar del arreglo de la variable 2
 XCC = Correlación entre las variables (salida)

HIST (N, X, XME, XMA, NI)

N = Tamaño del arreglo
 X = Nombre del arreglo
 XME = Límite menor del histograma
 XMA = Límite mayor del histograma
 NI = Número de intervalos del histograma

DHIST (N, X, XME, XMA, NV)

N = Tamaño del arreglo
 X = Nombre del arreglo
 XME = Límite menor del histograma
 XMA = Límite mayor
 NV = Número de valores discretos entre XME y XMA

STAT (N, X, XM, XMS)

N = Tamaño del arreglo de datos a calcular
 X = Nombre del arreglo
 XM = Media Calculada (salida)
 XMS = Desviación estándar (salida)

Apendice IV

Rutinas estadísticas

```
C      SUBROUTINE STAT(N,X, XM,XDS)
C      ESTA RUTINA CALCULA LA MEDIA "XM" Y LA DESVIACION ESTANDAR "XDS"
C      PARA LOS "N" VALORES EN EL ARREGLO "X"
C
C      DIMENSION X(N)
C
C      CALCULA LA MEDIA
C
C      SUMM=0.0
C      DO 5 I=1,N
5      SUMM=SUMM+X(I)
C      XM=SUMM/FLOAT(N)
C
C      CALCULA LA DESVIACION ESTANDAR
C
C      SUMV=0.0
C      DO 10 I=1,N
10     SUMV=SUMV+(X(I)-XM)*(X(I)-XM)
C      XDS=SQRT(SUMV/FLOAT(N-1))
C      RETURN
C      END
C
C      SUBROUTINE AUTO(N,NR,X, XM,XDS,XAC)
C      ESTA RUTINA CALCULA EL COEFICIENTE DE AUTOCORRELACION "XAC"
C      PARA LOS DATOS ALMACENADOS EN EL ARREGLO "X" EL RETARDO ES "NR"
C
C      DIMENSION X(N)
C      SUM=0.0
C      NNR=N-NR
C      DO 40 I=1,NNR
40     IR=I+NR
C      SUM=SUM+(X(I)-XM)*(X(IR)-XM)
C      XAC=SUM/(FLOAT(NNR-1)*XDS*XDS)
C      RETURN
C      END
C
C      SUBROUTINE CORR (N,X1,X2,X1M,X1DS,X2M,X2DS,XCC)
C      ESTA RUTINA CALCULA EL COEFICIENTE DE CORRELACION "XCC"
C      PARA DATOS EN LOS ARREGLOS "X1" ,"X2"
C
C      DIMENSION X1(N),X2(N)
C      SUM=0.0
C      DO 40 I=1,N
40     SUM=SUM+(X1(I)-X1M)*(X2(I)-X2M)
C      XCC=SUM/(FLOAT(N-1)*X1DS*X2DS)
C      RETURN
C      END
C
C      SUBROUTINE HISTO(N,X,XME,XMA,NI)
```


C
C ESTA RUTINA PRODUCE UN HISTOGRAMA DE LOS DATOS DE ENTRADA "X"
C USANDO INTERVALOS IGUALMENTE ESPACIADOS "NI" ENTRE "XME" Y "XMA"

C
C
C PARAMETROS DE ENTRADA

C N,- NUMERO DE PUNTOS DE DATOS
C X,- ARREGLO DE LOS VALORES DE DATOS
C XME,- LIMITE INFERIOR DEL HISTOGRAMA
C XMA,- LIMITE SUPERIOR DEL HISTOGRAMA
C NI,- NUMERO DE INTERVALOS DEL HISTOGRAMA

C
C DIMENSION X(N),KONT(40)
C LOGICAL*1 AST(100)
C DO 99 I=1,100
C AST(I)='*'
99 CONTINUE

C
C RN=N
C RNI=NI
C DELT=(XMA-XME)/RNI
C DO 10 I=1,NI
10 KONT(I)=0
C DO 90 I=1,N
C IF (X(I) .LE. XME) GOTO 50
C IF (X(I) .GT. XMA) GOTO 70

C
C INT=(X(I)-XME)/DELT +1.0
C KONT(INT)=KONT(INT) +1
C GOTO 90
50 KONT(1)=KONT(1)+1
C GOTO 90
70 KONT(NI)=KONT(NI)+1
C GOTO 90
90 CONTINUE

C
C SALIDA
C WRITE(5,1000)
1000 FORMAT(/,20X,'DATOS DEL HISTOGRAMA'//,6X,'RANGO',6X,'FREC.',
C 1 4X,'FREC. RELATIVA',/,4X,'DE',6X,'A'//)

C CALCULA LAS FRECUENCIAS RELATIVAS
C DO 150 J=1,NI
C RK=KONT(I)
C RF=RK/RN
C RI=I
C RII=I-1
C RX=XME+RI*DELT
C RXI=XME+RII*DELT

C
C CALCULO DEL NUMERO DE ASTERISCOS A IMPRIMIR

C IRF=RF*100. + .1
C IF (IRF .NE. 0) GOTO 140

```
1010 WRITE(5,1010)RX1,RX,KONT(I),RF
      FORMAT(1X,F6.2,1X,F6.2,2X,I4,3X,F7.5,' !')
      GOTO 150

C
C      IMPRIME HISTOGRAMA
140   WRITE(5,1011)RX1,RX,KONT(I),RF,(AST(J),J=1,IRF)
1011  FORMAT(1X,F6.2,1X,F6.2,2X,I4,3X,F7.5,' !',100A1)
150   CONTINUE
      WRITE(5,1040) N
1040  FORMAT(/,' TOTAL DE REPLICACIONES :',I5)
      RETURN
      END

C
C
```

Apendice V

```
C
C   PROGRAMOTECA DE VARIABLES ALEATORIAS
C
C   SUBROUTINE UNIFRM(NSEED,RAN,A,B)
C
C   ESTA SUBROUTINA GENERA UN NUMERO ALEATORIO UNIFORMEMENTE
C   DISTRIBUIDO ENTRE A Y B.
C
C   CALL RAND(NSEED,RAN)
C   RAN=A+(B-A)*RAN
C   RETURN
C   END
C
C   SUBROUTINE BERN(NSEED,NRAN,P)
C
C   ESTA SUBROUTINA GENERA UN NUMERO ENTERO ALEATORIO "NRAN"
C   DESDE UNA DISTRIBUCION DE BERNOULLI CON PARAMETRO "P"
C
C   NRAN=0
C   CALL RAND (NSEED,P)
C   IF (R .LE. P ) NRAN=1
C   RETURN
C   END
C
C   SUBROUTINE BINOM(NSEED,NRAN,N,P)
C
C   ESTA SUBROUTINA GENERA UN NUMERO ALEATORIO "NRAN"
C   DESDE UNA BINOMIAL CON PARAMETROS "N" Y "P"
C
C   NRAN=0
C   DO 6 I= 1,N
C   CALL RAND(NSEED,R)
C   IF (R .LE. P ) NRAN=NRAN+1
6   CONTINUE
C   RETURN
C   END
C
C   SUBROUTINE ERLANG(NSEED,RAN,A,K)
C
C   ESTA SUBROUTINA GENERA UN NUMERO ALEATORIO "RAN"
C   DESDE UNA ERLANG CON PARAMETROS "A" Y "K"
C
C   FR=1.0
C   DO 4 I=1,K
C   CALL RAND(NSEED,RAN)
4   FR=FR*RAN
C   RAN=-ALOG(FR)/A
C   RETURN
C   END
C
```

```
C
C
C     ESTA SUBROUTINA GENERA UN NUMERO ALEATORIO "RANA"
C     DESDE UNA EXPONENCIAL CON MEDIA "RMEDIA"
C
C     CALL RAND(NSEED,RAN)
C     RANA=-RMEDIA*ALOG(RAN)
C     RETURN
C     END
C
C
C     SUBROUTINE GEO(NSEED,NRAN,P)
C
C     ESTA SUBROUTINA GENERA UN NUMERO ENTERO ALEATORIO "NRAN"
C     DESDE UNA GEOMETRICA CON PARAMETRO "P"
C
C     NRAN=0
C     CALL RAND(NSEED,R)
C     IF (R .GT. P) RETURN
C     NRAN=NRAN+1
C     GOTO 2
C     END
C
C
C     SUBROUTINE NORMAL(NSEED,RAN,RMEDIA,RDS)
C
C     ESTA SUBROUTINA GENERA UN NUMERO ALEATORIO "RAN"
C     NORMALMENTE DISTRIBUIDO CON MEDIA "RMEDIA" Y
C     DESVIACION ESTANDAR "RDS" USANDO UNA APROXIMACION
C     BASADA EN EL TEOREMA CENTRAL DEL LIMITE
C
C     SUM=0.0
C     DO 4 I=1,12
C     CALL RAND(NSEED,RAN)
C     SUM=SUM+RAN
C     RAN=RDS*(SUM-.6.)+RMEDIA
C     RETURN
C     END
C
C
C     SUBROUTINE POISSN(NSEED,NRAN,RMEDIA)
C
C     ESTA SUBROUTINA GENERA UN NUMERO ENTERO ALEATORIO "NRAN"
C     DESDE UNA POISSON CON MEDIA "RMEDIA"
C
C     NRAN=0
C     E=EXP(-RMEDIA)
C     PR=1.0
C     CALL RAND(NSEED,R)
C     PR=PR*R
C     IF ((PR-E) .LT. 0.0) RETURN
C     NRAN=NRAN+1
C     END
```