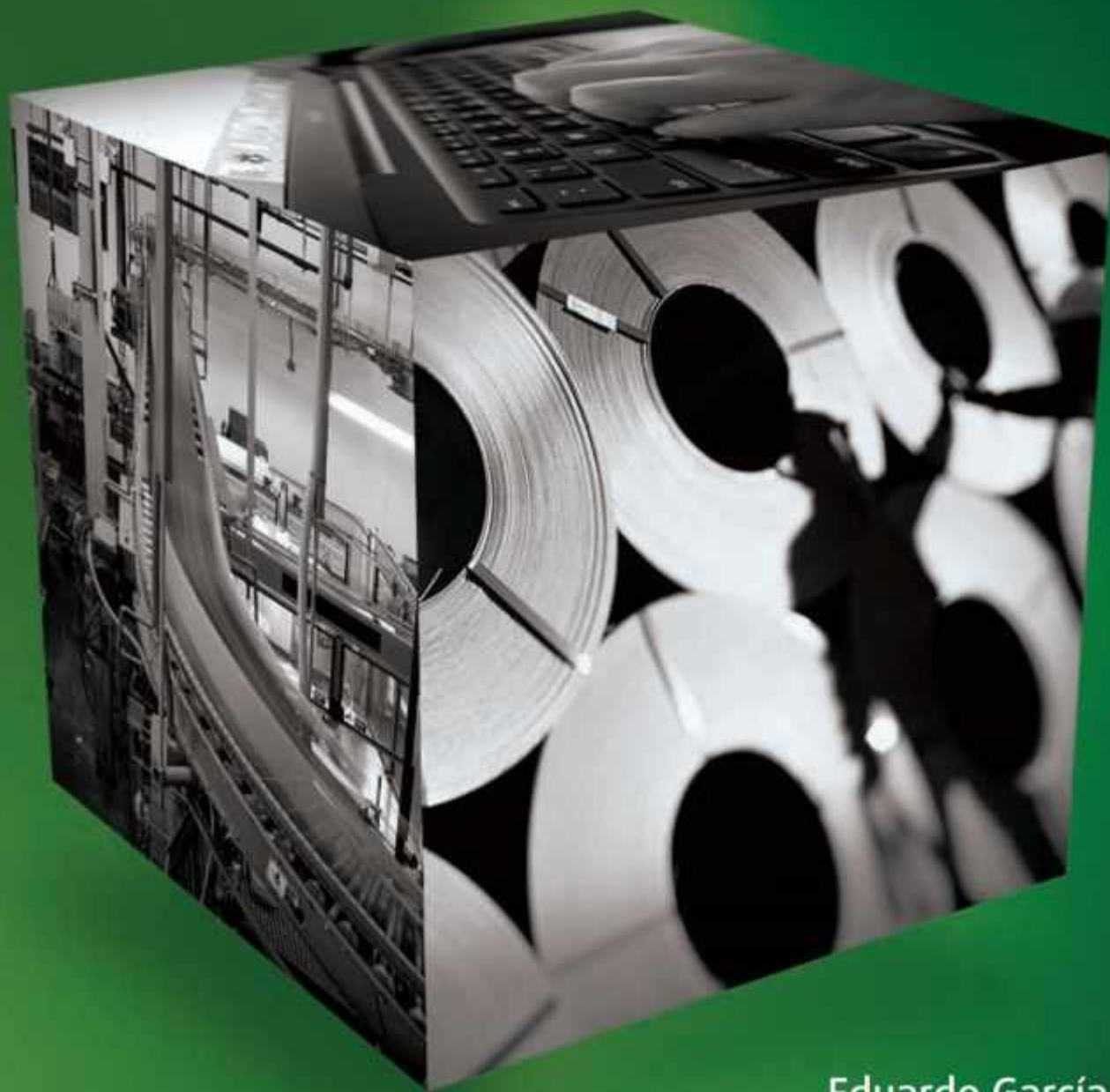


Simulación

y análisis de sistemas con

ProModel[®]

2^a edición



Eduardo García Dunna
Heriberto García Reyes
Leopoldo E. Cárdenas Barrón

Simulación y análisis de sistemas con ProModel

Simulación y análisis de sistemas con ProModel

Segunda edición

Eduardo García Dunna
Heriberto García Reyes
Leopoldo E. Cárdenas Barrón

Tecnológico de Monterrey

REVISIÓN TÉCNICA

Luis Ernesto Mancilla Espinosa

División de Estudios de Posgrado e Investigación

Instituto Tecnológico de León

Victoria Álvarez Ureña

Departamento de Ingeniería Industrial

Centro Universitario de Ciencias Exactas e Ingenierías - CUCEI

Universidad de Guadalajara

Liliana Yadira Castellanos López

Departamento de Ingeniería Industrial

Instituto Tecnológico Superior del

Occidente del Estado de Hidalgo

PEARSON

**GARCÍA DUNNA, EDUARDO;
GARCÍA REYES, HERIBERTO
y CÁRDENAS BARRÓN, LEOPOLDO E.**

**Simulación y análisis de sistemas con
ProModel. Segunda edición**

PEARSON, México, 2013

ISBN: 978-607-32-1511-4
Área: Ingeniería

Formato: 18.5 × 23.5 cm

Páginas: 360

Edición en español

Director Educación Superior: Mario Contreras
Editor Sponsor: Luis Miguel Cruz Castillo
e-mail: luis.cruz@pearson.com
Editor de Desarrollo: Bernardino Gutiérrez Hernández
Supervisor de Producción: Rodrigo Romero Villalobos
Gerente Editorial Educación
Superior Latinoamérica: Marisa de Anta

SEGUNDA EDICIÓN, 2013

D.R. © 2013 por Pearson Educación de México, S.A. de C.V.
Atacomulco 500-5° Piso
Col. Industrial Atoto, C.P. 53519
Naucalpan de Juárez, Estado de México
e-mail: editorial.universidades@pearsoned.com

Cámara Nacional de la Industria Editorial Mexicana. Reg. núm. 1031.

Reservados todos los derechos. Ni la totalidad ni parte de esta publicación pueden reproducirse, registrarse o transmitirse, por un sistema de recuperación de información, en ninguna forma ni por ningún medio, sea electrónico, mecánico, fotoquímico, magnético o electroóptico, por fotocopia, grabación o cualquier otro, sin permiso previo por escrito de los coeditores.

El préstamo, alquiler o cualquier otra forma de cesión de uso de este ejemplar requerirá también la autorización de los coeditores o de sus representantes.

ISBN versión impresa: 978-607-32-1511-4

ISBN E-Book: 978-607-32-1501-5

ISBN E-Chapter: 978-607-32-1502-2

Impreso en México. Printed in Mexico.

1 2 3 4 5 6 7 8 9 0 – 15 14 13 12

PEARSON

www.pearsonenespañol.com

Contenido

Introducción	ix
CD-ROOM con ProModel	x
Agradecimientos	x
Capítulo 1 Principios básicos de la simulación	1
1.1 Introducción a la simulación	2
1.2 Definiciones de simulación	4
1.3 Ventajas y desventajas de la simulación	9
1.4 Elementos clave para garantizar el éxito de un modelo de simulación	10
1.5 Pasos para realizar un estudio de simulación	12
1.6 Problemas	16
Referencias	20
Capítulo 2 Números pseudoaleatorios	21
2.1 Los números pseudoaleatorios	22
2.2 Generación de números pseudoaleatorios	22
2.2.1 Algoritmo de cuadrados medios	24
2.2.2 Algoritmo de productos medios	25
2.2.3 Algoritmo de multiplicador constante	26
2.2.4 Algoritmo lineal	27
2.2.5 Algoritmo congruencial multiplicativo	29
2.2.6 Algoritmo congruencial aditivo	30
2.2.7 Algoritmos congruenciales no lineales	31
2.3 Propiedades de los números pseudoaleatorios entre 0 y 1	32
2.4 Pruebas estadísticas para los números pseudoaleatorios	34
2.4.1 Prueba de medias	35
2.4.2 Prueba de varianza	36
2.4.3 Pruebas de uniformidad	38
2.4.4 Pruebas de independencia	41
2.5 Problemas	52
Referencias	58

Capítulo 3	Variables aleatorias	59
3.1	Definición de variable aleatoria	60
3.2	Tipos de variables aleatorias	60
3.3	Determinación del tipo de distribución de un conjunto de datos	62
3.3.1	Prueba Chi-cuadrada	63
3.3.2	Prueba de Kolmogorov-Smirnov	65
3.3.3	Prueba de Anderson-Darling	68
3.3.4	Ajuste de datos con Stat::Fit	74
3.4	Generación de variables aleatorias	78
3.4.1	Método de la transformada inversa	78
3.4.2	Método de convolución	86
3.4.3	Método de composición	89
3.4.4	Método de transformación directa	92
3.5	Expresiones comunes de algunos generadores de variables aleatorias	93
3.6	Problemas	96
	Referencias	112
Capítulo 4	Simulación de variables aleatorias	113
4.1	Verificación y validación de los modelos de simulación	114
4.1.1	Simulaciones terminales	114
4.2	Simulaciones no terminales o de estado estable	117
4.2.1	Longitud de las réplicas	117
4.3	Modelos de simulación	121
4.3.1	Modelo de una línea de espera con un servidor	122
4.3.2	Modelo de un proceso de ensamble e inspección	126
4.3.3	Modelo de un sistema de inventarios	130
4.4	Selección de lenguajes de simulación	134
4.5	Caso de estudio 1	135
4.6	Caso de estudio 2	137
4.7	Problemas	138
Capítulo 5	Simulación con ProModel	151
5.1	Introducción al uso de ProModel	152
5.2	Elementos básicos	152
5.3	Estructura de programación en ProModel	153

5.4	Construcción de un modelo	153
5.4.1	Modelo M/M/1 de líneas de espera	153
5.4.2	Mejoramiento visual del modelo	166
5.4.3	Modelado de un sistema que incluye más de un proceso	175
5.4.4	Inclusión de gráficos de fondo en el modelo	183
5.5	Arribos cíclicos	186
5.6	Caso integrador	190
5.7	Problemas	192
Capítulo 6	Instrucciones de ProModel	201
6.1	Uso de la biblioteca de funciones probabilísticas	202
6.2	Recursos	207
6.3	Paros en los equipos	212
6.4	Reglas de ruteo	220
6.5	Ensamblajes, acumulación y agrupamiento de piezas	225
6.6	Transporte entre estaciones	241
6.7	Instrucciones de control	252
6.8	Optimización con Simrunner	257
	Introducción	257
	SimRunner	258
6.9	Caso integrador 1	269
6.10	Caso integrador 2	271
6.11	Problemas	277
Anexo 1	Distribuciones de probabilidad	295
A.1	Distribuciones continuas	296
A.2	Distribuciones discretas	305
Anexo 2	Reportes estadísticos en ProModel (formato Viewer 2.0)	311
Anexo 3	Reportes estadísticos en ProModel	321
Anexo 4	Distribuciones de probabilidad	331

Introducción

La simulación es una forma de estudiar los procesos aleatorios, los cuales se encuentran prácticamente en todas las operaciones de sistemas de producción y servicios. Aprender a modelar con simulación estocástica discreta es un reto demandante, principalmente por la complejidad del tema y porque el proceso de simulación y el análisis de los resultados requieren de un razonable conocimiento de probabilidad, estadística y computación. La experiencia nos ha mostrado que estos temas son difíciles de dominar, y desafortunadamente la mayoría de los estudiantes no se sienten cómodos al trabajar con ellos.

En este texto presentamos los conceptos de la modelación de los procesos estocásticos, el análisis estadístico de la información y la relación de éstos con la simulación estocástica discreta, utilizando como herramienta el programa ProModel. ProModel y su entorno con el usuario fueron desarrollados específicamente para simplificar la interacción entre el estudiante y la computadora. ProModel es una poderosa herramienta de análisis que, sin embargo, es fácil de usar.

Para un mejor manejo de los temas, el libro se ha dividido en seis capítulos. El capítulo 1 establece los conceptos básicos relacionados con un proyecto de simulación, e incluye una introducción a la técnica y la metodología para su desarrollo. El capítulo 2 presenta los números aleatorios, base de los modelos estocásticos, sus propiedades, manejo y generación, así como todos los requerimientos para ser considerados como tales. También se han agregado más problemas y fórmulas de generación de números pseudoaleatorios, como son los métodos MINSTD y Super-Duper.

El capítulo 3 ofrece los conceptos de pruebas de bondad de ajuste, para determinar la distribución de probabilidad asociada a las variables de decisión y eventos en el sistema a modelar, para con ello generar variables aleatorias que puedan ser usadas durante la simulación. Este capítulo incorpora el uso de la herramienta Stat:Fit, que se incluye junto en el CD que acompaña este libro, la cual permite determinar automáticamente la distribución de probabilidad de las variables y eventos a modelar en el sistema; con este fin se han modificado muchos de los problemas, para poder ser analizados con Stat:Fit, también se ha agregado una gran cantidad de problemas que involucran la generación de variables aleatorias con el método de la transformada inversa.

El capítulo 4 maneja los conceptos de validación y análisis de los modelos de simulación; y presenta al final del capítulo ejemplos de simulación MonteCarlo, desarrollados en hojas de cálculo, sobre líneas de espera, procesos de ensamble y sistemas de inventarios, con la esperanza de que el lector sea capaz de realizar modelos simples usando una hoja de cálculo. Se han agregado dos casos de estudio: en el primero se requiere de la optimización de un sistema de producción, y el segundo es sobre el diseño de un sistema de muestreo. Adicionalmente se ha agregado más del doble de problemas que en la versión anterior.

El capítulo 5 presenta las características y bondades de ProModel a través de ejemplos que guían al usuario en la construcción de los modelos. El capítulo 6, por su parte, cubre elementos más complejos de programación que le permitirán ampliar las capacidades de modelación. En estos capítulos se han agregado nuevos conceptos de programación, específicamente aquellos que tienen que ver con arribos cíclicos, la creación de funciones de usuario, el uso de la programación de paros por turnos, el manejo de materiales usando grúas viajeras, ejemplos de instrucciones de control del tipo IF-THEN-ELSE, WHILE-DO, entre otros, y el uso de la herramienta SimRunner que permite la optimización de los modelos desarrollados con ProModel.

En estos capítulos se han agregado casos integradores y mayor cantidad de ejercicios para fortalecer el aprendizaje.

Finalmente, el nuevo apéndice describe el significado de los resultados obtenidos en los reportes con la nueva versión Output Viewer 4.0 de ProModel.

CD-ROM con ProModel

Es importante destacar que el CD-ROM que se incluye con el libro contiene la versión estudiantil más reciente de ProModel con todas las funciones de la versión profesional. La única restricción es en cuanto al tamaño de los modelos que pueden construirse.

Agradecimientos

Agradecemos a las personas e instituciones que han hecho posible la publicación de esta segunda edición. A nuestros estudiantes, quienes con sus comentarios nos han permitido agregar nuevo material y corregir detalles que se nos escaparon en la primera edición. Al departamento de Ingeniería Industrial y de Sistemas del ITESM Campus Monterrey por el tiempo que nos dieron para que este libro llegara a ser realidad. Queremos agradecer a ProModel por permitirnos incluir su software, y en especial a Daniel Villarreal Parás por su apoyo constante e incondicional para lograrlo. Finalmente, agradecemos al equipo editorial de Pearson por creer en nosotros, especialmente a Luis Cruz Castillo y Óscar Ortega, con quienes hemos trabajado durante estos últimos años.

Eduardo García Dunna



Capítulo 1

Principios básicos de la simulación

- 1.1** Introducción a la simulación
 - 1.2** Definiciones de simulación
 - 1.3** Ventajas y desventajas de la simulación
 - 1.4** Elementos clave para garantizar el éxito de un modelo de simulación
 - 1.5** Pasos para realizar un estudio de simulación
 - 1.6** Problemas
- Referencias

1.1 Introducción a la simulación

La creación de nuevos y mejores desarrollos en el área de la computación ha traído consigo innovaciones muy importantes tanto en la toma de decisiones como en el diseño de procesos y productos. Una de las técnicas para realizar estudios piloto, con resultados rápidos y a un relativo bajo costo, se basa en la modelación —la cual se conoce como simulación— que se ha visto beneficiada por estos avances.

Actualmente, el analista que desea hacer uso de esta herramienta dispone de una gran cantidad de software de simulación que le permite tomar decisiones en temas muy diversos. Por ejemplo, en aplicaciones donde se usan lenguajes de programación general como Fortran, para analizar esquemas de administración del inventario,^[1] en aplicaciones para analizar y mejorar la administración de la cadena de suministro,^[2] en análisis y validación de medidas de desempeño en sistemas de manufactura,^[3] en aplicaciones de mejoras mediante despliegues seis sigma,^[4] o en procesos de capacitación a través de la simulación como herramienta base.^[5] Sin duda, la flexibilidad en cuanto a la modelación, el análisis y el mejoramiento de sistemas ha hecho de la simulación una herramienta cuyo uso y desarrollo se han visto alentados de manera significativa. En la actualidad resulta fácil encontrar paquetes de software con gran capacidad de análisis, así como mejores animaciones (2D y 3D) y aplicaciones para generación de reportes con información relevante para la toma de decisiones. En general, dichos paquetes —ya sea orientados a procesos, a servicios, o de índole general— nos proveen de una enorme diversidad de herramientas estadísticas que permiten tanto un manejo más eficiente de la información relevante bajo análisis como una mejor presentación e interpretación.

El concepto de **simulación** engloba soluciones para muchos propósitos diferentes. Por ejemplo, podríamos decir que el modelo de un avión a escala que se introduce a una cámara por donde se hace pasar un flujo de aire, puede simular los efectos que experimentará un avión real cuando se vea sometido a turbulencia. Por otro lado, algunos paquetes permiten hacer la representación de un proceso de fresado o torneado: una vez que el usuario establezca ciertas condiciones iniciales, podrá ver cómo se llevaría a cabo el proceso real, lo que le da la posibilidad de revisarlo sin necesidad de desperdiciar material ni poner en riesgo la maquinaria.

Por lo tanto, la simulación se refiere a un gran conjunto de métodos y aplicaciones que buscan imitar el comportamiento de sistemas reales, generalmente por medio de una computadora con un software apropiado.

Existen distintos **modelos** de simulación que permiten representar situaciones reales de diferentes tipos. Podemos tener modelos físicos —como el del avión que mencionamos en el párrafo anterior— o modelos matemáticos, a los cuales pertenecen los modelos de simulación de eventos discretos. Asimismo, los modelos pueden diferenciarse según el tipo de ecuaciones matemáticas que los componen. Por ejemplo, los **modelos continuos** son aquellos en los que las relaciones entre las variables relevantes de la situación real se definen por medio de ecuaciones diferenciales, ya que éstas permiten conocer el comportamiento de las variables en cierto tiempo. Problemas tales como saber de qué manera se transfiere el calor en un molde, o determinar cómo fluye cierto material dentro de una tubería, e incluso prever el comportamiento del nivel de un tan-

que de gasolina al paso del tiempo, mientras el vehículo está en marcha, pueden simularse con este tipo de modelo.

Además de modelos continuos tenemos **modelos discretos**. En ellos el comportamiento que nos interesa analizar puede representarse por medio de ecuaciones evaluadas en un punto determinado. Por ejemplo, si hacemos un muestreo del número de personas que llegaron a un banco en un lapso específico, podemos simular esta variable con ecuaciones ligadas a distribuciones de probabilidad que reflejen dicho comportamiento.

Otro tipo de clasificación es el de los modelos dinámicos o estáticos. Los **modelos dinámicos** son aquellos en los que el estado del sistema que estamos analizando cambia respecto del tiempo. Por ejemplo, el número de personas que hacen fila para entrar a una sala de cine varía con el tiempo. Por otro lado, los **modelos estáticos** representan un resultado bajo un conjunto de situaciones o condiciones determinado; por ejemplo, al lanzar un dado los únicos valores que se puede obtener son 1, 2, 3, 4, 5 o 6, de manera que el resultado de la simulación será uno de tales valores posibles; a esto se le conoce generalmente como simulación de Monte Carlo.

Por último, podemos hablar de **modelos determinísticos** y **modelos probabilísticos**, llamados también **estocásticos**. Los primeros se refieren a relaciones constantes entre los cambios de las variables del modelo. Por ejemplo, si las cajas empleadas en un proceso contienen siempre 5 productos, cada vez que se añada una caja al inventario éste se incrementará en 5 unidades. Si, por el contrario, hay una distribución de probabilidad en el proceso de manera que, por ejemplo, algunas cajas contienen 3 productos y otras 4, el inventario se modificará según el número de piezas de cada caja y, en consecuencia, será necesario un modelo estocástico. En el caso de la simulación de eventos discretos hablaremos de modelos matemáticos, discretos, dinámicos, y que pueden incluir variables determinísticas y probabilísticas.

En este libro nos ocuparemos del proceso que se basa en el uso de ecuaciones matemáticas y estadísticas, conocido como **simulación de eventos discretos**. Este proceso consiste en relacionar los diferentes eventos que pueden cambiar el estado de un sistema bajo estudio por medio de distribuciones de probabilidad y condiciones lógicas del problema que se esté analizando. Por ejemplo, un proceso de inspección donde sabemos estadísticamente que 0.2% de los productos tiene algún tipo de defecto, puede simularse con facilidad mediante una simple hoja de cálculo, al considerar estadísticas de rechazos y productos sin defecto, y al asignar una distribución de probabilidad con 0.2% de oportunidad de defecto para cada intento de inspección.

En este capítulo abordaremos las definiciones básicas de los conceptos de la simulación de eventos discretos. En el capítulo 2 se presentarán algunos otros elementos relevantes, como los números pseudoaleatorios y las pruebas estadísticas necesarias para comprobar esta aleatoriedad, la generación de variables aleatorias, y la caracterización de algunas distribuciones de probabilidad de uso común en la simulación. El capítulo 3 muestra cómo usar estos números pseudoaleatorios para modelar sistemas, y una vez que sepamos utilizarlos podremos realizar una simulación sencilla con la ayuda de una hoja de cálculo. En el capítulo 4 se presenta simulación de variables aleatorias. En el resto de los capítulos describiremos el uso del software comercial ProModel, del cual incluimos una versión limitada.

1.2 Definiciones de simulación

Para poder realizar un buen estudio de simulación es necesario entender los conceptos básicos que componen nuestro modelo.

Comenzaremos por definir el concepto de **simulación de eventos discretos** como el conjunto de relaciones lógicas, matemáticas y probabilísticas que integran el comportamiento de un sistema bajo estudio cuando se presenta un evento determinado. El objetivo del modelo de simulación consiste, precisamente, en comprender, analizar y mejorar las condiciones de operación relevantes del sistema.

En la definición anterior encontramos elementos como sistema, modelo y evento, de los que se desprenden otros conceptos importantes dentro de una simulación. A continuación abundaremos en cada uno.

La definición básica de **sistema** nos dice que se trata de un conjunto de elementos que se interrelacionan para funcionar como un todo; desde el punto de vista de la simulación, tales elementos deben tener una frontera clara. Por ejemplo, podemos hablar del sistema de atención a clientes en un banco, del sistema de inventarios de una empresa, o del sistema de atención en la sala de emergencia de un hospital. Cada uno puede dividirse en elementos que son relevantes para la construcción de lo que será su modelo de simulación; entre ellos tenemos **entidades, estado del sistema, eventos actuales y futuros, localizaciones, recursos, atributos, variables, y el reloj de la simulación**, los cuales a continuación se describen:

Una **entidad** por lo general es la representación de los flujos de entrada y salida en un sistema; al entrar a un sistema una entidad es el elemento responsable de que el estado del sistema cambie. Ejemplos de entidades pueden ser; los clientes que llegan a la caja de un banco, las piezas que llegan a un proceso, o el embarque de piezas que llega a un inventario.

El **estado del sistema** es la condición que guarda el sistema bajo estudio en un momento de tiempo determinado; es como una fotografía de lo que está pasando en el sistema en cierto instante. El estado del sistema se compone de variables o características de operación puntuales (digamos el número de piezas que hay en el sistema en ese momento), y de variables o características de operación acumuladas, o promedio (como podría ser el tiempo promedio de permanencia de una entidad en el sistema, en una fila, almacén o equipo).

Un **evento** es un cambio en el estado actual del sistema; por ejemplo, la entrada o salida de una entidad, la finalización de un proceso en un equipo, la interrupción o reactivación de una operación (digamos por un descanso del operario), o la descompostura de una máquina. Podemos catalogar estos eventos en dos tipos: **eventos actuales**, aquellos que están sucediendo en el sistema en un momento dado, y **eventos futuros**, cambios que se presentarán en el sistema después del tiempo de simulación, de acuerdo con una programación específica. Por ejemplo, imagine que cierta pieza entra a una máquina para que ésta realice un proceso. El evento actual sería precisamente que la entidad llamada "pieza" se encuentra en la máquina. El evento futuro podría ser el momento en que la máquina concluirá su trabajo con la pieza y ésta seguirá su camino hacia el siguiente proceso lógico, de acuerdo con la programación: almacenamiento, inspección o entrada a otra máquina. Para ilustrar con mayor claridad estas definiciones veamos la siguiente figura.

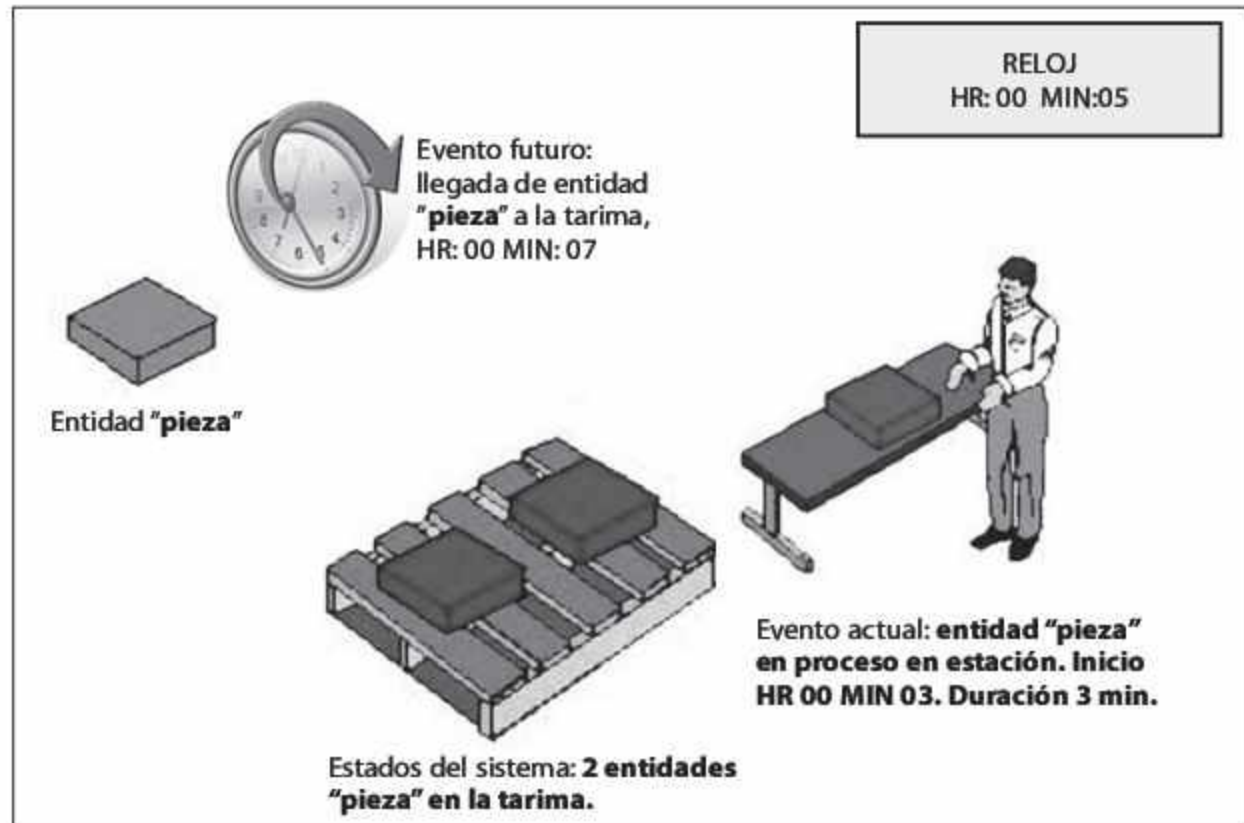


Figura 1.1
Representación
de conceptos de
simulación.

Adicional a la figura 1.1 podemos ilustrar los cambios en el estado del sistema de manera tabular y la manera en que va cambiando con cada evento en el tiempo. Si bien es cierto que en un modelo real se puede considerar una gran variedad de estadísticas, como podría ser el tiempo de permanencia en el sistema, el tiempo que pasa entre estaciones, el tiempo de procesos y el promedio de piezas. Cada una de estas estadísticas se vuelve a calcular hasta que un nuevo evento altera el estado del sistema actual; de aquí la importancia de identificar este tipo de simulación como eventos discretos. Esta representación se muestra en la siguiente tabla, la cual sólo considera el número de entidades en el sistema, en un modelo real ambas tablas contendrían más información.

Tabla de eventos futuros			Historia de la simulación			
Reloj	Evento futuro	Estatus	Reloj	Evento	Estado tarima	Estado estación
00:02	Llegada pieza	Realizado	00:00	Inicio de simulación	0 piezas	0 piezas
00:03	Mover pieza a estación	Realizado	00:02	Llegada de pieza	1 pieza	0 piezas
00:04	Llegada de la pieza	Realizado	00:03	Pieza en estación	0 piezas	1 pieza
00:05	Llegada de la pieza	Realizado	00:04	Llegada de pieza	1 pieza	1 pieza
00:06	Fin de proceso estación	En espera	00:05	Llegada de pieza	2 piezas	1 pieza
00:07	Llegada de la pieza	En espera				

RELOJ
HR: 00 MIN 05

Tabla 1.1 Relación eventos y estado del sistema.

Además del esquema transaccional (pieza en tarima -> pieza en estación) que se presenta en un modelo de simulación, es necesario considerar algunos otros elementos que también forman parte de este tipo de modelaciones. A continuación describiremos algunos de ellos.

Las **localizaciones** son todos aquellos *lugares en los que la pieza puede detenerse para ser transformada o esperar a serlo*. Dentro de estas localizaciones tenemos almacenes, bandas transportadoras, máquinas, estaciones de inspección, etcétera. En el caso del gráfico mostrado en la figura 1.1 la tarima y la estación serían consideradas localizaciones del modelo. Observe que en el caso de la estación una sola localización puede ser representada gráficamente por varias figuras. En la estación observamos una mesa y a una persona que en conjunto forman una sola localización. En términos de simulación algunos paquetes permiten la animación de lo que se programó. En estos paquetes la representación iconográfica es sólo para aspectos visuales y no le resta o agrega potencia al modelo. Es decir, podríamos haber colocado una casa en lugar de la estación con mesa y operador y el modelo nos hubiese dado el mismo resultado.

Los **recursos** son aquellos *dispositivos —diferentes a las localizaciones— necesarios para llevar a cabo una operación*. Por ejemplo, un montacargas que transporta una pieza de un lugar a otro: una persona que realiza la inspección en una estación y toma turnos para descansar; una herramienta necesaria para realizar un proceso pero que no forma parte de una localización específica, sino que es trasladada de acuerdo con los requerimientos de aquel.

Un **atributo** es una *característica de una entidad*. Por ejemplo, si la entidad es un motor, los atributos serían su color, peso, tamaño o cilindraje. Los atributos son muy útiles para diferenciar entidades sin necesidad de generar una nueva, y pueden adjudicarse al momento de la creación de la entidad, o asignarse y/o cambiarse durante el proceso.

Como indica su nombre, las **variables** son *condiciones cuyos valores se crean y modifican por medio de ecuaciones matemáticas y relaciones lógicas*. Pueden ser *continuas* (por ejemplo, el costo promedio de operación de un sistema) o *discretas* (como el número de unidades que deberá envasarse en un contenedor). Las variables son muy útiles para realizar conteos de piezas y ciclos de operación, así como para determinar características de operación del sistema.

El **reloj de la simulación** es el *contador de tiempo de la simulación*, y su función consiste en responder preguntas tales como cuánto tiempo se ha utilizado el modelo en la simulación, y cuánto tiempo en total se quiere que dure esta última. En general, el reloj de simulación se relaciona con la tabla de eventos futuros, pues al cumplirse el tiempo programado para la realización de un evento futuro, éste se convierte en un evento actual. Regresando al ejemplo de la figura 1.1, cuando el tiempo de proceso se cumpla, la pieza seguirá su camino hasta su siguiente localización, si ésta es la última del sistema lo más probable es que su siguiente proceso sea salir del sistema; el reloj simula precisamente ese tiempo.

Podemos hablar de dos tipos de reloj de simulación: el **reloj de simulación absoluto**, que parte de cero y termina en un tiempo total de simulación definido, y el **reloj de simulación relativo**, que sólo considera el lapso que transcurre entre dos eventos. Por ejemplo, podemos decir que el tiempo de proceso de una pieza es relativo, mientras que el absoluto sería el tiempo global de la simulación: desde que la pieza entró a ser procesada hasta el momento en el que terminó su proceso.

Ejemplo 1.1

Un taller recibe ciertas piezas, mismas que son acumuladas en un almacén temporal en donde esperan a ser procesadas. Esto ocurre cuando un operario transporta las piezas del almacén a un torno. Desarrolle un modelo que incluya el número de piezas que hay en el almacén y que esperan ser atendidas en todo momento, y el número de piezas procesadas en el torno.

En la siguiente figura podemos observar cómo se vería un modelo de simulación para este ejemplo.

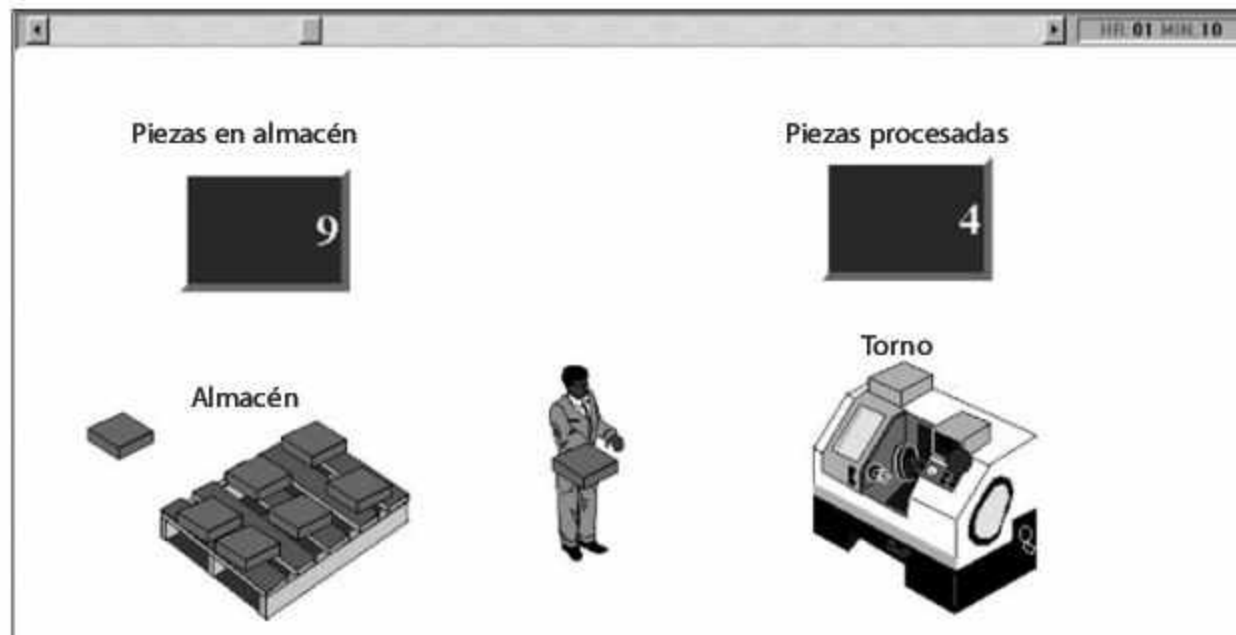


Figura 1.2
Modelo de simulación para el ejemplo 1.1.

En este ejemplo podemos identificar algunos de los elementos que participan en un modelo de simulación, de acuerdo con las definiciones que hemos comentado:

Sistema: En este caso, el sistema está conformado por el conjunto de elementos interrelacionados para el funcionamiento del proceso: las piezas, el almacén temporal, el operario, el torno.

Entidades: En este modelo sólo tenemos una entidad; las piezas, que representan los flujos de entrada al sistema del problema bajo análisis.

Estado del sistema: Podemos observar que cuando llevamos 1 hora 10 minutos de simulación (vea el extremo superior derecho de la figura) en el almacén se encuentran 9 piezas esperando a ser procesadas; el operario está transportando una pieza más para procesarla en el torno. El torno, por lo tanto, no está trabajando en ese momento, aunque ya ha procesado 4 piezas. Adicional a estos datos, podemos llevar un control de otras estadísticas relacionadas con el estado del sistema, como el tiempo promedio de permanencia de las piezas en los estantes del almacén temporal o en el sistema global.

Eventos: Entre otros, podríamos considerar como eventos de este sistema el tiempo de descanso del operario o la salida de una pieza tras ser procesada por el torno. Además es posible identificar un evento futuro: la llegada de la siguiente pieza al sistema (tendríamos más eventos de este tipo respecto de las piezas que esperan a que el operario las tome).

Localizaciones: En este caso tenemos el almacén al que deberán llegar las piezas y en el que esperarán a ser procesadas, así como el torno en donde esto ocurrirá.

Recursos: En este modelo, un recurso es el operario que transporta las piezas del almacén al torno.

Atributos: Digamos que (aunque no se menciona en el ejemplo) las piezas pueden ser de tres tamaños diferentes. En este caso, un atributo llamado tamaño podría agregarse a la información de cada pieza que llega al sistema, para más adelante seleccionar el tipo de operación que deberá realizarse y el tiempo necesario para llevarla a cabo de acuerdo con dicho atributo.

Variables: Tenemos dos variables definidas en este caso: el número de piezas en el almacén y el número de piezas procesadas en el torno.

Reloj de la simulación: Como se puede ver en la esquina superior derecha de la figura 1.2, en este momento la simulación lleva 1 hora 10 minutos. El reloj de la simulación continuará avanzando hasta el momento que se haya establecido para el término de la simulación, o hasta que se cumpla una condición lógica para detenerla, por ejemplo, el número de piezas que se desean simular.

Otro concepto importante que vale la pena definir es el de **réplica** o **corrida** de la simulación. Cuando ejecutamos el modelo una vez, los valores que obtenemos de las variables y parámetros al final del tiempo de simulación generalmente serán distintos de los que se producirán si lo volvemos a correr con diferentes números pseudoaleatorios. Por lo tanto, es necesario efectuar más de una réplica del modelo que se esté analizando, con la finalidad de obtener estadísticas de intervalo que nos den una mejor ubicación del verdadero valor de la variable bajo los diferentes escenarios que se presentan al modificar los números pseudoaleatorios en cada oportunidad.

De esta manera, la pregunta clave es: ¿cuánto tiempo se debe simular un modelo para obtener resultados confiables? En general, podemos decir que todas las variables que se obtienen en términos de promedios presentan dos diferentes etapas: un **estado transitorio** y un **estado estable**. El primero se presenta al principio de la simulación; por ejemplo, en el arranque de una planta, cuando no tiene material en proceso: el último de los procesos estará inactivo hasta que el primer cliente llegue, y si el tiempo de simulación es bajo, su impacto sobre la utilización promedio de este proceso será muy alto, lo cual no ocurriría si el modelo se simulara lo suficiente para lograr una compensación. En el estado transitorio hay mucha variación entre los valores promedio de las variables de decisión del modelo, por lo que formular conclusiones con base en ellos sería muy arriesgado, toda vez que difícilmente nos darían una representación fiel de la realidad.

Por otro lado, en el **estado estable** los valores de las variables de decisión permanecen muy estables, y presentan sólo variaciones poco significativas. En este momento las decisiones que se tomen serán mucho más confiables. Sin embargo, no todas las variables convergen al estado estable con la misma rapidez: algunas pasan con más lentitud que otras de un estado transitorio a uno estable. Es responsabilidad del analista verificar que las variables de decisión del modelo se encuentren en estado estable antes de detener el tiempo de la simulación (vea la figura 1.3).

Otro factor importante para decidir el tiempo de simulación es el costo de la corrida. Mayor tiempo de simulación requiere más tiempo computacional, lo cual implica, necesariamente, un costo más alto. Por supuesto, la situación empeora si a esto le agregamos que en algunos casos es preciso efectuar más de tres réplicas.

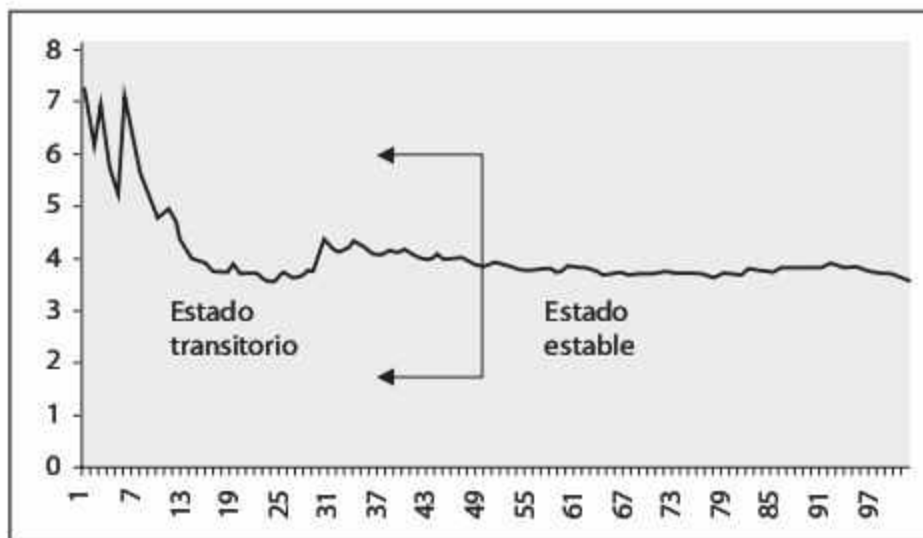


Figura 1.3
Gráfica de estabilización de una variable.

1.3 Ventajas y desventajas de la simulación

Como hemos visto hasta ahora, la simulación es una de las diversas herramientas con las que cuenta el analista para tomar decisiones y mejorar sus procesos. Sin embargo, se debe destacar que, como todas las demás opciones de que disponemos, la simulación de eventos discretos presenta ventajas y desventajas que se precisa tomar en cuenta al decidir si es apta para resolver un problema determinado.

Dentro de las ventajas más comunes que ofrece la simulación podemos citar las siguientes:

- a) Es muy buena herramienta para conocer el impacto de los cambios en los procesos, sin necesidad de llevarlos a cabo en la realidad.
- b) Mejora el conocimiento del proceso actual ya que permite que el analista vea cómo se comporta el modelo generado bajo diferentes escenarios.
- c) Puede utilizarse como medio de capacitación para la toma de decisiones.
- d) Es más económico realizar un estudio de simulación que hacer muchos cambios en los procesos reales.

- e) Permite probar varios escenarios en busca de las mejores condiciones de trabajo de los procesos que se simulan.
- f) En problemas de gran complejidad, la simulación permite generar una buena solución.
- g) En la actualidad los paquetes de software para simulación tienden a ser más sencillos, lo que facilita su aplicación.
- h) Gracias a las herramientas de animación que forman parte de muchos de esos paquetes es posible ver cómo se comportará un proceso una vez que sea mejorado.

Éstas son algunas de las desventajas que la simulación puede presentar:

- a) Aunque muchos paquetes de software permiten obtener el mejor escenario a partir de una combinación de variaciones posibles, la simulación *no* es una herramienta de optimización.
- b) La simulación puede ser costosa cuando se quiere emplearla en problemas relativamente sencillos de resolver, en lugar de utilizar soluciones analíticas que se han desarrollado de manera específica para ese tipo de casos.
- c) Se requiere bastante tiempo —por lo general meses— para realizar un buen estudio de simulación; por desgracia, no todos los analistas tienen la disposición (o la oportunidad) de esperar ese tiempo para obtener una respuesta.
- d) Es preciso que el analista domine el uso del paquete de simulación y que tenga sólidos conocimientos de estadística para interpretar los resultados.
- e) En algunas ocasiones el cliente puede tener falsas expectativas de la herramienta de simulación, a tal grado que le asocia condiciones similares a un video juego o a una bola de cristal que le permite predecir con exactitud el futuro.

1.4 Elementos clave para garantizar el éxito de un modelo de simulación

Pese a los beneficios que conlleva la simulación, es imposible garantizar que un modelo tendrá éxito. Existen ciertas condiciones clave que pueden traer problemas si no se les pone atención al momento de usar la simulación para la toma de decisiones. A continuación destacaremos algunas de las causas por las que un modelo de simulación podría no tener los resultados que se desean.

Tamaño insuficiente de la corrida. Como se mencionó antes, para poder llegar a conclusiones estadísticas válidas a partir de los modelos de simulación es necesario que las variables aleatorias de respuesta se encuentren en estado estable. El problema estriba en que, por lo general, cuando el modelo consta de más de una variable de decisión, es difícil que éstas alcancen un estado estable al mismo tiempo: es posible que una se encuentre estable y la otra no en un momento determinado, por lo que las conclusiones respecto de la segunda variable no serán confiables en cuanto a la estadística.

Variable(s) de respuesta mal definida(s). Aun cuando el modelo de simulación sea muy eficiente y represente en gran medida la realidad, si la variable de respuesta seleccionada

no es la apropiada, será imposible tomar decisiones que tengan impacto en la operación del sistema bajo estudio.

Por ejemplo, digamos que una variable de respuesta es el nivel de inventarios de cierto producto, sin embargo, la política de la empresa establece que no se debe parar ninguno de los procesos de fabricación. En consecuencia, el problema no será el inventario final, sino el ritmo de producción necesario para que aquel cumpla con los requerimientos de diseño que se desean.

Errores al establecer las relaciones entre las variables aleatorias. Un error común de programación es olvidar las relaciones lógicas que existen entre las variables aleatorias del modelo, o minimizar su impacto. Si una de estas variables no está definida de manera correcta, es posible tener un modelo que se apegue a la realidad actual; sin embargo, si el sistema no se lleva hasta su máxima capacidad para observar su comportamiento, podría resultar imposible visualizar el verdadero impacto de las deficiencias.

Errores al determinar el tipo de distribución asociado a las variables aleatorias del modelo. Este tipo de problema es muy similar al anterior, pero en este caso se utilizan distribuciones que no son las más adecuadas o que responden únicamente a un intento de simplificar los estudios estadísticos. Digamos, por ejemplo, que se nos dan los siguientes parámetros de producción aproximados: mínimo 10, máximo 40, y promedio 30. En esta circunstancia la tentación de simplificar el estudio de la variable asignándole una distribución triangular con parámetros (10, 30, 40) es muy grande; no obstante, hacerlo afectaría de manera importante los resultados de la simulación, pues el modelo podría alejarse de lo que sucede en la realidad.

Falta de un análisis estadístico de los resultados. Un problema común por el que la simulación suele ser objeto de crítica, radica en asumir que se trata de una herramienta de optimización. Esta apreciación es incorrecta, ya que involucra variables aleatorias y características propias de un modelo que incluye probabilidades. Por lo mismo —como se apuntó antes—, es necesario realizar varias corridas a fin de producir diferentes resultados finales para las variables de respuesta y, a partir de esos valores, obtener intervalos de confianza que puedan dar un rango en dónde encontrar los valores definitivos. Este tipo de problemas se presentan también al comparar dos escenarios: podríamos encontrar un mejor resultado para uno de ellos, pero si los intervalos de confianza de las variables de respuesta se traslapan resultaría imposible decir que el resultado de un escenario es mejor que el del otro. De hecho, en lo que a la estadística se refiere, ambos resultados pueden ser iguales. En ese caso incrementar el tamaño de corrida o el número de réplicas puede ayudar a obtener mejores conclusiones.

Uso incorrecto de la información obtenida. Un problema que se presenta en ocasiones es el uso incorrecto de la información recabada para la realización del estudio, ya sea a través de un cliente o de cualquier otra fuente. Muchas veces esta información se recolecta, analiza y administra de acuerdo con las necesidades propias de la empresa, lo que implica que no siempre está en el formato y la presentación que se requiere para la simulación. Si la información se utiliza para determinar los parámetros del modelo sin ser

depurada y reorganizada, es muy probable que la precisión de los resultados del estudio se vea afectada.

Falta o exceso de detalle en el modelo. Otro punto importante a considerar es el nivel de detalle del modelo. En muchas ocasiones algún proceso se simplifica tanto que tiende a verse como una “caja negra” que nos impide ver qué ocurre en el interior, aunque sí haya entrada y salida de datos que interactúan con otras partes del modelo. Cuando esto sucede, el impacto que podrían tener los subprocessos que se llevan a cabo en la “caja negra” (es decir, del proceso sobresimplificado) no se incluye en la simulación. Por ejemplo, si se analiza un sistema de distribución y se da por sentado que el almacén *siempre* surte sus pedidos, no incluiremos el impacto de los tiempos necesarios para surtir las órdenes, ni la posibilidad de que haya faltantes de producto; excluirémos también los horarios de comida, en los que no se surten pedidos, y las fallas en los montacargas que transportan los pedidos hasta los camiones para su distribución. Por otra parte, si el modelo se hace demasiado detallado, tanto el tiempo dedicado al estudio como el costo de llevarlo a cabo podrían incrementarse sustancialmente. Es labor del encargado de la simulación sugerir y clarificar los niveles de detalle que se requieren en el modelo, resaltando los alcances y limitaciones de cada uno.

1.5 Pasos para realizar un estudio de simulación

Debemos considerar que —igual a lo que ocurre con otras herramientas de investigación— la realización de un estudio de simulación requiere la ejecución de una serie de actividades y análisis que permitan sacarle el mejor provecho. A continuación se mencionan los pasos básicos para realizar un estudio de simulación, aunque en muchas ocasiones será necesario agregar otros o suprimir algunos de los aquí enumerados, de acuerdo con la problemática en cuestión.

1. **Definición del sistema bajo estudio.** En esta etapa es necesario conocer el sistema a modelar. Para ello se requiere saber qué origina el estudio de simulación y establecer los supuestos del modelo: es conveniente definir con claridad las variables de decisión del modelo, determinar las interacciones entre éstas, y establecer con precisión los alcances y limitaciones que aquel podría llegar a tener.

Antes de concluir este paso es recomendable contar con la información suficiente para lograr establecer un modelo conceptual o un mapa mental del sistema bajo estudio, el cual debe incluir sus fronteras y todos los elementos que lo componen, además de las interacciones entre ellos, los flujos de productos, las personas y los recursos, así como las variables de mayor interés para el problema.

2. **Generación del modelo de simulación base.** Una vez que se ha definido el sistema en términos de un modelo conceptual, la siguiente etapa del estudio consiste en la generación de un modelo de simulación base. No es preciso que este modelo sea demasiado detallado, pues se requiere mucha más información estadística sobre el comportamiento de las variables de decisión del sistema. La generación de este modelo es el primer reto para el programador de la simulación, ya que debe traducir a un lenguaje de simulación la información que se

obtuvo en la etapa de definición del sistema, e incluir las interrelaciones de todos los posibles subsistemas que existan en el problema a modelar. En caso de que se requiera una animación, éste también es un buen momento para definir qué gráfico puede representar mejor el sistema que se modela.

Igual que ocurre en otras ramas de la investigación de operaciones, “la simulación exige ciencia y arte en la generación de sus modelos”. El realizador de un estudio de simulación es, en este sentido, como un artista que debe usar toda su creatividad para realizar un buen modelo que refleje la realidad del problema que se está analizando. Conforme se avanza en el modelo base se pueden ir agregando las variables aleatorias del sistema, con sus respectivas distribuciones de probabilidad asociadas.

3. **Recolección y análisis de datos.** Es posible comenzar la recopilación de la información estadística de las variables aleatorias del modelo de manera paralela a la generación del modelo base. En esta etapa se debe establecer qué información es útil para la determinación de las distribuciones de probabilidad asociadas a cada una de las variables aleatorias necesarias para la simulación. Aunque en algunos casos se logra contar con datos estadísticos, suele suceder que el formato de almacenamiento o de generación de reportes no es el apropiado para facilitar el estudio. Por ello, es muy importante dedicar el tiempo suficiente a esta actividad. De no contar con la información requerida o en caso de desconfiar de la disponible, será necesario realizar un estudio estadístico del comportamiento de la variable que se desea identificar, para luego incluirla en el modelo. Más adelante se hará el análisis de los datos indispensables para asociar una distribución de probabilidad a una variable aleatoria, así como las pruebas que se le deben aplicar. Al finalizar la recolección y análisis de datos para todas las variables del modelo, se tendrán las condiciones para generar una versión preliminar del problema que se está simulando.
4. **Generación del modelo preliminar.** En esta etapa se integra la información obtenida a partir del análisis de los datos, los supuestos del modelo y todos los datos necesarios para crear un modelo lo más cercano posible a la realidad del problema bajo estudio. En algunos casos —sobre todo cuando se trata del diseño de un nuevo proceso o esquema de trabajo— no se cuenta con información estadística, por lo que debe estimarse un rango de variación o determinar (con ayuda del cliente) valores constantes que permitan realizar el modelado. Si éste es el caso, el encargado de la simulación puede, con base en su experiencia, realizar algunas sugerencias de distribuciones de probabilidad que comúnmente se asocian al tipo de proceso que se desea incluir en el modelo. Al finalizar esta etapa el modelo está listo para su primera prueba: su verificación o, en otras palabras, la comparación con la realidad.
5. **Verificación del modelo.** Una vez que se han identificado las distribuciones de probabilidad de las variables del modelo y se han implantado los supuestos acordados, es necesario realizar un proceso de verificación de datos para comprobar la propiedad de la programación del modelo, y comprobar que todos los parámetros usados en la simulación funcionen correctamente. Ciertos problemas, en especial aquellos que requieren muchas operaciones de programación

o que involucran distribuciones de probabilidad difíciles de programar, pueden ocasionar que el comportamiento del sistema sea muy diferente del que se esperaba. Por otro lado, no se debe descartar la posibilidad de que ocurran errores humanos al alimentar el modelo con la información. Incluso podría darse el caso de que los supuestos iniciales hayan cambiado una o varias veces durante el desarrollo del modelo. Por lo tanto, debemos asegurarnos de que el modelo que se va a ejecutar esté basado en los más actuales.

Una vez que se ha completado la verificación, el modelo está listo para su comparación con la realidad del problema que se está modelando. A esta etapa se le conoce también como validación del modelo.

- 6. Validación del modelo.** El proceso de validación del modelo consiste en realizar una serie de pruebas simultáneas con información de entrada real para observar su comportamiento y analizar sus resultados.

Si el problema bajo simulación involucra un proceso que se desea mejorar, el modelo debe someterse a prueba con las condiciones actuales de operación, lo que nos dará como resultado un comportamiento similar al que se presenta realmente en nuestro proceso. Por otro lado, si se está diseñando un nuevo proceso la validación resulta más complicada. Una manera de validar el modelo en este caso, consiste en introducir algunos escenarios sugeridos por el cliente y validar que el comportamiento sea congruente con las expectativas que se tienen de acuerdo con la experiencia. Cualquiera que sea la situación, es importante que el analista conozca bien el modelo, de manera que pueda justificar aquellos comportamientos que sean contrarios a la experiencia de los especialistas que participan en su validación.

- 7. Generación del modelo final.** Una vez que el modelo se ha validado, el analista está listo para realizar la simulación y estudiar el comportamiento del proceso. En caso de que se desee comparar escenarios diferentes para un mismo problema, éste será el *modelo raíz*; en tal situación, el siguiente paso es la definición de los escenarios a analizar.
- 8. Determinación de los escenarios para el análisis.** Tras validar el modelo es necesario acordar con el cliente los escenarios que se quieren analizar. Una manera muy sencilla de determinarlos consiste en utilizar un escenario pesimista, uno optimista y uno intermedio para la variable de respuesta más importante. Sin embargo, es preciso tomar en cuenta que no todas las variables se comportan igual ante los cambios en los distintos escenarios, por lo que tal vez sea necesario que más de una variable de respuesta se analice bajo las perspectivas pesimista, optimista e intermedia. El riesgo de esta situación radica en que el analista podría realizar un diseño de experimentos capaz de generar una gran cantidad de réplicas, lo que redundaría en un incremento considerable de costo, análisis y tiempo de simulación. Es por ello que muchos paquetes de simulación cuentan con herramientas para realizar este proceso, las cuales eliminan la animación y acortan los tiempos de simulación. Estas herramientas permiten realizar varias réplicas del mismo escenario para obtener resultados con estadísticas importantes respecto de la toma de decisiones (por ejemplo, los intervalos de confianza).

Podemos decir que la realización de un proyecto de simulación implica tres grandes fases. El diseño del modelo del problema a analizar, la construcción del modelo y la experimentación que se puede realizar con éste. La figura 1.5 ilustra una manera alternativa de presentar este mismo proceso.^[6]

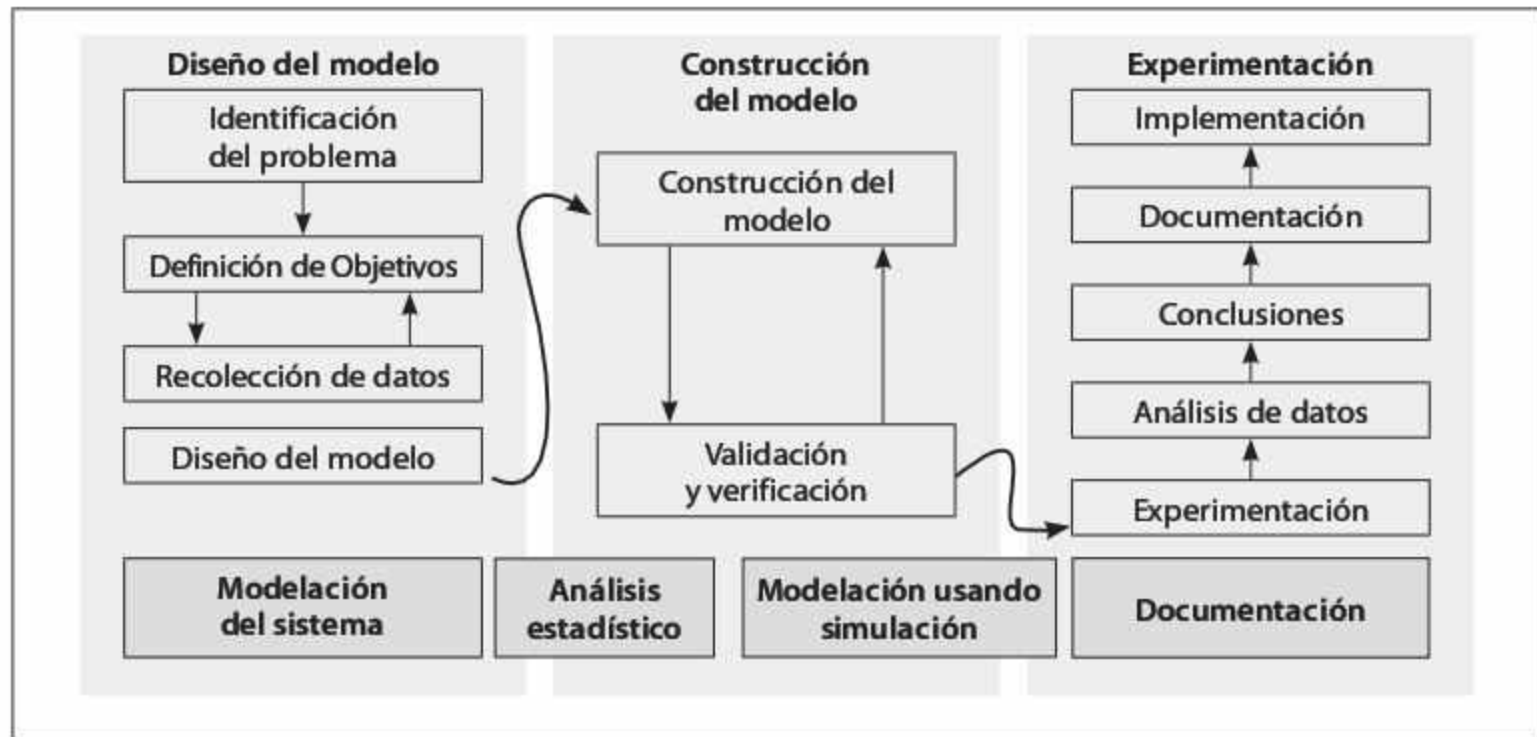


Figura 1.5 Representación del ciclo de un proyecto de simulación.

Como podemos observar en la figura 1.5, la construcción del modelo es sólo una parte del beneficio de una herramienta como simulación. También se puede usar para hacer análisis con diferentes escenarios y tomar decisiones sustentadas estadísticamente. En el siguiente capítulo revisaremos cómo se modela la variabilidad natural de los procesos.

1.6 Problemas

1. Determine los elementos de cada uno de los siguientes sistemas, de acuerdo con lo que se comentó en la sección 1.2.
 - a) La sala de emergencia de un hospital
 - b) Un banco mercantil
 - c) Una línea telefónica de atención a clientes
 - d) La recepción de un hotel
 - e) Un taller de tornos
 - f) El proceso de pintura de un automóvil
 - g) Un hospital
 - h) Un sistema de respuesta en caso de emergencias

2. Defina los elementos de cada uno de estos sistemas, de acuerdo con lo que analizó en la sección 1.2.
 - a) El sistema de mantenimiento de los equipos de una empresa, llevado a cabo por una cuadrilla de personas
 - b) Un aeropuerto
 - c) Una bodega de distribución de productos
 - d) Una línea embotelladora de refrescos
 - e) Un sistema de control de tránsito para la ciudad
 - f) Una línea de armado de refrigeradores
 - g) Una supermercado
 - h) Un taller de mantenimiento de moldes

3. ¿Cuáles podrían ser las entidades de cada uno de los siguientes sistemas?
 - a) Un cajero automático
 - b) Un sistema automático de inspección de botellas
 - c) Una máquina dobladora de lámina
 - d) Un proceso de empaque de televisores
 - e) Una sala de urgencias de un hospital
 - f) Un almacén de producto terminado
 - g) Una línea de embutido de carnes

4. ¿Cuáles podrían ser las entidades de cada uno de los siguientes sistemas?
 - a) Un sistema de distribución de paquetería
 - b) Un sistema de cobranza
 - c) Un conmutador telefónico
 - d) Un departamento de devolución de mercancía
 - e) El abordaje de un crucero por el Caribe
 - f) Un taller de reparación de motores
 - g) Un centro de despacho y asignación de pedidos de productos

5. Determine qué atributos podrían ser relevantes para la simulación de los siguientes sistemas.
 - a) El maquinado de una familia de engranes
 - b) Un proceso de pintura de refrigeradores
 - c) Un sistema de recepción de materia prima
 - d) Un proceso de soldadura para varios productos
 - e) Una sala de tratamientos dentales
 - f) Un sistema de registro de pacientes en un hospital
 - g) Un sistema de localización de mercancía por radio frecuencia

6. ¿Qué atributos podrían ser relevantes para la simulación de los siguientes sistemas?
 - a) Un proceso de empaque de 10 productos por caja, donde cada producto es diferente
 - b) Un proceso de separación de 3 productos para enviarlos a sus respectivas áreas de procesamiento

- c) Un sistema de inspección de calidad de piezas maquinadas
 - d) Un sistema de programación de mantenimiento que califica sus trabajos como urgentes y no urgentes, además de asignarles etiquetas de "Pendiente de asignar", "Asignado", "En proceso" y "Terminado"
7. Especifique las variables que podrían ser relevantes en los siguientes sistemas.
- a) El maquinado de una familia de engranes
 - b) Un proceso de pintura de refrigeradores
 - c) Un sistema de recepción de materia prima
 - d) Un proceso de soldadura para varios productos
 - e) Una sala de tratamientos dentales
 - f) Un sistema de registro de pacientes en un hospital
 - g) Un sistema de localización de mercancía por radio frecuencia
8. De los siguientes sistemas, ¿cuáles podrían ser clasificados como eventos?
- a) Una sala de urgencias de un hospital
 - b) Un sistema de emergencias epidemiológicas
 - c) Un sistema de control de calidad de una planta que fabrica botellas
 - d) Un restaurante de comida rápida
 - e) Una terminal de camiones de carga
 - f) Una aduana comercial de importaciones y exportaciones
9. Determine el promedio móvil de los números de la siguiente tabla y grafique los promedios. ¿Llega a estado estable la gráfica? En caso afirmativo, ¿a partir de qué valor se puede considerar el inicio del estado estable?

0.6435	0.9849	0.9152	0.8327	0.2803	0.1730	0.9002	0.1853	0.3499	0.7368
0.0168	0.1133	0.5673	0.5013	0.0330	0.9814	0.7602	0.1865	0.5518	0.1064
0.3553	0.3846	0.3063	0.1319	0.3769	0.3809	0.5290	0.8586	0.6225	0.5425
0.1242	0.2806	0.9285	0.4257	0.5007	0.9997	0.2072	0.0580	0.5460	0.3910
0.4006	0.2376	0.3883	0.7998	0.9111	0.5554	0.6080	0.6724	0.0332	0.9451
0.2944	0.5657	0.4072	0.6198	0.6809	0.7154	0.8810	0.3028	0.5950	0.3131
0.1438	0.7546	0.0982	0.4946	0.1837	0.5438	0.6598	0.6460	0.8039	0.1599
0.7612	0.8071	0.5163	0.5810	0.6720	0.6020	0.0120	0.4502	0.4228	0.2734
0.4776	0.1012	0.0935	0.4389	0.7195	0.7738	0.8939	0.5225	0.1220	0.8265
0.6031	0.9288	0.1209	0.5537	0.1219	0.9657	0.9734	0.9955	0.2281	0.1084

Promedio móvil: $r_n = \frac{1}{n} \sum_{i=1}^n r_i$ para $n = 1, 2, \dots, 100$

10. Determine el promedio móvil de los números de la siguiente tabla y grafique los promedios. ¿Llega a estado estable la gráfica? En caso afirmativo, ¿a partir de qué valor se puede considerar el inicio del estado estable?

0.1762	0.0477	0.5245	0.6735	0.9922	0.3669	0.1380	0.6584	0.5371	0.4580
0.9750	0.7266	0.2094	0.5885	0.5842	0.5057	0.2614	0.6131	0.8510	0.9502
0.9770	0.6959	0.2955	0.4447	0.2856	0.3545	0.2401	0.5406	0.0547	0.4552
0.4181	0.7080	0.5093	0.1922	0.0685	0.3380	0.7969	0.3670	0.4124	0.9608
0.4409	0.8448	0.4257	0.0763	0.0513	0.8583	0.9419	0.8389	0.0096	0.0633
0.6124	0.8186	0.1288	0.8095	0.1313	0.8238	0.9628	0.0736	0.8992	0.3657
0.7017	0.8310	0.2849	0.5471	0.3716	0.7481	0.0009	0.0936	0.2608	0.5415
0.9343	0.5679	0.0116	0.3081	0.6192	0.4047	0.9659	0.5638	0.7183	0.5649
0.1087	0.8235	0.9399	0.6169	0.0711	0.3051	0.4250	0.5276	0.2523	0.1242
0.8740	0.0961	0.2166	0.5799	0.2477	0.1443	0.4937	0.7373	0.9575	0.1706

Promedio móvil: $r_n = \frac{1}{n} \sum_{i=1}^n r_i$ para $n = 1, 2, \dots, 100$

11. Genere en una hoja de cálculo 100 números con la función $x_i = -3 \ln(1 - r_i)$; donde r_i es un número pseudoaleatorio entre cero y uno, obtenido a partir de la función ALEATORIO de la hoja de cálculo. Suponga que estos valores son tiempos de proceso de cierta pieza. Determine un promedio móvil de estos valores conforme se va realizando el procesamiento de las piezas, y grafique ese promedio. ¿El tiempo promedio de proceso es estable? ¿Y si en lugar de 100 se generan 200 números? (*Sugerencia:* Para evitar que se recalculen los números aleatorios es necesario copiarlos y pegarlos mediante pegado especial de sólo valores).
12. En una hoja de cálculo genere 100 números con la función $x_i = 5 + 10r_i$; donde r_i es un número pseudoaleatorio entre cero y uno, obtenido a partir de la función ALEATORIO de la hoja de cálculo. Suponga que estos valores son tiempos de atención a clientes en un banco. Determine un promedio móvil de estos valores conforme se va realizando la atención de los clientes, y grafique ese promedio. ¿El tiempo promedio de atención a clientes es estable? ¿Qué pasa si ahora se generan 200 números?
13. Genere en una hoja de cálculo números con las siguientes funciones:
- $x_i = -20 * \ln(r_i * r_j)$
 - $x_i = -10 * \ln(r_i * r_j * r_k * r_l)$
 - $x_i = -5 * \ln(r_i * r_j * r_k * r_l * r_m * r_n * r_o * r_p)$

Donde $r_i, r_j, r_k, r_l, r_m, r_n, r_o, r_p$ son números pseudoaleatorios entre cero y uno, obtenidos a partir de la función ALEATORIO de la hoja de cálculo. Suponga que estos valores son tiempos de reparación de una máquina. Determine un promedio móvil de estos valores conforme se van realizando mantenimientos, y grafique ese promedio. ¿Cuál de las tres funciones llega a estado estable más rápidamente? ¿En qué punto del tiempo llega cada una a estado estable? ¿A qué valor converge el promedio en cada una?

Referencias

- [1] Goodsell, C. A. y Van Kely, T. J. (2004). "Inventory Management Simulation at CAT Logistics", En J. A. Joines, R. R. Barton, K. Kang, y P. A. Fishwick (Eds), pp 1185-1190.
- [2] Chan, F.T.S., N. K. H., Tang, H. C. W., Lau, y R. W. L., Ip (2002). "A Simulation Approach in Supply Chain Management", *Integrated Manufacturing Systems*, volumen 13 (número 2), pp 117-122.
- [3] O'Kane, J. (2004). "Simulating production performance: cross case analysis and policy implications", *Industrial Management and Data Systems*, volumen 104 (número 4), pp. 309-321.
- [4] Mahanti, R. y Antony, J (2005). "Confluence of Six Sigma, Simulation and Software Development", *Managerial Auditing Journal*, volumen 20 (número 7), pp. 739-762.
- [5] Strauss, U. (2006). "Using a Business Simulation to Develop Key Skills – the MERKIS Experience", *Industrial and Commercial Training*, volumen 38 (número 4), pp. 213-216.
- [6] García, H. y Centeno, M. A. (2009). "S.U.C.C.E.S.S.F.U.L.: a Framework for Designing Discrete Event Simulation Course", *Proceedings of the 2009 Winter Simulation Conference*, M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin y R. G. Ingalls (Eds), pp. 289-298.



Capítulo 2

Números pseudoaleatorios

- 2.1** Los números pseudoaleatorios
 - 2.2** Generación de números pseudoaleatorios
 - 2.3** Propiedades de los números pseudoaleatorios entre 0 y 1
 - 2.4** Pruebas estadísticas para los números pseudoaleatorios
 - 2.5** Problemas
- Referencias

2.1 Los números pseudoaleatorios

Para poder realizar una simulación que incluya variabilidad dentro de sus eventos, es preciso generar una serie de números que sean aleatorios por sí mismos, y que su aleatoriedad se extrapole al modelo de simulación que se está construyendo. Como puede comprender, en la construcción del modelo los números aleatorios juegan un papel relevante.

Así, una de las primeras tareas que es necesario llevar a cabo consiste en determinar si los números que utilizaremos para “correr” o ejecutar la simulación son realmente aleatorios o no. Por desgracia, precisar lo anterior con absoluta certidumbre resulta muy complicado, ya que para ello tendríamos que generar un número infinito de valores que nos permitiera comprobar la inexistencia de correlaciones entre ellos. Esto sería muy costoso y tardado, además volvería impráctico el uso de la simulación aun con las computadoras más avanzadas.

A pesar de lo anterior, podemos asegurar que el conjunto de números que utilizaremos en una simulación se comporta de manera muy similar a un conjunto de números totalmente aleatorios; por ello es que se les denomina *números pseudoaleatorios*. Casi todas las aplicaciones comerciales tienen varios generadores de números pseudoaleatorios que pueden generar un conjunto muy grande de números sin mostrar correlación entre ellos. En el presente capítulo discutiremos algunos de los métodos de generación de números pseudoaleatorios, y precisaremos qué características deben tener para emplearlos como una fuente confiable de variabilidad dentro de los modelos. Asimismo, se mostrarán algunas de las pruebas más comunes para comprobar qué tan aleatorios son los números obtenidos con dichos generadores.

2.2 Generación de números pseudoaleatorios

Para realizar una simulación se requieren números aleatorios en el intervalo $(0,1)$, a los cuales se hará referencia como r_i , es decir, una secuencia $r_i = \{r_1, r_2, r_3, \dots, r_n\}$ que contiene “ n ” números, todos ellos diferentes. El valor “ n ” recibe el nombre de *periodo o ciclo de vida* del generador que creó la secuencia r_i .

Los r_i constituyen la parte medular de la simulación de procesos estocásticos, y por lo regular, se usan para generar el comportamiento de variables aleatorias, tanto continuas como discretas. Debido a que no es posible generar números realmente aleatorios, consideramos los r_i como números pseudoaleatorios generados por medio de algoritmos determinísticos que requieren parámetros de arranque.

Para simular el comportamiento de una o más variables aleatorias es necesario contar con un conjunto suficientemente grande de r_i que permita, por ejemplo, que la secuencia tenga al menos un periodo de vida de $n = 2^{31} = 2,147,483,648$. De acuerdo con L’Ecuyer⁽⁴⁾ una secuencia de r_i con periodo de vida de $n = 2^{31}$ es relativamente pequeña; de hecho, incluso una secuencia de r_i que contenga un ciclo de vida de $n = 2^{64}$ se considera pequeña. En la actualidad, contamos ya con generadores y procesadores capaces de construir una secuencia de r_i con periodo de vida de $n = 2^{200}$.

Tal vez se preguntará; ¿por qué debe interesarnos construir una secuencia de números r_i lo bastante grande? A continuación ilustraremos la razón mediante un ejemplo.

Suponga que queremos simular el tiempo de atención a clientes en un banco que tiene 5 cajeros en paralelo, cada uno de los cuales atiende alrededor de 50 clientes diarios. Para simular el tiempo de atención se requiere un generador de variable aleatoria en función de r_i , como por ejemplo $T_i = 5 + 2r_i$, expresado en minutos para toda $i = 1, 2, 3, \dots, n$. (El tema de generadores de variables aleatorias se presenta en el capítulo 3). Si simulamos el tiempo de atención de manera aislada, es decir, sin considerar el tiempo transcurrido desde la llegada de éstos, serán necesarios $5 \times 50 = 250$ números r_i para simular un día; si deseáramos simular 5 días se necesitarían $250 \times 5 = 1250$ r_i . Ahora bien, si consideramos el tiempo desde la llegada de los clientes, precisaríamos de 250 r_i para simular el tiempo transcurrido desde la llegada al banco de los 250 clientes por día, y $250 \times 5 = 1250$ r_i para simular el correspondiente al total de clientes atendidos durante 5 días. Por lo tanto, se requerirán 2500 números pseudoaleatorios r_i para simular la operación del banco durante 5 días.

Como se mencionó en el capítulo 1, los resultados no pueden basarse en una sola simulación del sistema; por el contrario, es necesario realizar varias *réplicas* de la misma, corriendo cada una de ellas con números pseudoaleatorios diferentes. Si retomamos el ejemplo del banco, simular 5 días otra vez significa que necesitamos otros 2500 números pseudoaleatorios en el intervalo $(0,1)$. En consecuencia, se requieren 5000 r_i para realizar la simulación del sistema de atención a clientes con dos réplicas.

Usted podrá imaginar cuántos números r_i serán necesarios para simular la operación del banco durante un año con 9 réplicas, o cuántos números r_i se requieren para simular un sistema productivo durante un año con varias líneas de producción, y cada una con varias estaciones, y cada estación con uno o más procesos.

Dada la importancia de contar con un conjunto de r_i suficientemente grande, en esta sección se presentan diferentes algoritmos determinísticos para obtenerlo. Por otra parte, es conveniente señalar que el conjunto de r_i debe ser sometido a una variedad de pruebas para verificar si los números que lo conforman son realmente independientes y uniformes. En la sección 2.4 veremos las pruebas estadísticas que determinan si un conjunto r_i tiene las propiedades de independencia y uniformidad. Una vez generado el conjunto r_i mediante un algoritmo determinístico, es necesario someterlo a las pruebas antes mencionadas: si las supera, podrá utilizarse en la simulación; de lo contrario, simplemente deberemos desecharlo.

Un conjunto de r_i debe seguir una distribución uniforme continua, la cual está definida por:

$$f(r) = \begin{cases} 1, & 0 \leq r \leq 1 \\ 0, & \text{en cualquier otro valor} \end{cases}$$

Generar un conjunto de r_i es una tarea relativamente sencilla; para ello, el lector sólo tiene que diseñar su propio algoritmo de generación. Lo que resulta difícil es diseñar un algoritmo que genere un conjunto de r_i con periodo de vida lo bastante grande (N), y que además pase sin problema las pruebas de uniformidad e independencia, lo cual implica evitar problemas como éstos:

- Que los números del conjunto r_i no estén uniformemente distribuidos, es decir, que haya demasiados r_i en un subintervalo y en otro muy pocos o ninguno.

- Que los números r_i generados sean discretos en lugar de continuos.
- Que la media del conjunto sea muy alta o muy baja, es decir, que esté por arriba o por debajo de $\frac{1}{2}$.
- Que la varianza del conjunto sea muy alta o muy baja, es decir, que se localice por arriba o por debajo del $\frac{1}{2}$ (la obtención de estos valores se discute en la sección 2.3).

En ocasiones se presentan también anomalías como números r_i seguidos por arriba o por debajo de la media; secuencia de r_i por arriba de la media, seguida de una secuencia por debajo de la media, y viceversa, o varios r_i seguidos en forma ascendente o descendente.

A continuación se presentan diferentes algoritmos determinísticos para generar los r_i , los cuales se clasifican en algoritmos no congruenciales y congruenciales. Los algoritmos no congruenciales que analizaremos en esta obra son: cuadrados medios, productos medios y multiplicador constante. Entre los algoritmos congruenciales se encuentran los algoritmos congruenciales lineales y los no lineales. En este libro abordaremos los algoritmos congruenciales lineales —tales como algoritmo congruencial lineal, multiplicativo y aditivo—, y los algoritmos no lineales, como el algoritmo de Blum, Blum y Shub, y el congruencial cuadrático.

2.2.1 Algoritmo de cuadrados medios

Este algoritmo no congruencial fue propuesto en la década de los cuarenta del siglo xx por Von Neumann y Metropolis.^[1] Requiere un número entero detonador (llamado *semilla*) con D dígitos, el cual es elevado al cuadrado para seleccionar del resultado los D dígitos del centro; el primer número r_i se determina simplemente anteponiendo el "0." a esos dígitos. Para obtener el segundo r_i se sigue el mismo procedimiento, sólo que ahora se elevan al cuadrado los D dígitos del centro que se seleccionaron para obtener el primer r_i . Este método se repite hasta obtener n números r_i . A continuación se presentan con más detalle los pasos para generar números con el algoritmo de cuadrados medios.

1. Seleccionar una semilla (X_0) con D dígitos ($D > 3$).
2. Sea Y_0 = resultado de elevar X_0 al cuadrado; sea X_1 = los D dígitos del centro, y sea r_1 = 0. D dígitos del centro.
3. Sea Y_i = resultado de elevar X_i al cuadrado; sea X_{i+1} = los D dígitos del centro, y sea r_i = 0. D dígitos del centro para toda $i = 1, 2, 3, \dots, n$.
4. Repetir el paso 3 hasta obtener los n números r_i deseados.

Nota: Si no es posible obtener los D dígitos del centro del número Y_i , agregue ceros a la izquierda del número Y_i .

Para ilustrar la mecánica del algoritmo de cuadrados medios se presenta el siguiente ejemplo.

Ejemplo 2.1

Generar los primeros 5 números r_i a partir de una semilla $X_0 = 5735$, de donde se puede observar que $D = 4$ dígitos.

Solución:

$$\begin{array}{lll} Y_0 = (5735)^2 = 32890225 & X_1 = 8902 & r_1 = 0.8902 \\ Y_1 = (8902)^2 = 79245604 & X_2 = 2456 & r_2 = 0.2456 \\ Y_2 = (2456)^2 = 06031936 & X_3 = 0319 & r_3 = 0.0319 \\ Y_3 = (0319)^2 = 101761 & X_4 = 0176 & r_4 = 0.0176 \\ Y_4 = (0176)^2 = 030976 & X_5 = 3097 & r_5 = 0.3097 \end{array}$$

El algoritmo de cuadrados medios generalmente es incapaz de generar una secuencia de r_i con periodo de vida n grande. Además, en ocasiones sólo es capaz de generar un número, por ejemplo, si $X_0 = 1000$, entonces $X_1 = 0000$; $r_1 = 0.0000$ y se dice que el algoritmo se degenera con la semilla de $X_0 = 1000$.

2.2.2 Algoritmo de productos medios

La mecánica de generación de números pseudoaleatorios de este algoritmo no congruencial es similar a la del algoritmo de cuadrados medios. La diferencia entre ambos radica en que el algoritmo de productos medios requiere dos semillas, ambas con D dígitos; además, en lugar de elevarlas al cuadrado, las semillas se multiplican y del producto se seleccionan los D dígitos del centro, los cuales formarán el primer número pseudoaleatorio $r_1 = 0.D$ dígitos. Después se elimina una semilla, y la otra se multiplica por el primer número de D dígitos, para luego seleccionar del producto los D dígitos que conformarán un segundo número r_2 . Entonces se elimina la segunda semilla y se multiplican el primer número de D dígitos por el segundo número de D dígitos; del producto se obtiene el tercer número r_3 . Siempre se irá eliminando el número más antiguo, y el procedimiento se repetirá hasta generar los n números pseudoaleatorios. A continuación se presentan con más detalle los pasos del método para generar números con el algoritmo de producto medios.

1. Seleccionar una semilla (X_0) con D dígitos ($D > 3$)
2. Seleccionar una semilla (X_1) con D dígitos ($D > 3$)
3. Sea $Y_0 = X_0 * X_1$; sea $X_2 =$ los D dígitos del centro, y sea $r_1 = 0.D$ dígitos del centro.
4. Sea $Y_1 = X_1 * X_2$; sea $X_3 =$ los D dígitos del centro, y sea $r_2 = 0.D$ dígitos del centro para toda $i = 1, 2, 3, \dots, n$.
5. Repetir el paso 4 hasta obtener los n números r_i deseados.

Nota: Si no es posible obtener los D dígitos del centro del número Y_i , agregue ceros a la izquierda del número Y_i .

Para ilustrar la mecánica del algoritmo de productos medios se presenta el siguiente ejemplo.

Ejemplo 2.2

Generar los primeros 5 números r_i a partir de las semillas $X_0 = 5015$ y $X_1 = 5734$; observe que ambas semillas tienen $D = 4$ dígitos.

Solución:

$Y_0 = (5015)(5734) = 28756010$	$X_2 = 7560$	$r_1 = 0.7560$
$Y_1 = (5734)(7560) = 43349040$	$X_3 = 3490$	$r_2 = 0.3490$
$Y_2 = (7560)(3490) = 26384400$	$X_4 = 3844$	$r_3 = 0.3844$
$Y_3 = (3490)(3844) = 13415560$	$X_5 = 4155$	$r_4 = 0.4155$
$Y_4 = (3844)(4155) = 15971820$	$X_6 = 9718$	$r_5 = 0.9718$

2.2.3 Algoritmo de multiplicador constante

Este algoritmo no congruencial es similar al algoritmo de productos medios. Los siguientes son los pasos necesarios para generar números pseudoaleatorios con el algoritmo de multiplicador constante.

1. Seleccionar una semilla (X_0) con D dígitos ($D > 3$).
2. Seleccionar una constante (a) con D dígitos ($D > 3$).
3. Sea $Y_0 = a * X_0$; sea X_1 los D dígitos del centro, y sea $r_1 = 0.D$ dígitos del centro.
4. Sea $Y_i = a * X_i$; sea X_{i+1} los D dígitos del centro, y sea $r_{i+1} = 0.D$ dígitos del centro para toda $i = 1, 2, 3, \dots, n$.
5. Repetir el paso 4 hasta obtener los n números r_i deseados.

Nota: Si no es posible obtener los D dígitos del centro del número Y_i , agregue ceros a la izquierda del número Y_i .

Para ilustrar la mecánica del algoritmo de multiplicador constante se presenta el siguiente ejemplo.

Ejemplo 2.3

Generar los primeros 5 números r_i a partir de la semilla $X_0 = 9803$ y con la constante $a = 6965$. Observe que tanto la semilla como la constante tienen $D = 4$ dígitos.

Solución:

$Y_0 = (6965)(9803) = 68277895$	$X_1 = 2778$	$r_1 = 0.2778$
$Y_1 = (6965)(2778) = 19348770$	$X_2 = 3487$	$r_2 = 0.3487$
$Y_2 = (6965)(3487) = 24286955$	$X_3 = 2869$	$r_3 = 0.2869$
$Y_3 = (6965)(2869) = 19982585$	$X_4 = 9825$	$r_4 = 0.9825$
$Y_4 = (6965)(9825) = 68431125$	$X_5 = 4311$	$r_5 = 0.4311$

2.2.4 Algoritmo lineal

Este algoritmo congruencial fue propuesto por D. H. Lehmer^[5] en 1951. Según Law y Kelton,^[3] no ha sido el más usado. El algoritmo congruencial lineal genera una secuencia de números enteros por medio de la siguiente ecuación recursiva:

$$X_{i+1} = (ax_i + c) \bmod(m) \quad i = 0, 1, 2, 3, \dots, n$$

donde X_0 es la semilla, a es la constante multiplicativa, c es una constante aditiva, y m es el módulo. $X_0 > 0$, $a > 0$, $c > 0$ y $m > 0$ deben ser números enteros. La operación "mod (m)" significa multiplicar X_i por a , sumar c , y dividir el resultado entre m para obtener el residuo X_{i+1} . Es importante señalar que la ecuación recursiva del algoritmo congruencial lineal genera una secuencia de números enteros $S = \{0, 1, 2, 3, \dots, m - 1\}$, y que para obtener números pseudoaleatorios en el intervalo (0,1) se requiere la siguiente ecuación:

$$r_i = \frac{X_i}{m-1} \quad i = 0, 1, 2, 3, \dots, n$$

Analice el siguiente ejemplo para comprender mejor la mecánica del algoritmo congruencial lineal:

Ejemplo 2.4

Generar 4 números entre 0 y 1 con los siguientes parámetros: $X_0 = 37$, $a = 19$, $c = 33$ y $m = 100$.

Solución:

$$\begin{array}{ll} X_1 = (19 \cdot 37 + 33) \bmod 100 = 36 & r_1 = 36/99 = 0.3636 \\ X_2 = (19 \cdot 36 + 33) \bmod 100 = 17 & r_2 = 17/99 = 0.1717 \\ X_3 = (19 \cdot 17 + 33) \bmod 100 = 56 & r_3 = 56/99 = 0.5656 \\ X_4 = (19 \cdot 56 + 33) \bmod 100 = 97 & r_4 = 97/99 = 0.9797 \end{array}$$

En el ejemplo anterior se dieron de manera arbitraria cada uno de los parámetros requeridos: X_0 , a , c , m . Sin embargo, para que el algoritmo sea capaz de lograr el máximo periodo de vida N , es preciso que dichos parámetros cumplan ciertas condiciones. Banks, Carson, Nelson y Nicol^[1] sugieren lo siguiente:

$$\begin{array}{l} m = 2^g \\ a = 1 + 4k \\ k \text{ debe ser entero} \\ c \text{ relativamente primo a } m \\ g \text{ debe ser entero} \end{array}$$

Bajo estas condiciones se obtiene un periodo de vida máximo: $N = m = 2^g$. Veamos un ejemplo más, tomando en cuenta lo anterior.

Ejemplo 2.5

Generar suficientes números entre 0 y 1 con los parámetros $X_0 = 6$, $k = 3$, $g = 3$ y $c = 7$, hasta encontrar el periodo de vida máximo (N).

Como podemos ver, si se cumplen las condiciones que Banks, Carson, Nelson y Nicol sugieren, se logrará el periodo máximo $N = m = 8$. A continuación se presenta el desarrollo de la generación de los números r_i .

$$a = 1 + 4(3) = 13 \text{ y } m = 2^3 = 8$$

$X_0 = 6$	
$X_1 = (13 \cdot 6 + 7) \bmod 8 = 5$	$r_1 = 5/7 = 0.714$
$X_2 = (13 \cdot 5 + 7) \bmod 8 = 0$	$r_2 = 0/7 = 0.000$
$X_3 = (13 \cdot 0 + 7) \bmod 8 = 7$	$r_3 = 7/7 = 1.000$
$X_4 = (13 \cdot 7 + 7) \bmod 8 = 2$	$r_4 = 2/7 = 0.285$
$X_5 = (13 \cdot 2 + 7) \bmod 8 = 1$	$r_5 = 1/7 = 0.142$
$X_6 = (13 \cdot 1 + 7) \bmod 8 = 4$	$r_6 = 4/7 = 0.571$
$X_7 = (13 \cdot 4 + 7) \bmod 8 = 3$	$r_7 = 3/7 = 0.428$
$X_8 = (13 \cdot 3 + 7) \bmod 8 = 6$	$r_8 = 6/7 = 0.857$

Es importante mencionar que el número generado en $X_8 = 6$ es exactamente igual a la semilla X_0 , y si continuáramos generando más números, éstos se repetirían. Además, sabemos que el algoritmo congruencial lineal genera una secuencia de números enteros $S = \{0, 1, 2, 3, \dots, m - 1\}$. Observe que en este caso se genera la secuencia $S = \{0, 1, 2, 3, 4, 5, 6, 7\}$.

Ejemplo 2.6

Consideremos de nuevo el ejemplo anterior, pero tratemos de infringir de manera arbitraria alguna de las condiciones. Supongamos que $a = 12$; se sabe que a no es el resultado de $1 + 4k$, donde k es un entero. Veamos el comportamiento del algoritmo congruencial lineal ante tal cambio.

Solución:

$$a = 1 + 4(3) = 13 \text{ y } m = 2^3 = 8$$

$X_0 = 6$	
$X_1 = (12 \cdot 6 + 7) \bmod 8 = 7$	$r_1 = 7/7 = 1.000$
$X_2 = (12 \cdot 7 + 7) \bmod 8 = 3$	$r_2 = 3/7 = 0.428$
$X_3 = (12 \cdot 3 + 7) \bmod 8 = 3$	$r_3 = 3/7 = 0.428$

El periodo de vida en este caso es $N = 2$, de manera que, como puede ver, el periodo de vida máximo no se logra. Como conclusión tenemos que si no se cumple alguna de las condiciones, el periodo de vida máximo $N = m$ no se garantiza, por lo que el periodo de vida será menor que m .

2.2.5 Algoritmo congruencial multiplicativo

El algoritmo congruencial multiplicativo surge del algoritmo congruencial lineal cuando $c = 0$. Entonces la ecuación recursiva es:

$$X_{i+1} = (aX_i) \bmod (m) \quad i = 0, 1, 2, 3, \dots, n$$

En comparación con el algoritmo congruencial lineal, la ventaja del algoritmo multiplicativo es que implica una operación menos a realizar. Los parámetros de arranque de este algoritmo son X_0 , a y m , los cuales deben ser números enteros y mayores que cero. Para transformar los números X_i en el intervalo $(0, 1)$ se usa la ecuación $r_i = X_i / (m - 1)$. De acuerdo con Banks, Carson, Nelson y Nicol,^[1] las condiciones que deben cumplir los parámetros para que el algoritmo congruencial multiplicativo alcance su máximo periodo N , son:

$$\begin{aligned} m &= 2^g \\ a &= 3 + 8k \quad \text{o} \quad a = 5 + 8k \\ k &= 0, 1, 2, 3, \dots \\ X_0 &= \text{debe ser un número impar} \\ g &= \text{debe ser entero} \end{aligned}$$

A partir de estas condiciones se logra un periodo de vida máximo $N = k/4 = 2^{g-2}$

Ejemplo 2.7

Generar suficientes números entre 0 y 1 con los siguientes parámetros: $X_0 = 17$, $k = 2$ y $g = 5$, hasta encontrar el periodo o ciclo de vida.

Solución:

$$a = 5 + 8(2) = 21 \quad \text{y} \quad m = 32$$

$$\begin{aligned} X_0 &= 17 \\ X_1 &= (21 \cdot 17) \bmod 32 = 5 & r_1 &= 5/31 = 0.612 \\ X_2 &= (21 \cdot 5) \bmod 32 = 9 & r_2 &= 9/31 = 0.2903 \\ X_3 &= (21 \cdot 9) \bmod 32 = 29 & r_3 &= 29/31 = 1.9354 \\ X_4 &= (21 \cdot 29) \bmod 32 = 1 & r_4 &= 1/31 = 0.3225 \\ X_5 &= (21 \cdot 1) \bmod 32 = 21 & r_5 &= 21/31 = 0.6774 \\ X_6 &= (21 \cdot 21) \bmod 32 = 25 & r_6 &= 25/31 = 0.8064 \\ X_7 &= (21 \cdot 25) \bmod 32 = 13 & r_7 &= 13/31 = 0.4193 \\ X_8 &= (21 \cdot 13) \bmod 32 = 17 & r_8 &= 17/31 = 0.5483 \end{aligned}$$

Si la semilla X_0 se repite, volverán a generarse los mismos números. Por lo tanto, el periodo de vida es $n = 8$, el cual corresponde a $N = m/4 = 32/4 = 8$.

Ejemplo 2.8

Ahora bien, si quebrantamos la condición de que la semilla sea un número impar, digamos con $X_0 = 12$, tenemos:

Solución:

$$\begin{aligned} X_0 &= 12 \\ X_1 &= (21 \cdot 12) \bmod 32 = 28 & r_1 &= 28/31 = 0.9032 \\ X_2 &= (21 \cdot 28) \bmod 32 = 12 & r_2 &= 12/31 = 0.3870 \end{aligned}$$

En vista de que la semilla X_0 se repite, volverán a generarse los mismos números. Por lo tanto, el periodo de vida es $N = 2$.

2.2.6 Algoritmo congruencial aditivo

Este algoritmo requiere una secuencia previa de n números enteros $X_1, X_2, X_3, X_4, \dots, X_n$ para generar una nueva secuencia de números enteros que empieza en $X_{n+1}, X_{n+2}, X_{n+3}, X_{n+4}, \dots$. Su ecuación recursiva es:

$$X_i = (X_{i-1} + X_{i-n}) \bmod (m) \quad i = n + 1, n + 2, n + 3, \dots, N$$

Los números r_i pueden ser generados mediante la ecuación

$$r_i = X_i / m - 1$$

Ejemplo 2.9

Generar 7 números pseudoaleatorios entre cero y uno a partir de la siguiente secuencia de números enteros: 65, 89, 98, 03, 69; $m = 100$.

Sean $X_1 = 65, X_2 = 89, X_3 = 98, X_4 = 03, X_5 = 69$. Para generar $r_1, r_2, r_3, r_4, r_5, r_6$ y r_7 antes es necesario generar $X_6, X_7, X_8, X_9, X_{10}, X_{11}, X_{12}$.

Solución:

$$\begin{aligned} X_6 &= (X_5 + X_1) \bmod 100 = (69 + 65) \bmod 100 = 34 & r_1 &= 34/99 = 0.3434 \\ X_7 &= (X_6 + X_2) \bmod 100 = (34 + 89) \bmod 100 = 23 & r_2 &= 23/99 = 0.2323 \\ X_8 &= (X_7 + X_3) \bmod 100 = (23 + 98) \bmod 100 = 21 & r_3 &= 21/99 = 0.2121 \\ X_9 &= (X_8 + X_4) \bmod 100 = (21 + 03) \bmod 100 = 24 & r_4 &= 24/99 = 0.2424 \\ X_{10} &= (X_9 + X_5) \bmod 100 = (24 + 69) \bmod 100 = 93 & r_5 &= 93/99 = 0.9393 \\ X_{11} &= (X_{10} + X_6) \bmod 100 = (93 + 34) \bmod 100 = 27 & r_6 &= 27/99 = 0.2727 \\ X_{12} &= (X_{11} + X_7) \bmod 100 = (27 + 23) \bmod 100 = 50 & r_7 &= 50/99 = 0.5050 \end{aligned}$$

2.2.7 Algoritmos congruenciales no lineales

En esta sección se analizarán dos algoritmos congruenciales no lineales: el congruencial cuadrático y el algoritmo presentado por Blum, Blum y Shub.^[2]

2.2.7.1 Algoritmo congruencial cuadrático

Este algoritmo tiene la siguiente ecuación recursiva:

$$X_{i+1} = (aX_i^2 + bX_i + c) \bmod(m) \quad i = 0, 1, 2, 3, \dots, N$$

En este caso, los números r_i pueden ser generados con la ecuación $r_i = x_i / (m - 1)$. De acuerdo con L'Ecuyer,^[4] las condiciones que deben cumplir los parámetros m , a , b y c para alcanzar un periodo máximo de $N = m$ son:

- $m = 2^g$
- a debe ser un número par
- c debe ser un número impar
- g debe ser entero
- $(b - 1) \bmod 4 = 1$

De esta manera se logra un periodo de vida máximo $N = m$.

Ejemplo 2.10

Generar, a partir del algoritmo congruencial cuadrático, suficientes números enteros hasta alcanzar el periodo de vida, para esto considere los parámetros $X_0 = 13$, $m = 8$, $a = 26$, $b = 27$ y $c = 27$. Como todas las condiciones estipuladas para los parámetros se satisfacen, es de esperarse que el periodo de vida del generador sea $N = m = 8$, tal como podrá comprobar al revisar los cálculos correspondientes, que se presentan a continuación.

Solución:

$$\begin{aligned} X_1 &= (26 \cdot 13^2 + 27 \cdot 13 + 27) \bmod(8) = 4 \\ X_2 &= (26 \cdot 4^2 + 27 \cdot 4 + 27) \bmod(8) = 7 \\ X_3 &= (26 \cdot 7^2 + 27 \cdot 7 + 27) \bmod(8) = 2 \\ X_4 &= (26 \cdot 2^2 + 27 \cdot 2 + 27) \bmod(8) = 1 \\ X_5 &= (26 \cdot 1^2 + 27 \cdot 1 + 27) \bmod(8) = 0 \\ X_6 &= (26 \cdot 0^2 + 27 \cdot 0 + 27) \bmod(8) = 3 \\ X_7 &= (26 \cdot 3^2 + 27 \cdot 3 + 27) \bmod(8) = 6 \\ X_8 &= (26 \cdot 6^2 + 27 \cdot 6 + 27) \bmod(8) = 5 \\ X_9 &= (26 \cdot 5^2 + 27 \cdot 5 + 27) \bmod(8) = 4 \end{aligned}$$

Por otro lado, el algoritmo cuadrático genera una secuencia de números enteros $S = \{0, 1, 2, 3, \dots, m - 1\}$, al igual que el algoritmo congruencial lineal.

2.2.7.2 Algoritmo de Blum, Blum y Shub^[2]

Si en el algoritmo congruencial cuadrático $a = 1$, $b = 0$ y $c = 0$, entonces se construye una nueva ecuación recursiva:

$$X_{i+1} = (X_i^2) \text{mod}(m) \quad i = 0, 1, 2, 3, \dots, n$$

La ecuación anterior fue propuesta por Blum, Blum y Shub^[2] como un nuevo método para generar números que no tienen un comportamiento predecible.

2.3 Propiedades de los números pseudoaleatorios entre 0 y 1

En la sección anterior hablamos de cómo generar números aleatorios usando diferentes métodos. Sin embargo, ¿de qué manera se puede garantizar que tales números son realmente aleatorios entre 0 y 1? ¿Cuáles son las características que los identifican?, ¿cuáles son sus parámetros? La respuesta es muy importante, dado que los números aleatorios serán utilizados en la simulación para generar los valores de cualquier variable aleatoria. En gran medida, conocer las propiedades que deben tener estos números aleatorios garantiza una buena simulación, por ello, se enumeran a continuación.

Media de los aleatorios entre 0 y 1. En vista de que estos números deben tener la misma probabilidad de presentarse, es preciso que su comportamiento muestre una distribución de probabilidad uniforme continua, con límite inferior cero y límite superior uno. La función de densidad de una distribución uniforme es la siguiente:

$$f(x) = \frac{1}{b-a} \quad a \leq x \leq b; \text{ en este caso, } a = 0 \text{ y } b = 1$$

Gráficamente se vería de la siguiente manera:

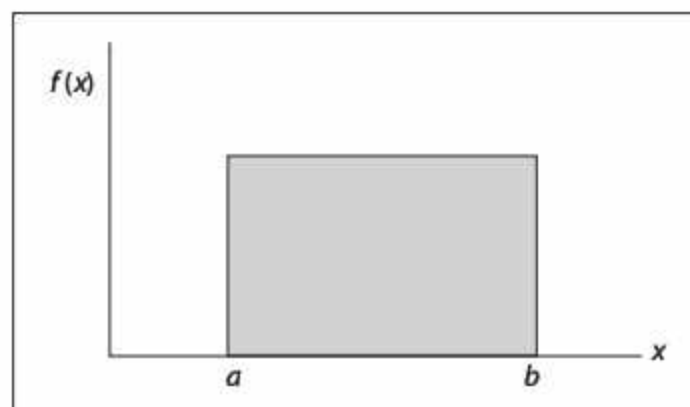


Figura 2.1
Forma general
de la distribución
uniforme entre a y b .

Para obtener la media de la distribución multiplicamos la función de densidad por x , y la integramos en todo el rango de la misma distribución de la siguiente manera:

$$E(x) = \int_a^b f(x) dx = \int_a^b \frac{x}{b-a} dx = \frac{x^2}{2(b-a)} \Big|_a^b$$

Sustituyendo los valores de $a = 0$ y $b = 1$.

$$E(x) = \frac{1}{2}$$

Por lo tanto, el valor esperado (es decir, la media de los números aleatorios entre 0 y 1) es $\mu = 0.5$.

Varianza de los números aleatorios. Si partimos de la misma distribución uniforme continua obtenemos la varianza de la distribución por medio de la ecuación:

$$V(x) = \sigma^2 = E(x^2) - \mu^2$$

Lo que nos da $E(x^2)$:

$$E(x^2) = \int_a^b \frac{1}{b-a} (x)^2 dx = \frac{x^3}{3(b-a)} \Big|_a^b = \frac{(b-a)^3}{3(b-a)} = \frac{(b-a)^2}{3}$$

Al sustituir $a = 0$ y $b = 1$, tenemos que:

$$E(x^2) = \frac{1}{3}$$

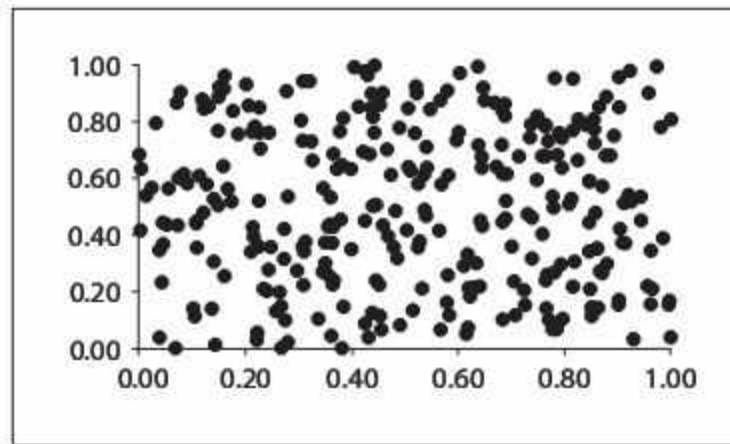
Por lo tanto,

$$V(x) = \frac{1}{3} - \left(\frac{1}{2}\right)^2 = \frac{1}{12}$$

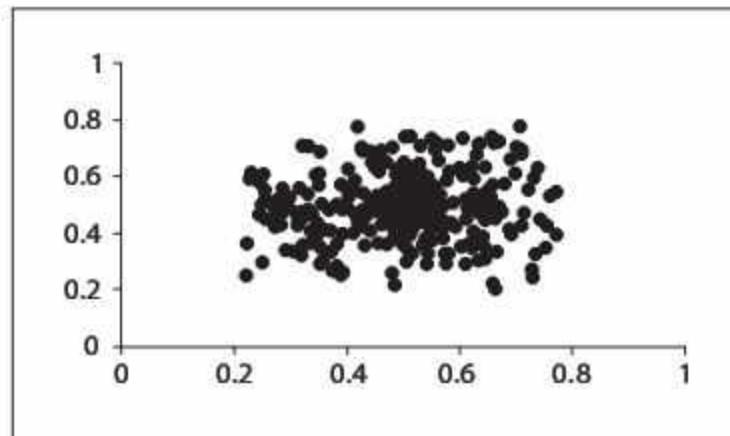
Dados estos resultados podemos decir que los números aleatorios entre 0 y 1 deben tener

$$\mu = \frac{1}{2} \quad \text{y} \quad \sigma^2 = \frac{1}{12}$$

Independencia. Ésta es una propiedad muy importante, e implica que los números aleatorios no deben tener correlación entre sí; es decir, deben ser independientes, de manera que puedan dispersarse de manera uniforme dentro de todo el espectro de valores posibles. La figura 2.2a muestra una gráfica totalmente dispersa en los valores posibles, y la figura 2.2b presenta una acumulación de los valores en la parte central, lo cual quiere decir que hay una correlación entre los mismos.



(a)



(b)

Figura 2.2
Valores uniformemente dispersos y valores correlacionados.

Es posible realizar una serie de pruebas para corroborar que no existe correlación entre los números aleatorios, e incluso para garantizar que no exista un sesgo o tendencia entre los dígitos de cada uno de ellos. Estas pruebas se revisarán con más detalle en la siguiente sección.

2.4 Pruebas estadísticas para los números pseudoaleatorios

En la sección 2.2 se presentaron diversos algoritmos para construir un conjunto r , pero ése es sólo el primer paso, ya que el conjunto resultante debe ser sometido a una serie de pruebas para validar si los números que lo integran son aptos para usarse en un estudio de simulación.

A continuación se analizarán las pruebas estadísticas básicas que se emplean generalmente para determinar si un conjunto de números pseudoaleatorios entre cero y uno cumplen con las propiedades básicas de independencia y uniformidad. El objetivo, en otras palabras, es validar que el conjunto r_i realmente está conformado por números aleatorios. Es importante mencionar que las pruebas que se discutirán no son únicas; si desea conocer otras, consulte Banks, Carson, Nelson y Nicol.^[1]

2.4.1 Prueba de medias

Una de las propiedades que deben cumplir los números del conjunto r_i , es que el valor esperado sea igual a 0.5. La prueba que busca determinar lo anterior es la llamada *prueba de medias*, en la cual se plantean las siguientes hipótesis:

$$H_0: \mu_{r_i} = 0.5$$

$$H_1: \mu_{r_i} \neq 0.5$$

La prueba de medias consiste en determinar el promedio de los n números que contiene el conjunto r_i , mediante la ecuación siguiente:

$$\bar{r} = \frac{1}{n} \sum_{i=1}^n r_i$$

Después se calculan los límites de aceptación inferior y superior con las ecuaciones siguientes:

$$L_{\bar{r}} = \frac{1}{2} - z_{\alpha/2} \left(\frac{1}{\sqrt{12n}} \right)$$

$$L_{S_{\bar{r}}} = \frac{1}{2} + z_{\alpha/2} \left(\frac{1}{\sqrt{12n}} \right)$$

Si el valor de \bar{r} se encuentra entre los límites de aceptación, concluimos que no se puede rechazar que el conjunto r_i tiene un valor esperado de 0.5 con un nivel de aceptación de $1 - \alpha$. En caso contrario se rechaza que el conjunto r_i tiene un valor esperado de 0.5.

Para el cálculo de los límites de aceptación se utiliza el estadístico $z_{\alpha/2}$, el cual se determina por medio de la tabla de la distribución normal estándar (ver apéndice).

Ejemplo 2.11

Considere los 40 números del conjunto r_i que se presenta a continuación, y determine si tienen un valor esperado de $\frac{1}{2}$ con un nivel de aceptación de 95 por ciento.

0.0449	0.1733	0.5746	0.049	0.8406	0.8349	0.92	0.2564
0.6015	0.6694	0.3972	0.7025	0.1055	0.1247	0.1977	0.0125
0.63	0.2531	0.8297	0.6483	0.6972	0.9582	0.9085	0.8524
0.5514	0.0316	0.3587	0.7041	0.5915	0.2523	0.2545	0.3044
0.0207	0.1067	0.3857	0.1746	0.3362	0.1589	0.3727	0.4145

El conjunto r_i contiene 40 números, por lo tanto, $n = 40$. Un nivel de aceptación de 95% implica que $\alpha = 5\%$. Enseguida procedemos a calcular el promedio de los números y los límites de aceptación:

$$\bar{r} = \frac{1}{n} \sum_{i=1}^n r_i = \frac{1}{40} \sum_{i=1}^{40} r_i$$

$$\bar{r} = \frac{1}{40} [0.04487 + 0.17328 + 0.57458 + 0.04901 + \dots + 0.33616 + 0.15885 + 0.37266 + 0.41453]$$

$$\bar{r} = 0.43250$$

$$LI_{\bar{r}} = \frac{1}{2} - z_{\alpha/2} \left(\frac{1}{\sqrt{12n}} \right) = \frac{1}{2} - z_{0.05/2} \left(\frac{1}{\sqrt{12(40)}} \right)$$

$$LI_{\bar{r}} = \frac{1}{2} - (1.96) \left(\frac{1}{\sqrt{12(40)}} \right) = 0.410538649$$

$$LS_{\bar{r}} = \frac{1}{2} + z_{\alpha/2} \left(\frac{1}{\sqrt{12n}} \right) = \frac{1}{2} + z_{0.05/2} \left(\frac{1}{\sqrt{12(40)}} \right)$$

$$LS_{\bar{r}} = \frac{1}{2} + (1.96) \left(\frac{1}{\sqrt{12(40)}} \right) = 0.589461351$$

Como el valor del promedio: $\bar{r} = 0.43250$ se encuentra entre los límites de aceptación, se concluye que no se puede rechazar que el conjunto de 40 números r_i tiene un valor esperado de 0.5, con un nivel de aceptación de 95 por ciento.

2.4.2 Prueba de varianza

Otra de las propiedades que debe satisfacer el conjunto r_i es que sus números tengan una varianza de 1/12. La prueba que busca determinar lo anterior es la *prueba de varianza*, que establece las siguientes hipótesis:

$$H_0: \sigma_{r_i}^2 = 1/12$$

$$H_1: \sigma_{r_i}^2 \neq 1/12$$

La prueba de varianza consiste en determinar la varianza de los n números que contiene el conjunto r_i , mediante la ecuación siguiente:

$$V(r) = \frac{\sum_{i=1}^n (r_i - \bar{r})^2}{n-1}$$

Después se calculan los límites de aceptación inferior y superior con las ecuaciones siguientes:

$$LS_{V(r)} = \frac{\chi^2_{\alpha/2, n-1}}{12(n-1)}$$

$$LJ_{V(r)} = \frac{\chi^2_{(1-\alpha)/2, n-1}}{12(n-1)}$$

Si el valor de $V(r)$ se encuentra entre los límites de aceptación, decimos que no se puede rechazar que el conjunto r_i tiene una varianza de $1/12$, con un nivel de aceptación de $1 - \alpha$; de lo contrario, se rechaza que el conjunto r_i tiene una varianza de $1/12$. Para obtener valores de los factores Chi-cuadrada vea los anexos del libro.

Ejemplo 2.12

Realizar la prueba de varianza a los 40 números r_i del ejemplo 2.11.

Considerando que $n = 40$ y $\alpha = 5\%$, procedemos a calcular la varianza de los números, y los límites de aceptación correspondientes:

$$V(r) = \frac{\sum_{i=1}^n (r_i - \bar{r})^2}{n-1} = \frac{\sum_{i=1}^{40} (r_i - 0.43250)^2}{40-1}$$

$$V(r) = \frac{1}{39} [(0.04487 - 0.43250)^2 + (0.17328 - 0.43250)^2 + \dots + (0.37266 - 0.43250)^2 + (0.41453 - 0.43250)^2]$$

$$V(r) = 0.08695062$$

$$LS_{V(r)} = \frac{\chi^2_{\alpha/2, n-1}}{12(n-1)} = \frac{\chi^2_{0.05/2, 39}}{12(39)} = \frac{58.1200541}{468} = 0.12418815$$

$$LJ_{V(r)} = \frac{\chi^2_{(1-\alpha)/2, n-1}}{12(n-1)} = \frac{\chi^2_{1-0.05/2, 39}}{12(39)} = \frac{23.6543003}{468} = 0.05054338$$

Dado que el valor de la varianza: $V(r) = 0.08695062$ está entre los límites de aceptación, podemos decir que no se puede rechazar que el conjunto de 40 números r_i tiene una varianza de $1/12 = 0.08333$.

2.4.3 Pruebas de uniformidad

Una de las propiedades más importantes que debe cumplir un conjunto de números r_i es la uniformidad. Para comprobar su acatamiento se han desarrollado pruebas estadísticas tales como las pruebas Chi-cuadrada y de Kolmogorov-Smirnov. En cualquiera de ambos casos, para probar la uniformidad de los números de un conjunto r_i es necesario formular las siguientes hipótesis:

$$H_0: r_i \sim U(0,1)$$

$$H_1: r_i \text{ no son uniformes}$$

Veamos a continuación cómo funciona cada una de estas pruebas.

2.4.3.1 Prueba Chi-cuadrada

La prueba Chi-cuadrada busca determinar si los números del conjunto r_i se distribuyen de manera uniforme en el intervalo $(0,1)$. Para llevar a cabo esta prueba es necesario dividir el intervalo $(0,1)$ en m sub-intervalos, en donde es recomendable $m = \sqrt{n}$. Luego se clasifica cada número pseudoaleatorio del conjunto r_i en los m intervalos. A la cantidad de números r_i que se clasifican en cada intervalo se le denomina *frecuencia observada* (O_i), y a la cantidad de números r_i que se espera encontrar en cada intervalo se le llama *frecuencia esperada* (E_i); teóricamente, la E_i es igual n/m . A partir de los valores de O_i y E_i se determina el estadístico χ_0^2 mediante la ecuación

$$\chi_0^2 = \sum_{i=1}^m \frac{(E_i - O_i)^2}{E_i}$$

Si el valor del estadístico χ_0^2 es menor al valor de tablas de $\chi_{\alpha, m-1}^2$, entonces no se puede rechazar que el conjunto de números r_i sigue una distribución uniforme. En caso contrario, se rechaza que r_i sigue una distribución uniforme.

Ejemplo 2.13

Realizar la prueba Chi-cuadrada a los siguientes 100 números de un conjunto r_i , con un nivel de confianza de 95 %.

0.347	0.832	0.966	0.472	0.797	0.101	0.696	0.966	0.404	0.603
0.993	0.371	0.729	0.067	0.189	0.977	0.843	0.562	0.549	0.992
0.674	0.628	0.055	0.494	0.494	0.235	0.178	0.775	0.797	0.252
0.426	0.054	0.022	0.742	0.674	0.898	0.641	0.674	0.821	0.19
0.46	0.224	0.99	0.786	0.393	0.461	0.011	0.977	0.246	0.881
0.189	0.753	0.73	0.797	0.292	0.876	0.707	0.562	0.562	0.821
0.112	0.191	0.584	0.347	0.426	0.057	0.819	0.303	0.404	0.64
0.37	0.314	0.731	0.742	0.213	0.472	0.641	0.944	0.28	0.663
0.909	0.764	0.999	0.303	0.718	0.933	0.056	0.415	0.819	0.444
0.178	0.516	0.437	0.393	0.268	0.123	0.945	0.527	0.459	0.652

Antes de proceder, es recomendable crear una tabla similar a la tabla 2.1, en donde se resumen los pasos que deben llevarse a cabo en la prueba Chi-cuadrada.

Tabla 2.1 Cálculos para la prueba Chi-cuadrada.

Intervalo	O_i	$E_i = \frac{n}{m}$	$\frac{(E_i - O_i)^2}{E_i}$
[0.00-0.10)	7	10	0.9
[0.10-0.20)	9	10	0.1
[0.20-0.30)	8	10	0.4
[0.30-0.40)	9	10	0.1
[0.40-0.50)	14	10	1.6
[0.50-0.60)	7	10	0.9
[0.60-0.70)	11	10	0.1
[0.70-0.80)	14	10	1.6
[0.80-0.90)	9	10	0.1
[0.90-1.00)	12	10	0.4

El estadístico $\chi_0^2 = \sum_{i=1}^{10} \frac{(E_i - O_i)^2}{E_i} = 6.2$ es menor al estadístico correspondiente de la Chi-cuadrada $\chi_{0.05,9}^2 = 16.9$. En consecuencia, no se puede rechazar que los números r_i siguen una distribución uniforme.

2.4.3.2 Prueba Kolmogorov-Smirnov

Propuesta por Kolmogorov y Smirnov, ésta es una prueba estadística que también nos sirve para determinar si un conjunto r_i cumple la propiedad de uniformidad. Es recomendable aplicarla en conjuntos r_i pequeños, por ejemplo, $n < 20$. El procedimiento es el siguiente:

1. Ordenar de menor a mayor los números del conjunto r_i

$$r_1 \leq r_2 \leq r_3 \leq \dots \leq r_n$$

2. Determinar los valores de: D^+ , D^- y D con las siguientes ecuaciones:

$$D^+ = \max_{1 \leq i \leq n} \left\{ \frac{i}{n} - r_i \right\}$$

$$D^- = \max_{1 \leq i \leq n} \left\{ r_i - \frac{i}{n} \right\}$$

$$D = \max(D^+, D^-)$$

3. Determinar el valor crítico $D_{\alpha,n}$ de acuerdo con la tabla de valores críticos de Kolmogorov-Smirnov para un grado de confianza α , y según el tamaño de la muestra n .
4. Si el valor D es mayor que el valor crítico $D_{\alpha,n}$, se concluye que los números del conjunto r_i no siguen una distribución uniforme; de lo contrario se dice que no se ha detectado diferencia significativa entre la distribución de los números del conjunto r_i y la distribución uniforme.

Ejemplo 2.14

Realizar la prueba Kolmogorov-Smirnov, con un nivel de confianza de 90%, al siguiente conjunto r_i de 10 números:

$$r_i = \{0.97, 0.11, 0.65, 0.26, 0.98, 0.03, 0.13, 0.89, 0.21, 0.69\}$$

El nivel de confianza de 90% implica $\alpha = 10\%$. Si se ordenan los números r_i de menor a mayor, la secuencia es:

0.03	0.11	0.13	0.21	0.26	0.65	0.69	0.89	0.97	0.98
------	------	------	------	------	------	------	------	------	------

Para determinar los valores de D^+ , D^- y D es recomendable realizar una tabla como la siguiente.

Tabla 2.2 Cálculos de la prueba Kolmogorov-Smirnov.

i	1	2	3	4	5	6	7	8	9	10
$\frac{i}{n}$	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1.00
r_i	0.03	0.11	0.13	0.21	0.26	0.65	0.69	0.89	0.97	0.98
$\frac{i-1}{n}$	0.00	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90
$\frac{i}{n} - r_i$	0.07	0.09	0.17	0.19	0.24	-0.05	0.01	-0.09	-0.07	0.02
$r_i - \frac{i-1}{n}$	0.03	0.01	-0.07	-0.09	-0.14	0.15	0.09	0.19	0.17	0.08
n	10									
D^+	0.24	D^-	0.19	D	0.24					

De acuerdo con la tabla de valores para la prueba Kolmogorov-Smirnov, el valor crítico $D_{0.10,10}$ correspondiente a $n = 10$ es $D_{0.10,10} = 0.368$, que resulta mayor al valor $D = 0.24$; por lo tanto, se concluye que los números del conjunto r_i se distribuyen uniformemente.

2.4.4 Pruebas de independencia

Recuerde que las dos propiedades más importantes que deben satisfacer los números de un conjunto r_i son uniformidad e independencia. En la sección anterior comentamos las pruebas que buscan determinar si los números del conjunto r_i son uniformes. A continuación hablaremos de las pruebas estadísticas que tratan de corroborar si los números en el intervalo $(0,1)$ son independientes o, en otras palabras, si parecen pseudoaleatorios.

Para probar la independencia de los números de un conjunto r_i , primero es preciso formular las siguientes hipótesis:

H_0 : los números del conjunto r_i son independientes

H_1 : los números del conjunto r_i no son independientes

2.4.4.1 Prueba de corridas arriba y abajo

El procedimiento de esta prueba consiste en determinar una secuencia de números (S) que sólo contiene unos y ceros, de acuerdo con una comparación entre r_i y r_{i-1} . Después se determina el número de corridas observadas, C_0 (una corrida se identifica como la cantidad de unos o ceros consecutivos). Luego se calcula el valor esperado, la varianza del número de corridas y el estadístico Z_0 mediante las ecuaciones:

$$\mu_{C_0} = \frac{2n-1}{3}$$

$$\sigma_{C_0}^2 = \frac{16n-29}{90}$$

$$Z_0 = \left| \frac{C_0 - \mu_{C_0}}{\sigma_{C_0}} \right|$$

Si el estadístico Z_0 es mayor que el valor crítico de $Z_{\alpha/2}$, se concluye que los números del conjunto r_i no son independientes. De lo contrario no se puede rechazar que el conjunto de r_i sea independiente.

Considere el siguiente conjunto r_i de 21 números:

$$r_i = \{0.89, 0.26, 0.01, 0.98, 0.13, 0.12, 0.69, 0.11, 0.05, 0.65, \\ 0.21, 0.04, 0.03, 0.11, 0.07, 0.97, 0.27, 0.12, 0.95, 0.02, 0.06\}$$

La secuencia de unos y ceros se construye de esta manera: se coloca un cero si el número r_i es menor que o igual al número r_i anterior; en caso de ser mayor que el número r_i

anterior, se pone un uno. Si se considera la secuencia de los 21 números del conjunto r_i que se dio antes, la secuencia de unos y ceros es:

$$S = \{0,0,1,0,0,1,0,0,1,0,0,0,1,0,1,0,0,1,0,1\}$$

Observe que la secuencia S contiene $n - 1$ números, en este caso 20. Esto se debe a que el primer número $r_1 = 0.89$ no tiene número anterior con el cual compararlo. Recuerde que una corrida se forma con unos consecutivos o ceros consecutivos. Por ejemplo los primeros dos ceros de la secuencia forman la primer corrida, que se dice que tiene una longitud de dos; el tercer número de la secuencia, uno, forma la segunda corrida con longitud de uno; después siguen dos ceros, los cuales forman la tercera corrida con longitud de dos; después sigue un uno, que forma la cuarta corrida con longitud de uno, etcétera. Mediante el proceso anterior se determina r_i que el número de corridas de la secuencia es $C_0 = 14$.

Ejemplo 2.15

Realizar la prueba de corridas arriba y abajo con un nivel de aceptación de 95% al siguiente conjunto de 40 números r_i :

0.34	0.83	0.96	0.47	0.79	0.99	0.37	0.72	0.06	0.18
0.67	0.62	0.05	0.49	0.59	0.42	0.05	0.02	0.74	0.67
0.46	0.22	0.99	0.78	0.39	0.18	0.75	0.73	0.79	0.29
0.11	0.19	0.58	0.34	0.42	0.37	0.31	0.73	0.74	0.21

Realizaremos la asignación de unos y ceros por renglón (o fila). Por lo tanto, la secuencia S es:

$$S = \{1,1,0,1,1,0,1,0,1,1,0,0,1,1,0,0,0,1,0,0,0,1,0,0,0,1,0,1,0,0,1,1,0,1,0,0,1,1,0\}$$

obteniéndose un valor de $C_0 = 24$ y $\alpha = 5\%$.

A continuación se presentan los cálculos correspondientes al valor esperado y a la varianza del número de corridas:

$$\mu_{C_0} = \frac{2n-1}{3} = \frac{2(40)-1}{3} = 26.333$$

$$\sigma_{C_0}^2 = \frac{16n-29}{90} = \frac{16(40)-29}{90} = 6.788$$

$$Z_0 = \left| \frac{C_0 - \mu_{C_0}}{\sigma_{C_0}} \right| = \left| \frac{24 - 26.333}{\sqrt{6.788}} \right| = 0.8954$$

Como el estadístico Z_0 es menor que el valor de tabla de la normal estándar para $Z_{\alpha/2} = Z_{5\%/2} = 1.96$, se concluye que no se puede rechazar que los números del conjunto r_i son independientes. Es decir, de acuerdo con esta prueba, los números son aptos para usarse en simulación.

2.4.4.2 Prueba de corridas arriba y abajo de la media

El procedimiento de esta prueba consiste en determinar una secuencia de unos y ceros, de acuerdo con una comparación entre los números del conjunto r_i y 0.5. Después se determina el número de corridas observadas, C_o , y los valores de n_0 y n_1 . C_o es el número de corridas en la secuencia, el cual está determinado de la misma manera que en la prueba de corridas arriba y abajo; n_0 es igual a la cantidad de ceros en la secuencia, y n_1 es igual a la cantidad de unos en la secuencia, y se cumple que $n_0 + n_1 = n$. (Recuerde que una corrida se identifica como la cantidad de unos o ceros consecutivos). Luego se calcula el valor esperado, la varianza del número de corridas, y el estadístico Z_0 con las siguientes ecuaciones:

$$\mu_{C_o} = \frac{2n_0n_1}{n} + \frac{1}{2}$$

$$\sigma_{C_o}^2 = \frac{2n_0n_1(2n_0n_1 - n)}{n^2(n-1)}$$

$$Z_0 = \frac{C_o - \mu_{C_o}}{\sigma_{C_o}}$$

Si el estadístico Z_0 está fuera del intervalo: $-Z_{\frac{\alpha}{2}} \leq Z_0 \leq Z_{\frac{\alpha}{2}}$, se concluye que los números del conjunto r_i no son independientes. De lo contrario no se puede rechazar que el conjunto de r_i es independiente.

Considere la siguiente secuencia de 10 números de un conjunto r_i :

$$r_i = \{0.67, 0.62, 0.05, 0.49, 0.59, 0.42, 0.05, 0.02, 0.74, 0.67\}$$

La secuencia de unos y ceros se construye de la siguiente manera: se asigna un uno si el número r_i es mayor que o igual a 0.5. En caso contrario se asignará un cero. Al seguir esta regla, la secuencia de unos y ceros es:

$$S = \{1, 1, 0, 0, 1, 0, 0, 0, 1, 1\}$$

El número de corridas se determina de la misma manera que en la prueba de corridas arriba y abajo. En este caso se tiene que el número de corridas de la secuencia S es $C_o = 5$. Por otra parte, la secuencia tiene 5 ceros y 5 unos, así que $n_0 = 5$ y $n_1 = 5$.

Ejemplo 2.16

Realizar la prueba de corridas arriba y abajo, con un nivel de aceptación de 95%, al siguiente conjunto de 50 números r_i :

0.809	0.042	0.432	0.538	0.225	0.88	0.688	0.772	0.036	0.854
0.397	0.268	0.821	0.897	0.07	0.721	0.087	0.35	0.779	0.482
0.136	0.855	0.453	0.197	0.444	0.799	0.809	0.691	0.545	0.857
0.692	0.055	0.348	0.373	0.436	0.29	0.015	0.834	0.599	0.724
0.564	0.709	0.946	0.754	0.677	0.128	0.012	0.498	0.6	0.913

Construiremos la secuencia de unos y ceros por renglón quedando de la siguiente manera:

$$S = \{1,0,0,1,0,1,1,1,0,1,0,0,1,1,0,1,0,0,1,0,0,1,0,0,0,1,1,1,1,1,1,1,1,0,0,0,1,1\}$$

A partir de la secuencia anterior se determina que hay 21 corridas, 23 ceros y 27 unos. Por lo tanto, $C_0 = 21$, $n_0 = 23$ y $n_1 = 27$. A continuación se presentan los cálculos del valor esperado y de la varianza del número de corridas:

$$\mu_{C_0} = \frac{2n_0n_1}{n} + \frac{1}{2} = \frac{2(23)(27)}{50} + \frac{1}{2} = 25.34$$

$$\sigma_{C_0}^2 = \frac{2n_0n_1(2n_0n_1 - n)}{n^2(n-1)} = \frac{2(23)(27)[2(23)(27) - 50]}{(50)^2(50-1)} = 12.08542$$

$$Z_0 = \frac{C_0 - \mu_{C_0}}{\sigma_{C_0}} = \frac{21 - 25.34}{\sqrt{12.08542}} = -1.2484$$

Como el valor de Z_0 cae dentro del intervalo $-1.96 \leq Z_0 = -1.2484 \leq 1.96$, se dice que no se puede rechazar que los números del conjunto r_i son independientes con un nivel de confianza de 95 %. De acuerdo con esta prueba, el conjunto de números r_i se puede usar en un estudio de simulación.

2.4.4.3 Prueba póker

Esta prueba consiste en visualizar el número r_i con cinco decimales (como si fuera una mano del juego de póker, con 5 cartas), y clasificarlo como: todos diferentes (TD), exactamente un par (1P), dos pares (2P), una tercia (T), una tercia y un par (TP), póker (P) y quintilla (Q). Por ejemplo, si $r_i = 0.69651$ se le clasifica como par, porque hay dos números 6. Ahora bien, consideremos el caso de $r_i = 0.13031$, el cual debe clasificarse como dos pares (dos números 1 y dos números 3). Finalmente, $r_i = 0.98898$ debe clasificarse como una tercia y un par, porque hay tres números 8 y dos números 9. La prueba póker se puede realizar a números r_i con tres, cuatro y cinco decimales. Para r_i con tres decimales sólo hay tres categorías de clasificación: todos diferentes (TD), un par (1P) y una tercia (T). Cuando se consideran r_i con cuatro decimales se cuenta con cinco opciones para clasificar los números: todos diferentes (TD), exactamente un par (1P), dos pares (2P), una tercia (T) y póker (P). Las tablas 2.3 a 2.5 presentan la probabilidad esperada para cada una de las categorías de clasificación de esta prueba para conjuntos r_i que contienen n números con 3, 4 y 5 decimales.

Tabla 2.3 Prueba póker para números con 3 decimales.

Categoría	Probabilidad	E_i
Todos diferentes (TD)	0.72	$0.72n$
Exactamente un par (1P)	0.27	$0.27n$
Tercia (T)	0.01	$0.01n$

Tabla 2.4 Prueba póker para números con 4 decimales.

Categoría	Probabilidad	E_i
Todos diferentes (TD)	0.5040	$0.5040n$
Exactamente 1 par (1P)	0.4320	$0.4320n$
2 pares (2P)	0.0270	$0.0270n$
Tercia (T)	0.0360	$0.0360n$
Póker (P)	0.0010	$0.0010n$

Tabla 2.5 Prueba póker para números con 5 decimales.

Categoría	Probabilidad	E_i
Todos diferentes (TD)	0.3024	$0.3024n$
Exactamente 1 par (1P)	0.5040	$0.5040n$
2 pares (2P)	0.1080	$0.1080n$
1 tercia y 1 par (TP)	0.0090	$0.0090n$
Tercia (T)	0.0720	$0.0720n$
Póker (P)	0.0045	$0.0045n$
Quintilla (Q)	0.0001	$0.0001n$

La prueba póker requiere el estadístico de la distribución Chi-cuadrada $\chi_{\alpha,6}^2$ para números con 5 decimales, $\chi_{\alpha,4}^2$ para números con 4 decimales, y $\chi_{\alpha,2}^2$ para números con 3 decimales. $\chi_{\alpha,6}^2$ tiene 6 grados de libertad, debido a que los números se clasifican en 7 categorías o clases: todos diferentes, exactamente un par, dos pares, una tercia y un par, una tercia, póker y quintilla.

El procedimiento de la prueba consiste en:

- Determinar la categoría de cada número del conjunto r_i .
- Contabilizar los números r_i de la misma categoría o clase para obtener la frecuencia observada (O_i).

c) Calcular el estadístico de la prueba χ_0^2 con la ecuación

$$\chi_0^2 = \sum_{i=1}^m \frac{(E_i - O_i)^2}{E_i},$$

donde E_i es la frecuencia esperada de números r_i en cada categoría, y m representa la cantidad de categorías o clases en las que se clasificaron los números r_i , siendo $m = 7$, $m = 5$, y $m = 3$ los números de categorías para la prueba póker con 5, 4 y 3 decimales, respectivamente. Por último:

d) comparar el estadístico de χ_0^2 con $\chi_{\alpha, m-1}^2$.

Si χ_0^2 es menor que $\chi_{\alpha, m-1}^2$, se dice que no se puede rechazar la independencia de los números del conjunto r_i . En caso contrario la independencia de los números del conjunto r_i se rechaza.

Ejemplo 2.17

Realice la prueba póker, con un nivel de aceptación de 95%, a los siguientes 30 números entre cero y uno, con 5 decimales.

0.06141	0.72484	0.94107	0.56766	0.14411	0.87648
0.81792	0.48999	0.18590	0.06060	0.11223	0.64794
0.52953	0.50502	0.30444	0.70688	0.25357	0.31555
0.04127	0.67347	0.28103	0.99367	0.44598	0.73997
0.27813	0.62182	0.82578	0.85923	0.51483	0.09099

Primero hay que clasificar cada número del conjunto r_i , asignándole las claves que se mencionaron antes. El resultado es el que se muestra en la tabla 2.6:

Tabla 2.6 Clasificación de los números de un conjunto r_i , de acuerdo con la prueba póker.

0.06141	1P	0.72484	1P	0.94107	TD	0.56766	T	0.14411	TP	0.87648	1P
0.81792	TD	0.48999	T	0.18590	TD	0.06060	TP	0.11223	2P	0.64794	1P
0.52953	1P	0.50502	2P	0.30444	T	0.70688	1P	0.25357	1P	0.31555	T
0.04127	TD	0.67347	1P	0.28103	TD	0.99367	1P	0.44598	1P	0.73997	2P
0.27813	TD	0.62182	1P	0.82578	1P	0.85923	TD	0.51483	TD	0.09099	TP

Para seguir con la prueba se recomienda hacer una tabla como la siguiente:

Tabla 2.7 Cálculos de la prueba póker.

Categorías	O_i	E_i	$\frac{(E_i - O_i)^2}{E_i}$
Todos diferentes (TD)	8	$(0.3024)(30) = 9.072$	0.12667
Exactamente 1 par (1P)	12	$(0.5040)(30) = 15.12$	0.64380
2 pares (2P)	3	$(0.1080)(30) = 3.24$	0.01777
1 tercia y 1 Par (TP)	3	$(0.0090)(30) = 0.27$	27.6033
Tercia (T)	4	$(0.0720)(30) = 2.16$	1.56740
Póker (P)	0	$(0.0045)(30) = 0.135$	0.135
Quintilla (Q)	0	$(0.0001)(30) = 0.003$	0.003

El estadístico $\chi_0^2 = \sum_{i=1}^7 \frac{(E_i - O_i)^2}{E_i} = 30.0969$ es mayor que el estadístico correspondiente

de la Chi-cuadrada: $\chi_{0.05, 6}^2 = 12.59$. En consecuencia, se rechaza que los números del conjunto r_i son independientes.

2.4.4.4 Prueba de series

Esta prueba consiste en comparar los números con el propósito de corroborar la independencia entre números consecutivos. Las hipótesis básicas son:

$H_0: r_i \sim$ Independientes

$H_1: r_i \sim$ Dependientes

La prueba funciona de esta manera: se inicia al crear una gráfica de dispersión entre los números consecutivos (r_i, r_{i+1}) ; luego se divide la gráfica en m casillas, como se muestra en la figura 2.3, con m como el valor entero más cercano a \sqrt{n} que permita formar de preferencia, aunque no necesariamente, una matriz cuadrada.

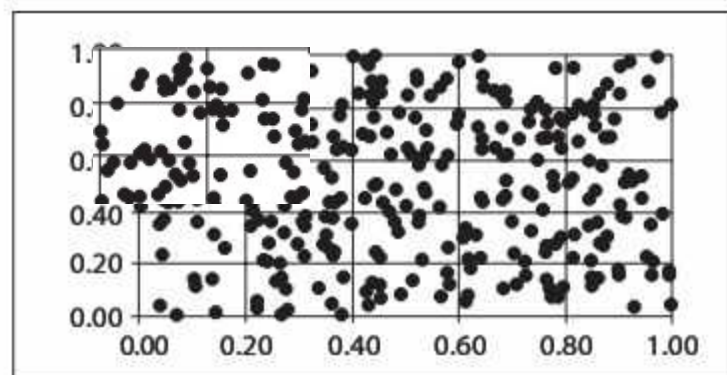


Figura 2.3
Gráfica de dispersión:
primer paso de la prueba
de series.

Enseguida se determina la frecuencia observada O_i al contabilizar el número de puntos en cada casilla y su correspondiente frecuencia esperada E_i , de acuerdo con $E_i = (n - 1)/m$, donde $n - 1$ es el número total de pares ordenados o puntos en la gráfica. Se procede entonces a calcular el error o estadístico de prueba $\chi_0^2 = \sum_{i=1}^m \frac{(E_i - O_i)^2}{E_i}$; finalmente, si el valor del error es menor que o igual al estadístico de tablas $\chi_{\alpha, m-1}^2$, no podemos rechazar la hipótesis de independencia entre números consecutivos.

Ejemplo 2.18

Realice la prueba de series a los siguientes 30 números, con un nivel de confianza de 95 %.

0.872	0.950	0.343	0.058	0.384
0.219	0.041	0.036	0.213	0.946
0.570	0.842	0.706	0.809	0.300
0.618	0.512	0.462	0.005	0.203
0.291	0.151	0.596	0.443	0.868
0.913	0.511	0.586	0.608	0.879

Para empezar, generamos la gráfica de dispersión (vea la figura 2.4) con la secuencia de los 29 pares ordenados $(x, y) = (r_i, r_{i+1})$ siguientes:

- $(r_1, r_2) = (0.872, 0.219)$
- $(r_2, r_3) = (0.219, 0.570)$
- $(r_3, r_4) = (0.570, 0.618)$
- $(r_4, r_5) = (0.618, 0.291)$
- $(r_5, r_6) = (0.291, 0.913)$
- $(r_6, r_7) = (0.913, 0.950)$
- $(r_{28}, r_{29}) = (0.203, 0.868)$
- $(r_{29}, r_{30}) = (0.868, 0.879)$

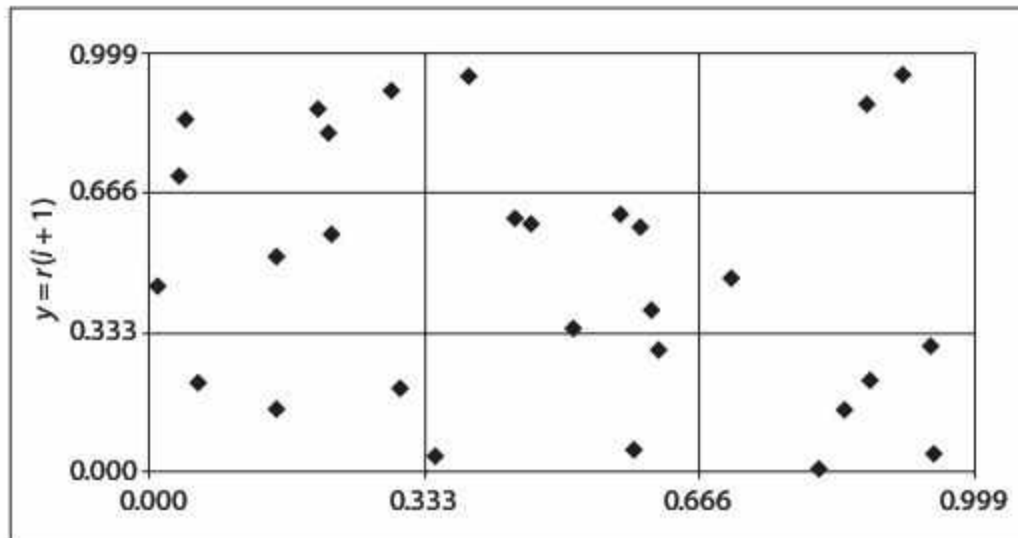


Figura 2.4
Gráfica de dispersión del ejemplo 2.18.

En la tabla 2.8 se presenta el resto del procedimiento: se contabiliza el número de puntos en cada casilla O_i , y se calcula la frecuencia esperada E_i de acuerdo con $E_i = 29/9$; en la última columna se presenta el cálculo del estadístico de prueba

$$\chi_0^2 = \sum_{i=1}^m \frac{(E_i - O_i)^2}{E_i} = \sum_{i=1}^9 \frac{(3.22 - O_i)^2}{3.22}$$

Tabla 2.8 Cálculos de la prueba de series.

Intervalo	O_i	$E_i = \frac{n-1}{m} = \frac{29}{9}$	$\chi_0^2 = \sum_{i=1}^m \frac{(E_i - O_i)^2}{E_i}$
1	3	3.22	0.015
2	3	3.22	0.015
3	5	3.22	0.984
4	3	3.22	0.015
5	6	3.22	2.400
6	1	3.22	1.530
7	5	3.22	0.984
8	1	3.22	1.530
9	2	3.22	0.462
Total	29	29	7.935

El valor de tablas $\chi_{0.05,8}^2 = 15.507$ es mayor que el error total de 7.935, por lo cual no podemos rechazar la hipótesis de independencia.

2.4.4.5 Prueba de huecos

Esta prueba consiste en comparar los números con el propósito de verificar el tamaño del "hueco" que existe entre ocurrencias sucesivas de un número. Las hipótesis fundamentales son:

$$H_0: r_i \sim \text{Independientes}$$

$$H_1: r_i \sim \text{Dependientes}$$

La prueba se inicia al definir un intervalo de prueba (α, β) , donde $(\alpha, \beta) \in (0, 1)$, posteriormente se construye una secuencia de unos y ceros de esta manera: se asigna un uno si el r_i pertenece al intervalo (α, β) , y un 0 si no pertenece a dicho intervalo. Por ejemplo, si se define un intervalo $(\alpha, \beta) = (0.6, 0.7)$ y se tiene la muestra de 10 números

$$r_i = \{0.67, 0.62, 0.05, 0.49, 0.59, 0.42, 0.64, 0.06, 0.74, 0.67\},$$

se asignará un uno si el r_i está entre 0.6 y 0.7; en caso contrario se asignará un cero. Si se sigue la regla anterior, la secuencia binaria es:

$$S = \{1, 1, 0, 0, 0, 0, 1, 0, 0, 1\}$$

El tamaño de hueco i se define como el número de ceros existentes entre unos consecutivos. En el caso de la secuencia de nuestro ejemplo tenemos $h = 3$ huecos, el primero de tamaño 0, el segundo de tamaño 4, y el tercero de tamaño 2 de acuerdo con:

$$S = \left\{ \underbrace{1, 1, 0, 0, 0, 0}_0, \underbrace{1, 0, 0, 1}_2 \right\}$$

A partir del conjunto anterior se determina la frecuencia observada O_i , al contabilizar el número de ocurrencias de cada tamaño de hueco y su correspondiente frecuencia esperada E_i , de acuerdo con $E_i = (h)(\beta - \alpha)(1 - (\beta - \alpha))^i$, donde h es el número total de huecos en la muestra. La frecuencia del último intervalo se puede calcular mediante la diferencia entre el total y la suma de las frecuencias esperadas de los intervalos anteriores. En la tabla 2.9 se muestra un resumen de estos cálculos.

Tabla 2.9 Frecuencias observadas y esperadas en la prueba de huecos.

Tamaño del hueco i	O_i	$E_i = (h)(\beta - \alpha)(1 - (\beta - \alpha))^i$ $E_i = (3)(0.7 - 0.6)(1 - (0.7 - 0.6))^i$	E_i
0	1	$(3)(0.1)(0.9)^0$	0.3
1	0	$(3)(0.1)(0.9)^1$	0.27
2	1	$(3)(0.1)(0.9)^2$	0.243
3	0	$(3)(0.1)(0.9)^3$	0.2187
4	1	$(3)(0.1)(0.9)^4$	0.1968
≥ 5	0	$(3)(0.1)(0.9)^5$	1.7715
Total	$h = 3$	$h = 3$	$h = 3$

Se procede entonces a calcular el error o estadístico de prueba $\chi_0^2 = \sum_{i=1}^m \frac{(E_i - O_i)^2}{E_i}$; por último, si este valor es menor que o igual al estadístico de tablas $\chi_{\alpha, m-1}^2$, no podemos rechazar la hipótesis de la independencia entre los números.

Ejemplo 2.19

Realizar la prueba de huecos a los siguientes 30 números, con un nivel de confianza de 95% para el intervalo $(\alpha - \beta) = (0.8, 1.0)$.

0.872	0.950	0.343	0.058	0.384
0.219	0.041	0.036	0.213	0.946
0.570	0.842	0.706	0.809	0.300
0.618	0.512	0.462	0.005	0.203
0.291	0.151	0.596	0.443	0.868
0.913	0.511	0.586	0.608	0.879

Tomando los números por renglón (o fila) y teniendo en cuenta el intervalo (0.8,1.0), la secuencia de unos y ceros es:

$$S = \{1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 1\}$$

Si calculamos los huecos de la muestra, tenemos:

$$S = \left\{ \underbrace{1, 1, 0, 0, 0, 0, 0, 0, 0, 0}_0, \underbrace{1, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}_1, \underbrace{1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}_1, \underbrace{1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0}_3 \right\}$$

El número de ocurrencias de cada tamaño de hueco O_i , su correspondiente frecuencia esperada E_i y el cálculo del estadístico de prueba se muestran en la tabla 2.10.

Tabla 2.10 Ejemplo de la prueba de huecos.

Tamaño del hueco i	O_i	$E_i = (h)(\beta - \alpha)(1 - (\beta - \alpha))^i$ $E_i = (7)(0.2)(0.8)^i$	$\frac{(E_i - O_i)^2}{E_i}$
0	2	1.4	0.2571
1	2	1.12	0.6914
2	0	0.896	0.896
3	1	0.7168	0.1119
4	0	0.5734	0.5734
≥ 5	2	$7(0.8)^5 = 2.2938$	0.0376
Total	$h = 7$	$h = 7$	2.5675

Ya que el estadístico de prueba $\chi_0^2 = \sum_{i=1}^6 \frac{(E_i - O_i)^2}{E_i} = 2.5675$ es menor que el estadístico de tablas $\chi_{\alpha, m-1}^2 = \chi_{0.05, 5}^2 = 11.07$, no podemos rechazar la hipótesis de independencia entre los números.

2.5 Problemas

- Determine el ciclo o periodo de vida de los siguientes generadores congruenciales.
 - $x_{i+1} = (21x_i + 15) \bmod (31)$ con $x_0 = 21$
 - $x_{i+1} = (13x_i + 9) \bmod (128)$ con $x_0 = 7$
 - $x_{i+1} = (17x_i) \bmod (31)$ con $x_0 = 23$
 - $x_{i+1} = (121 + x_i) \bmod (256)$ con $x_0 = 17$
 - $x_{i+1} = (21x_i + 15x_{i-1}) \bmod (64)$ con $x_0 = 21$ y $x_1 = 43$
- Determine el ciclo o periodo de vida de los siguientes generadores congruenciales.
 - $X_{i+1} = (137 * X_i + 47) \bmod 17$
 - $X_{i+1} = (191 * X_i + 17) \bmod 23$
 - $X_{i+1} = (237 * X_i + 71) \bmod 37$
 - $X_{i+1} = (117 * X_i + 31) \bmod 19$
 - $X_{i+1} = (157 * X_i + 47) \bmod 37$
 - $X_{i+1} = (321 * X_i + 11) \bmod 27$
- Programe en una hoja de cálculo la serie congruencial $x_{i+1} = (553 + 121x_i) \bmod (177)$ con $x_0 = 23$, y haga lo que se indica.
 - Determine el ciclo o periodo de vida.
 - Realice las pruebas de media, varianza y uniformidad.
- Realice las pruebas de uniformidad, series y corridas a los primeros 100 aleatorios de los siguientes generadores
 - $X_{i+1} = (1117 * X_i + 3057) \bmod 1679567$; semilla 1457
 - $X_{i+1} = (2177 * X_i + 2367) \bmod 1351867$; semilla 1117
- Para cada uno de los generadores del problema anterior tome ahora los datos de 101 al 200 y realice las pruebas de media, varianza y póker.
- Programe en una hoja de cálculo la generación automática de números pseudoaleatorios con el método de cuadrados medios. Genere una muestra de 50 números con la semilla 5735, y determine con un nivel de aceptación de 90 % si son uniformes entre 0 y 1.
- Realice las pruebas de media, varianza y uniformidad a los 50 números de la tabla siguiente, con un nivel de aceptación de 95 %.

0.8797	0.3884	0.6289	0.8750	0.5999	0.8589	0.9996	0.2415	0.3808	0.9606
0.9848	0.3469	0.7977	0.5844	0.8147	0.6431	0.7387	0.5613	0.0318	0.7401
0.4557	0.1592	0.8536	0.8846	0.3410	0.1492	0.8681	0.5291	0.3188	0.5992
0.9170	0.2204	0.5991	0.5461	0.5739	0.3254	0.0856	0.2258	0.4603	0.5027
0.8376	0.6235	0.3681	0.2088	0.1525	0.2006	0.4720	0.4272	0.6360	0.0954

8. Genere la secuencia de aleatorios del generador congruencial $x_{i+1} = (71x_i) \bmod (357)$ con $x_0 = 167$ y efectúe lo que se indica:
- Realice la prueba de corridas arriba y abajo.
 - Realice la prueba de corridas arriba y abajo de la media.
9. Obtenga una secuencia de aleatorios de tamaño $n = 200$ con el generador congruencial MINSTD.^[6] $x_{i+1} = (16807x_i) \bmod (2147483647)$ con $x_0 = 1$ y efectúe lo que se indica:
- Realice la prueba de media, varianza y uniformidad.
 - Realice la prueba de corridas arriba y abajo.
 - Realice la prueba de corridas arriba y abajo de la media.
 - Realice la prueba de póker y series.
 - Realice la prueba de huecos.
10. Obtenga una secuencia de aleatorios con el generador congruencial Super- Duper.^[7] $x_{i+1} = (69069x_i) \bmod (4294967296)$ con $x_0 = 1$ y efectúe lo que se indica:
- Realice la prueba de media, varianza y uniformidad.
 - Realice la prueba de corridas arriba y abajo.
 - Realice la prueba de corridas arriba y abajo de la media.
 - Realice la prueba de póker y series.
 - Realice la prueba de huecos.
11. Determine si la siguiente lista de 100 números de 2 dígitos tiene una distribución uniforme con un nivel de aceptación de 90 %.

0.78	0.98	0.24	0.73	0.43	0.16	0.78	0.47	0.18	0.55
0.04	0.29	0.68	0.77	0.16	0.03	0.79	0.22	0.37	0.80
0.96	0.26	0.91	0.55	0.75	0.55	0.64	0.39	0.53	0.45
0.61	0.14	0.38	0.12	0.40	0.74	0.78	0.98	0.27	0.60
0.43	0.67	0.62	0.32	0.53	0.54	0.24	0.29	0.18	0.08
0.82	0.94	0.19	0.98	0.41	1.00	0.74	0.92	0.14	0.43
0.83	0.88	0.18	0.21	0.50	0.13	0.43	0.69	0.08	0.12
0.22	0.50	0.16	0.11	0.18	0.89	0.80	0.42	0.29	0.87
0.83	0.79	0.65	0.28	0.78	0.49	0.36	0.86	0.87	0.64
0.51	0.07	0.18	0.94	0.50	0.22	0.66	0.91	0.48	0.24

12. Utilice la prueba de póker con nivel de aceptación de 95 % para comprobar la hipótesis de que los números de la siguiente lista son aleatorios.

0.5632	0.2395	0.5583	0.8050	0.4166	0.5454	0.5491	0.5593	0.7725	0.2326
0.1020	0.4708	0.5690	0.3802	0.8224	0.6866	0.7098	0.9352	0.1388	0.4535
0.0945	0.1357	0.9191	0.1503	0.1645	0.9770	0.1301	0.1100	0.2523	0.4439
0.9499	0.9415	0.7413	0.9335	0.0805	0.8295	0.4575	0.1863	0.5504	0.8926
0.9035	0.1133	0.1115	0.8761	0.0007	0.6222	0.4605	0.0688	0.9164	0.3482
0.9419	0.3802	0.8765	0.5340	0.6593	0.8266	0.5932	0.4277	0.9162	0.7300
0.0927	0.4691	0.5736	0.5615	0.1909	0.2143	0.2672	0.7864	0.3218	0.4765
0.5581	0.0888	0.3969	0.0151	0.8605	0.9615	0.7752	0.0461	0.1122	0.7559
0.4251	0.7327	0.8791	0.4445	0.8864	0.6384	0.6607	0.2892	0.8905	0.5126
0.7184	0.0512	0.5982	0.3277	0.0407	0.2668	0.5557	0.8139	0.3261	0.7949
0.2236	0.1455	0.5083	0.6106	0.7605	0.9788	0.0204	0.6006	0.1452	0.1234

13. Determine, mediante las pruebas de independencia (corridas arriba y abajo, corridas arriba y debajo de la media, de póker, de series o de huecos) si los 100 números de la tabla son pseudoaleatorios con un nivel de aceptación de 90 %.

0.78	0.98	0.24	0.73	0.43	0.16	0.78	0.47	0.18	0.55
0.04	0.29	0.68	0.77	0.16	0.03	0.79	0.22	0.37	0.80
0.96	0.26	0.91	0.55	0.75	0.55	0.64	0.39	0.53	0.45
0.61	0.14	0.38	0.12	0.40	0.74	0.78	0.98	0.27	0.60
0.43	0.67	0.62	0.32	0.53	0.54	0.24	0.29	0.18	0.08
0.82	0.94	0.19	0.98	0.41	1.00	0.74	0.92	0.14	0.43
0.83	0.88	0.18	0.21	0.50	0.13	0.43	0.69	0.08	0.12
0.22	0.50	0.16	0.11	0.18	0.89	0.80	0.42	0.29	0.87
0.83	0.79	0.65	0.28	0.78	0.49	0.36	0.86	0.87	0.64
0.51	0.07	0.18	0.94	0.50	0.22	0.66	0.91	0.48	0.24

14. Abra el directorio telefónico en la primera página de la letra D y seleccione los últimos 5 dígitos de los primeros 50 números telefónicos. Determine si esta selección es aleatoria con un nivel de aceptación de 95 %; utilice para ello las pruebas de corridas arriba y abajo, arriba y abajo de la media, y póker.
15. Observe y anote los 4 dígitos de las placas de 100 automóviles que pasen por alguna calle. Determine si esta selección es aleatoria con un nivel de aceptación de 95%; utilice para ello las pruebas de corridas arriba y abajo, arriba y abajo de la media, y la prueba de series.

16. Determine con la prueba de corridas arriba y abajo si los 50 números de la tabla son independientes con un nivel de aceptación de 90 %.

0.6069	0.5316	0.5929	0.4131	0.2991	0.6848	0.8291	0.1233	0.2497	0.9481
0.4411	0.8195	0.3521	0.8068	0.1062	0.5384	0.9287	0.7954	0.7271	0.5739
0.4029	0.2549	0.1003	0.5523	0.1897	0.8725	0.4439	0.6056	0.8310	0.4709
0.1926	0.0266	0.5696	0.7504	0.8542	0.6045	0.2269	0.7970	0.3738	0.1284
0.6367	0.9543	0.5385	0.2574	0.2396	0.3468	0.4105	0.5143	0.2014	0.9900

17. Determine, con la prueba de corridas arriba y abajo de la media, si los 50 números de la tabla son independientes con un nivel de aceptación de 90 %.

0.6351	0.0272	0.0227	0.3827	0.0659	0.3683	0.2270	0.7323	0.4088	0.2139
0.4271	0.4855	0.2028	0.1618	0.5336	0.7378	0.3670	0.6637	0.1864	0.6734
0.9498	0.9323	0.0265	0.4696	0.7730	0.9670	0.7500	0.5259	0.5269	0.5406
0.3641	0.0356	0.2181	0.0866	0.6085	0.4468	0.0539	0.9311	0.3128	0.1562
0.8559	0.7280	0.7789	0.1746	0.6637	0.0687	0.5494	0.1504	0.8397	0.2995

18. Utilice la prueba de series para determinar si los 50 números de la tabla son independientes con un nivel de aceptación de 90 %.

0.5858	0.8863	0.8378	0.3203	0.4115	0.2710	0.9238	0.1959	0.9268	0.6702
0.6213	0.4360	0.6279	0.8415	0.5786	0.0543	0.3567	0.1655	0.3880	0.8080
0.1931	0.0843	0.9152	0.6093	0.7587	0.4515	0.3203	0.5139	0.7070	0.9123
0.1242	0.8826	0.9921	0.8523	0.6723	0.8540	0.4722	0.4781	0.2101	0.1680
0.8658	0.4028	0.6136	0.8720	0.1126	0.5857	0.9172	0.8943	0.8095	0.6408

19. Genere en una hoja de cálculo 200 números aleatorios en una misma columna, use la función predeterminada ALEATORIO (o RAND). Copie estos valores y ubíquelos en la siguiente columna, pero desfázelos una posición. Copie el último de los valores en el lugar que quedó vacío al principio, y haga una gráfica de relación XY. ¿Se observa que los datos están dispersos de manera uniforme?
20. Obtenga la media y la varianza de los datos del problema 18. ¿Son exactamente los mismos que para una distribución uniforme entre cero y uno? ¿A qué atribuye esta diferencia?
21. Un método multiplicativo mixto genera 19500 números de 3 dígitos, de los cuales 13821 tienen todos sus dígitos diferentes, 5464 pares y 215 tercias. Calcule el error respecto de las frecuencias esperadas bajo la prueba póker.
22. Un método congruencial genera 71500 números de 4 dígitos, de los cuales 3500 se clasifican como 2 pares. Calcule el error de este evento respecto de su frecuencia esperada bajo la prueba póker.

23. Al realizar la prueba póker a 50 números aleatorios de 4 dígitos, el resultado del error total es de 11.07. ¿Aceptaría la hipótesis de independencia con nivel de aceptación de 95 %?
24. Al realizar la prueba póker a X cantidad de números aleatorios de 6 dígitos, el resultado del error total es de 15.51. ¿Aceptaría la hipótesis de independencia con nivel de aceptación de 95 %?
25. Un método congruencial genera 357500 números de 6 dígitos, de los cuales 17500 se clasifican como 2 pares. Calcule el error de este evento respecto de su frecuencia esperada bajo la prueba póker.
26. ¿Cuáles de las aseveraciones siguientes son correctas?
- La prueba póker requiere números aleatorios de 5 dígitos.
 - Si acepto que los números son uniformes (0,1), no necesito hacer la prueba de media = $\frac{1}{2}$ y de varianza = $\frac{1}{12}$.
 - Si acepto la prueba de series los números no contienen ciclos o tendencias.
 - Si acepto la prueba de media = $\frac{1}{2}$ y la de varianza = $\frac{1}{12}$, entonces los números son uniformes (0,1).
27. La siguiente tabla muestra los resultados de la prueba de series después de clasificar los números entre 0 y 1.

92	85	90	93
88	98	93	90
96	91	86	88
100	85	84	81

- Calcule el error total existente (C) entre lo real y lo teórico.
 - ¿Existe evidencia estadística de falta de independencia de la secuencia de números con un nivel de aceptación de 90 %?
28. Calcule la cantidad mínima y máxima de corridas que deben existir en una secuencia de 17000 números para concluir que son números aleatorios con un nivel de confianza de 95 %.
29. Genere 100 números pseudoaleatorios usando cualquier hoja de cálculo, y realice las pruebas de corridas, uniformidad e independencia. ¿Bajo este análisis es posible considerar que el generador de números aleatorios que tiene la hoja de cálculo usada es confiable?
30. Para los siguientes números pseudoaleatorios determine por medio de una prueba de series si se pueden considerar independientes.
- Utilice 9 casillas equiprobables

- b) Utilice 16 casillas equiprobables
 c) Utilice 25 casillas equiprobables.

0.1108	0.9082	0.4738	0.1181	0.7422	0.5005	0.6438	0.1873	0.7379	0.4999
0.3316	0.2974	0.2480	0.0048	0.9841	0.8873	0.6742	0.4242	0.2729	0.5008
0.1813	0.7348	0.4232	0.5429	0.7373	0.3440	0.4653	0.1465	0.1511	0.7932
0.6113	0.1508	0.4751	0.2777	0.4041	0.2909	0.4072	0.7725	0.7611	0.5241
0.4128	0.7647	0.8461	0.3529	0.9966	0.6576	0.1951	0.1844	0.0851	0.9698
0.1883	0.1328	0.4944	0.5323	0.0333	0.1106	0.3004	0.1128	0.4664	0.4820
0.2977	0.9579	0.6782	0.2963	0.9185	0.8141	0.9240	0.5254	0.8034	0.7441
0.5114	0.3399	0.2121	0.0674	0.8534	0.2144	0.9660	0.2248	0.9855	0.9394
0.0213	0.0164	0.8420	0.8373	0.7677	0.9857	0.6978	0.4136	0.4721	0.2812
0.6239	0.0502	0.1848	0.1978	0.5805	0.1240	0.9546	0.2802	0.6087	0.7146

31. Dado el siguiente generador de aleatorios $X_{i+1} = (2177 * X_i + 2367) \bmod 1351867$; semilla 1117. Tome los primeros números entre cero y uno para completar 100 pares de datos. Luego elabore una prueba de series como se pide.
- a) Considere clases en incrementos de 0.125 en el eje X y de 0.25 para Y. Es decir, una matriz de 32 casillas.
- b) Considere clases en incremento de 0.2 en el eje X y 0.25 para Y. Es decir, una matriz de 20 casillas.
- c) Basado en los dos casos, ¿podemos decir que los aleatorios generados son independientes?
32. La siguiente tabla muestra los resultados de la prueba de huecos con $\beta - \alpha = 0.1$ después de clasificar los números uniformes.

Tamaño del hueco	Frecuencia observada
0	5
1	4
2	3
3	3
> 3	25
Total	40

- a) Calcule el error total existente entre lo real y lo teórico.
- b) ¿Se puede considerar que esta muestra es pseudoaleatoria con un nivel de aceptación de 90 %?

33. Determine, mediante la prueba de huecos, con $\alpha = 0.5$ y $\beta = 0.8$, si los 50 números de la tabla son independientes con un nivel de aceptación de 90 %.

0.5858	0.8863	0.8378	0.3203	0.4115	0.2710	0.9238	0.1959	0.9268	0.6702
0.6213	0.4360	0.6279	0.8415	0.5786	0.0543	0.3567	0.1655	0.3380	0.8080
0.1931	0.0843	0.9152	0.6093	0.7587	0.4515	0.3203	0.5139	0.7070	0.9123
0.1242	0.8826	0.9921	0.8523	0.6723	0.8540	0.4722	0.4781	0.2101	0.1680
0.8658	0.4028	0.6136	0.8720	0.1126	0.5857	0.9172	0.8943	0.8095	0.6408

Referencias

- [1] Banks, J., Carson, J.S., Nelson, B.L., y Nicol, D.M.(2005). *Discrete-Event System Simulation* (4th ed.) NuevaJersey: Prentice Hall.
- [2] Blum, L., Blum, M. y Shub, M. A(1986). Simple Unpredictable Pseudo-random Number Generator, *SIAM J. Comput*, volumen 15 (número 2), pp. 364-383.
- [3] Law, A.M. y Kelton, W.D. (2000). *Simulation Modeling and Analysis* (3th. ed.): McGraw Hill.
- [4] L'Ecuyer, P (1994b). Uniform Random Number Generation, *Ann. Of Operations Research.*, 53, pp. 77-120.
- [5] Lehmer D.H. (1951). *Proceedings of the Second Symposium on Large-Scale Digital Computing Machinery*, Massachussetts: Harvard University Press, Cambridge.
- [6] Lewis P.A., Goodman A.S., and Miller J.M. (1969). A pseudo-random number generator for the System/360, *IBM Syst. J.*, volúmenes 8, 2, pp.136-146.
- [7] Margsalia D.H. (1972). The structure of linear congruential sequences, *Applications of Number Theory to Numerical Analysis*, S.K. Zaremba, pp. 248-285, Nueva York: Academic Press.



Capítulo 3

Variables aleatorias

- 3.1** Definición de variable aleatoria
 - 3.2** Tipos de variables aleatorias
 - 3.3** Determinación del tipo de distribución de un conjunto de datos
 - 3.4** Generación de variables aleatorias
 - 3.5** Expresiones comunes de algunos generadores de variables aleatorias
 - 3.6** Problemas
- Referencias

3.1 Definición de variable aleatoria

A lo largo de los capítulos anteriores hemos mencionado que un modelo de simulación permite lograr un mejor entendimiento de prácticamente cualquier sistema. Para ello, resulta indispensable obtener la mejor aproximación a la realidad, lo cual se consigue componiendo el modelo con base en variables aleatorias que interactúen entre sí. Pero, ¿cómo podemos determinar qué tipo de distribución tiene una variable aleatoria?, ¿cómo podemos usarla en el modelo una vez que conocemos su distribución asociada? En este capítulo comentaremos los métodos y las herramientas que pueden dar respuesta a estas interrogantes clave para la generación del modelo.

Podemos decir que las variables aleatorias son aquellas que tienen un comportamiento probabilístico en la realidad. Por ejemplo, el número de clientes que llegan cada hora a un banco depende del momento del día, del día de la semana y de otros factores: por lo general, la afluencia de clientes será mayor al mediodía que muy temprano por la mañana; la demanda será más alta el viernes que el miércoles; habrá más clientes un día de pago que un día normal, etcétera. Debido a estas características, las variables aleatorias deben cumplir reglas de distribución de probabilidad como éstas:

- La suma de las probabilidades asociadas a todos los valores posibles de la variable aleatoria x es uno.
- La probabilidad de que un posible valor de la variables x se presente siempre es mayor que o igual a cero.
- El valor esperado de la distribución de la variable aleatoria es la media de la misma, la cual a su vez estima la verdadera media de la población.
- Si la distribución de probabilidad asociada a una variable aleatoria está definida por más de un parámetro, dichos parámetros pueden obtenerse mediante un estimador no sesgado. Por ejemplo, la varianza de la población σ^2 puede ser estimada usando la varianza de una muestra que es s^2 . De la misma manera, la desviación estándar de la población σ , puede estimarse mediante la desviación estándar de la muestra s .

3.2 Tipos de variables aleatorias

Podemos diferenciar las variables aleatorias de acuerdo con el tipo de valores aleatorios que representan. Por ejemplo, si habláramos del número de clientes que solicitan cierto servicio en un periodo de tiempo determinado, podríamos encontrar valores tales como 0, 1, 2, ..., n , es decir, un comportamiento como el que presentan las distribuciones de probabilidad discretas. Por otro lado, si habláramos del tiempo que tarda en ser atendida una persona, nuestra investigación tal vez arrojaría resultados como 1.54 minutos, 0.028 horas o 1.37 días, es decir, un comportamiento similar al de las distribuciones de probabilidad continuas. Si consideramos lo anterior podemos diferenciar entre variables aleatorias discretas y las continuas.

Variables aleatorias discretas. Este tipo de variables deben cumplir con estos parámetros:

$$P(x) \geq 0$$

$$\sum_{i=0}^{\infty} p_i = 1$$

$$P(a \leq x \leq b) = \sum_{i=a}^b p_i = p_a + \dots + p_b$$

Algunas distribuciones discretas de probabilidad son la uniforme discreta, la de Bernoulli, la hipergeométrica, la de Poisson y la binomial (vea la figura 3.1). Podemos asociar a estas u otras distribuciones de probabilidad el comportamiento de una variable aleatoria. Por ejemplo, si nuestro propósito al analizar un muestreo de calidad consiste en decidir si la pieza bajo inspección es buena o no, estamos realizando un experimento con 2 posibles resultados: la pieza es buena o la pieza es mala. Este tipo de comportamiento está asociado a una distribución de Bernoulli. Por otro lado, si lo que queremos es modelar el número de usuarios que llamarán a un teléfono de atención a clientes, el tipo de comportamiento puede llegar a parecerse a una distribución de Poisson. Incluso podría ocurrir que el comportamiento de la variable no se pareciera a otras distribuciones de probabilidad conocidas. Si éste fuera el caso, es perfectamente válido usar una distribución empírica que se ajuste a las condiciones reales de probabilidad. Esta distribución puede ser una ecuación o una suma de términos que cumpla con las condiciones necesarias para ser considerada una distribución de probabilidad.

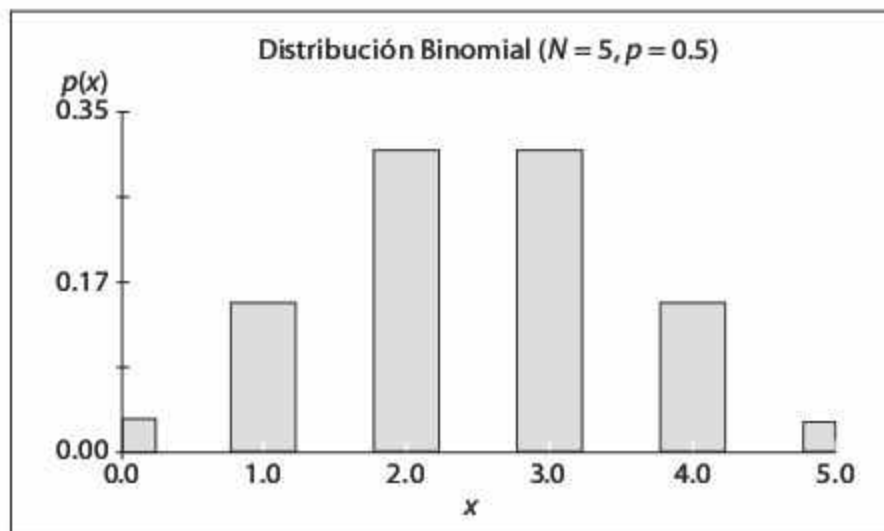


Figura 3.1
Distribución de probabilidad de una variable aleatoria discreta.

Variables aleatorias continuas. Este tipo de variables se representan mediante una ecuación que se conoce como *función de densidad de probabilidad*. Dada esta condición, cambiamos el uso de la sumatoria por la de una integral para conocer la función acumulada de la variable aleatoria. Por lo tanto, las variables aleatorias continuas deben cumplir los siguientes parámetros.

$$P(x) \geq 0$$

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

$$P(a \leq x \leq b) = P(a < x < b) = F(x) = \int_a^b f(x)dx$$

Entre las distribuciones de probabilidad tenemos la uniforme continua, la exponencial, la normal, la de Weibull, la Chi-cuadrada y la de Erlang (vea la figura 3.2). De la misma forma que en las distribuciones discretas, algunos procesos pueden ser asociados a ciertas distribuciones.

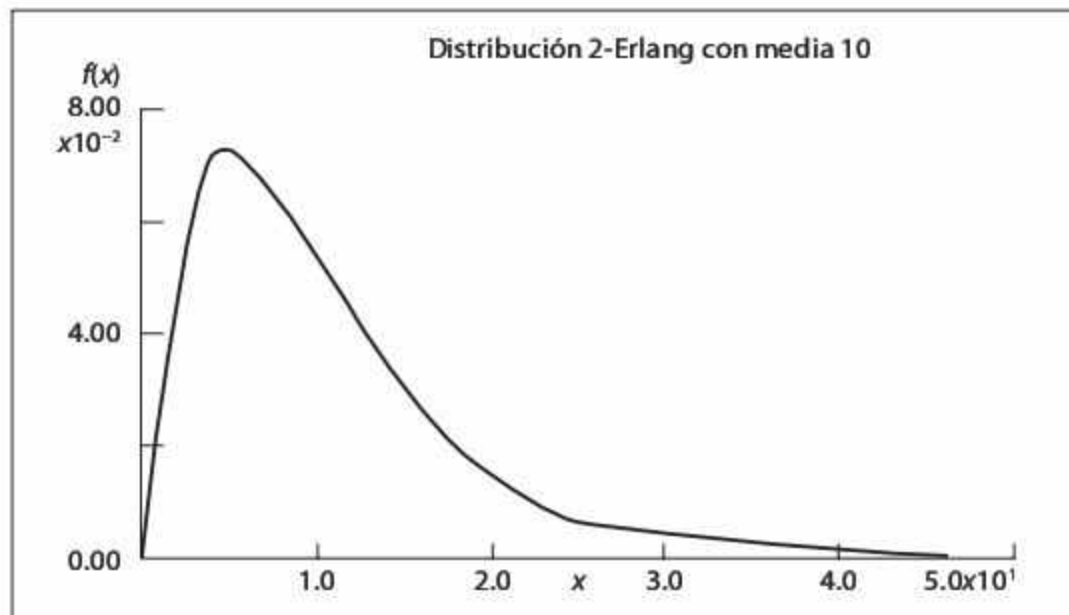


Figura 3.2
Distribución de probabilidad de una variable aleatoria continua.

Por ejemplo, es posible que el tiempo entre llegadas de los clientes a un sistema tenga una distribución de probabilidad muy semejante a una exponencial, o que el tiempo que le toma a un operario realizar una serie de tareas se comporte de manera muy similar a la dispersión que presenta una distribución normal. Sin embargo, debemos hacer notar que este tipo de distribuciones tienen sus desventajas, ya que el rango de valores posibles implica que existe la posibilidad de tener tiempos entre llegada de clientes infinitos o tiempos de ensamble infinitos, situaciones lejanas a la realidad. Por fortuna, es muy poco probable de se presenten este tipo de eventos, aunque el analista de la simulación debe estar consciente de cómo pueden impactar valores como los descritos en los resultados del modelo. En las siguientes secciones revisaremos algunas herramientas útiles para lograr ese objetivo.

3.3 Determinación del tipo de distribución de un conjunto de datos

La distribución de probabilidad de los datos históricos puede determinarse mediante las pruebas Chi-cuadrada, de Kolmogorov-Smirnov y de Anderson-Darling. En esta sección se revisarán los procedimientos de cada una de estas pruebas, así como la forma de realizarlas a través de Stat::Fit, una herramienta complementaria de ProModel.

3.3.1 Prueba Chi-cuadrada

Se trata de una prueba de hipótesis a partir de datos, basada en el cálculo de un valor llamado *estadístico de prueba*, al cual suele compararse con un valor conocido como *valor crítico*, mismo que se obtiene, generalmente, de tablas estadísticas. El procedimiento general de la prueba es:

1. Obtener al menos 30 datos de la variable aleatoria a analizar.
2. Calcular la media y varianza de los datos.
3. Crear un histograma de $m = \sqrt{n}$ intervalos, y obtener la frecuencia observada en cada intervalo O_i .
4. Establecer explícitamente la hipótesis nula, mediante una distribución de probabilidad que se ajuste a la forma del histograma.
5. Calcular la frecuencia esperada, E_i , a partir de la función de probabilidad propuesta.
6. Calcular el estadístico de prueba

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

7. Definir el nivel de significancia de la prueba, α , y determinar el valor crítico de la prueba, $\chi_{\alpha, m-k-1}^2$ (k es el número de parámetros estimados en la distribución propuesta).
8. Comparar el estadístico de prueba con el valor crítico. Si el estadístico de prueba es menor que el valor crítico no se puede rechazar la hipótesis nula.

Ejemplo 3.1

Éstos son los datos del número de automóviles que entran a una gasolinera cada hora:

14	7	13	16	16	13	14	17	15	16
13	15	10	15	16	14	12	17	14	12
13	20	8	17	19	11	12	17	9	18
20	10	18	15	13	16	24	18	16	18
12	14	20	15	10	13	21	23	15	18

Determinar la distribución de probabilidad con un nivel de significancia α de 5 %.

El histograma (vea la figura 3.3) de los $n = 50$ datos, que considera $m = 11$ intervalos, la media muestral de 15.04 y la varianza muestral de 13.14, permite establecer la siguiente hipótesis:

- H_o : Poisson ($\lambda = 15$) automóviles/hora
 H_1 : Otra distribución

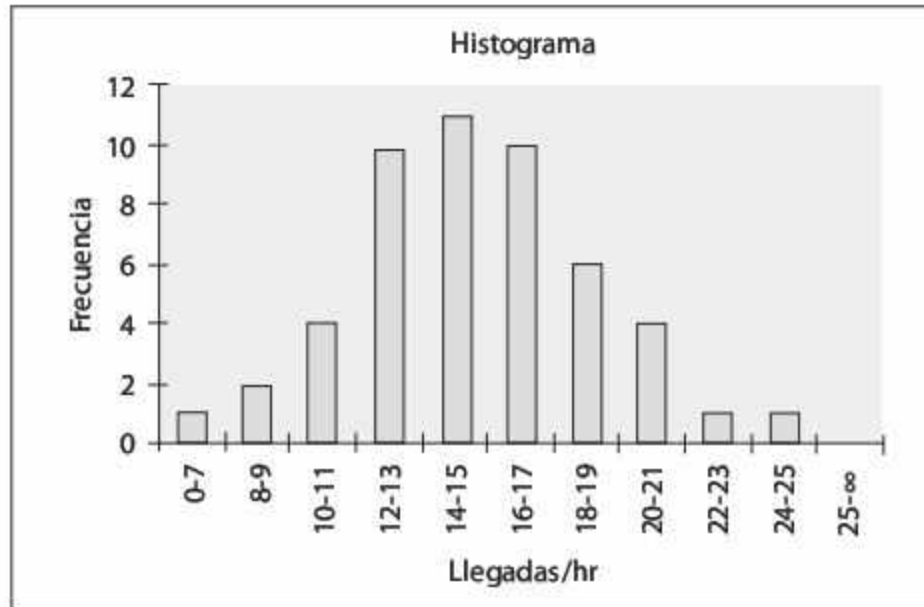


Figura 3.3
Histograma de frecuencias de la llegada de automóviles a la gasolinera.

Comenzamos por calcular la probabilidad de cada intervalo a partir de la función de probabilidad de Poisson:

$$p(x) = \frac{\lambda^x e^{-\lambda}}{x!} \quad x = 0, 1, 2, \dots$$

$$p(x) = \frac{15^x e^{-15}}{x!} \quad x = 0, 1, 2, \dots$$

Por ejemplo, para el intervalo 8-9

$$p(x = 8, 9) = \frac{15^8 e^{-15}}{8!} + \frac{15^9 e^{-15}}{9!} = 0.0519$$

Enseguida calculamos la frecuencia esperada en cada intervalo, multiplicando la probabilidad $p(x)$ por el total de datos de la muestra:

$$E_i = np(x)$$

$$E_i = 50p(x)$$

Y luego estimamos el estadístico de prueba:

$$\chi_o^2 = \sum_{i=1}^m \frac{(E_i - O_i)^2}{E_i} = \frac{(0.9001 - 1)^2}{0.9001} + \frac{(2.5926 - 2)^2}{2.5926} + \dots + \frac{(0.3092 - 0)^2}{0.3092} = 1.7848$$

A partir de los cálculos anteriores se obtiene la tabla 3.1.

Tabla 3.1 Cálculos para la prueba Chi-cuadrada.

Intervalo	O_i	$p(x)$	$E_i = 50 * p(x)$	Error
0-7	1	0.0180	0.9001	0.0111
8-9	2	0.0519	2.5926	0.1354
10-11	4	0.1149	5.7449	0.5300
12-13	10	0.1785	8.9233	0.1299
14-15	11	0.2049	10.2436	0.0559
16-17	10	0.1808	9.0385	0.1023
18-19	6	0.1264	6.3180	0.0160
20-21	4	0.0717	3.5837	0.0483
22-23	1	0.0336	1.6821	0.2766
24-25	1	0.0133	0.6640	0.1700
25-8	0	0.0062	0.3092	0.3092
Total	50	1	50	1.78481

El valor del estadístico de prueba, $\chi_o^2 = 1.7848$, comparado con el valor de tablas crítico, $\chi_{0.05, 11-0-1}^2 = 18.307$, indica que no podemos rechazar la hipótesis de que la variable aleatoria se comporta de acuerdo con una distribución de Poisson, con una media de 15 automóviles/hora.

3.3.2 Prueba de Kolmogorov-Smirnov

Desarrollada en la década de los treinta del siglo XX, esta prueba permite —al igual que la prueba Chi-cuadrada— determinar la distribución de probabilidad de una serie de datos. Una limitante de la prueba de Kolmogorov-Smirnov estriba en que solamente se puede aplicar al análisis de variables continuas. El procedimiento general de la prueba es:

1. Obtener al menos 30 datos de la variable aleatoria a analizar.
2. Calcular la media y la *varianza* de los datos.
3. Crear un histograma de $m = \sqrt{n}$ intervalos, y obtener la frecuencia observada en cada intervalo O_i .
4. Calcular la probabilidad observada en cada intervalo $PO_i = O_i / n$, esto es, dividir la frecuencia observada O_i entre el número total de datos, n .
5. Acumular las probabilidades PO_i para obtener la probabilidad observada hasta el i -ésimo intervalo, POA_i .
6. Establecer de manera explícita la hipótesis nula, para esto se propone una distribución de probabilidad que se ajuste a la forma del histograma.
7. Calcular la probabilidad esperada acumulada para cada intervalo, PEA_i , a partir de la función de probabilidad propuesta.
8. Calcular el estadístico de prueba

$$c = \max |PEA_i - POA_i| \quad i = 1, 2, 3, \dots, k, \dots, m$$

9. Definir el nivel de significancia de la prueba α , y determinar el valor crítico de la prueba, $D_{\alpha,n}$ (consulte la tabla de valores críticos de la prueba de Kolmogorov-Smirnov en la sección de apéndices).
10. Comparar el estadístico de prueba con el valor crítico. Si el estadístico de prueba es menor que el valor crítico no se puede rechazar la hipótesis nula.

Ejemplo 3.2

A continuación se muestra un estudio del comportamiento del tiempo entre roturas de cierto filamento, medido en minutos/rotura.

4.33	1.61	2.16	2.88	0.70	0.44	1.59	2.15	8.59	7.36
9.97	7.86	5.49	0.98	4.52	2.12	4.44	0.82	6.96	3.04
2.81	14.39	3.44	9.92	4.38	8.04	2.18	6.19	4.48	9.66
4.34	1.76	2.30	5.24	11.65	10.92	12.16	6.60	0.85	4.82
1.36	3.53	6.58	1.45	8.42	3.69	2.44	0.28	1.90	2.89

Determinar la distribución de probabilidad con un nivel de significancia α de 5 %.

El histograma (vea la figura 3.4) de los $n = 50$ datos con $m = 8$ intervalos, la media muestral de 4.7336 y la varianza muestral de 12.1991 permiten estimar un parámetro de forma de 1.38 y un parámetro de escala de 5.19, y establecer la hipótesis:

- H_0 : Weibull ($\alpha = 1.38, \beta = 5.19$) minutos/rotura
 H_1 : Otra distribución

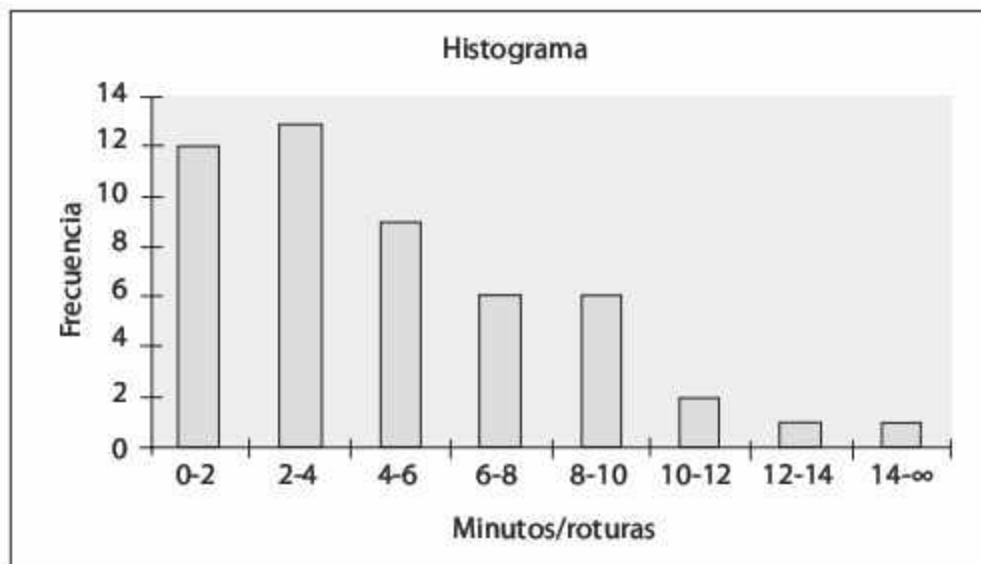


Figura 3.4
Histograma de frecuencias del tiempo entre roturas.

Iniciamos el procedimiento con el cálculo de la probabilidad observada en cada intervalo:

$$PO_i = \frac{O_i}{n} = \frac{O_i}{50} = \left\{ \frac{12}{50}, \frac{13}{50}, \frac{9}{50}, \frac{6}{50}, \frac{6}{50}, \frac{2}{50}, \frac{1}{50}, \frac{1}{50} \right\}$$

para después calcular la probabilidad observada acumulada hasta el intervalo i .

$$POA_i = \frac{\sum O_i}{n} = \frac{\sum O_i}{50} = \left\{ \frac{12}{50}, \frac{25}{50}, \frac{34}{50}, \frac{40}{50}, \frac{46}{50}, \frac{48}{50}, \frac{49}{50}, \frac{50}{50} \right\} = \{0.24, 0.50, \dots, 1\}$$

Posteriormente calculamos la probabilidad que se espera se acumule de cada intervalo PEA_i a partir de la función de probabilidad acumulada de Weibull

$$F(x) = \int_0^x \alpha \beta^{-\alpha} x^{\alpha-1} e^{-\left(\frac{x}{\beta}\right)^\alpha} dx$$

$$F(x) = 1 - e^{-\left(\frac{x}{\beta}\right)^\alpha}$$

$$F(x) = 1 - e^{-\left(\frac{x}{5.19}\right)^{1.38}}$$

Por ejemplo, para el intervalo con el límite superior de 8:

$$PEA_8 = F(x=8) = 1 - e^{-\left(\frac{8}{5.19}\right)^{1.38}} = 0.8375$$

Por último, calculamos el estadístico de prueba

$$c = \max |POA_i - PEA_i| = \max \{ |0.24 - 0.2353|, |0.50 - 0.5025|, \dots, |1 - 1| \} = 0.0375$$

A partir de los cálculos anteriores se obtiene la tabla 3.2:

Tabla 3.2 Cálculos para la prueba de Kolmogorov-Smirnov.

Intervalo	O_i	PO_i	POA_i	PEA_i	$ POA_i - PEA_i $
0-2	12	0.24	0.24	0.23526	0.0047
2-4	13	0.26	0.50	0.50247	0.0025
4-6	9	0.18	0.68	0.70523	0.0252
6-8	6	0.12	0.80	0.83747	0.0375
8-10	6	0.12	0.92	0.91559	0.0044
10-12	2	0.04	0.96	0.95839	0.0016
12-14	1	0.02	0.98	0.98042	0.0004
14-8	1	0.02	1.00	1	0.0000
Total	50	1		c	0.0375

El valor del estadístico de prueba, $c = 0.0375$, comparado con el valor de tablas crítico, $D_{0.05,50} = 0.1923$, indica que no podemos rechazar la hipótesis de que la variable aleatoria se comporta de acuerdo con una distribución de Weibull con parámetro de escala 5.19 y parámetro de forma 1.38.

3.3.3 Prueba de Anderson-Darling

Se dio a conocer en 1954. Esta prueba tiene como propósito corroborar si una muestra de variables aleatorias proviene de una población con una distribución de probabilidad específica. En realidad se trata de una modificación de la prueba de Kolmogorov-Smirnov, aunque tiene la virtud de detectar las discrepancias en los extremos de las distribuciones. La principal desventaja de la prueba de Anderson-Darling estriba en que es necesario calcular los valores críticos para cada distribución. La prueba es muy sensible en los extremos de la distribución, por lo que debe usarse con mucho cuidado en distribuciones con límite inferior acotado, y no es confiable para distribuciones de tipo discreto. En la actualidad es posible encontrar tablas de valores críticos para las distribuciones normal, log-normal, exponencial, log-logística, de Weibull y valor extremo tipo I. El procedimiento general de la prueba es:

1. Obtener n datos de la variable aleatoria a analizar.
2. Calcular la media y la varianza de los datos.
3. Organizar los datos en forma ascendente: $Y_i \quad i = 1, 2, \dots, n$
4. Ordenar los datos en forma descendente. $Y_{n+1-i} \quad i = 1, 2, \dots, n$
5. Establecer de manera explícita la hipótesis nula, al proponer una distribución de probabilidad.
6. Calcular la probabilidad esperada acumulada para cada número Y_i , $PEA(Y_i)$, y la probabilidad esperada acumulada para cada número Y_{n+1-i} , $PEA(Y_{n+1-i})$, a partir de la función de probabilidad propuesta.
7. Calcular el estadístico de prueba

$$A_n^2 = - \left[n + \frac{1}{n} \sum_{i=1}^n (2i - 1) [\ln PEA(Y_i) + \ln(1 - PEA(Y_{n+1-i}))] \right]$$

8. Ajustar el estadístico de prueba de acuerdo con la distribución de probabilidad propuesta.
9. Definir el nivel de significancia de la prueba α , y determinar su valor crítico, $a_{\alpha,n}$ (vea la tabla 3.3).
10. Comparar el estadístico de prueba con el valor crítico. Si el estadístico de prueba es menor que el valor crítico no se puede rechazar la hipótesis nula.

Tabla 3.3 Estadísticos de prueba y valores críticos para la prueba de Anderson-Darling.

Distribución	Estadístico de prueba ajustado	Valores críticos α			
		0.1	.05	.025	.001
Parámetros conocidos $n \geq 5$	A_n^2	1.933	2.492	3.070	3.857
Normal	$A_n^2 \left(1 + \frac{4}{n} - \frac{25}{n^2} \right)$	0.632	0.751	0.870	1.029
Exponencial	$A_n^2 \left(1 + \frac{3}{5n} \right)$	1.070	1.326	1.587	1.943
De Weibull	$A_n^2 \left(1 + \frac{1}{5\sqrt{n}} \right)$	0.637	0.757	0.877	1.038
Log-logística	$A_n^2 \left(1 + \frac{1}{4\sqrt{n}} \right)$	0.563	0.660	0.769	0.906

Ejemplo 3.3

Los siguientes son los datos de un estudio del tiempo de atención a los clientes en una florería, medido en minutos/cliente:

9.400	8.620	9.346	13.323	7.112	13.466	5.764	8.974	9.831	10.056
7.445	6.619	9.260	6.775	8.306	5.633	8.864	13.944	8.952	9.355
10.489	6.306	12.685	11.078	6.957	9.532	9.192	11.731	11.350	14.389
12.553	8.045	9.829	11.804	9.274	12.190	10.270	14.751	9.237	6.515
12.397	8.453	9.628	13.838	9.935	7.827	9.269	8.690	11.515	8.527

Determinar la distribución de probabilidad con un nivel de significancia α de 5 por ciento.

El histograma (vea la figura 3.5) de los $n = 50$ datos, que considera $m = 10$ intervalos, la media muestral de 9.786 y la varianza muestral de 5.414, permite establecer la siguiente hipótesis nula:

H_0 : Normal ($\mu = 10, \sigma = 2.0$) minutos/cliente

H_1 : Otra distribución

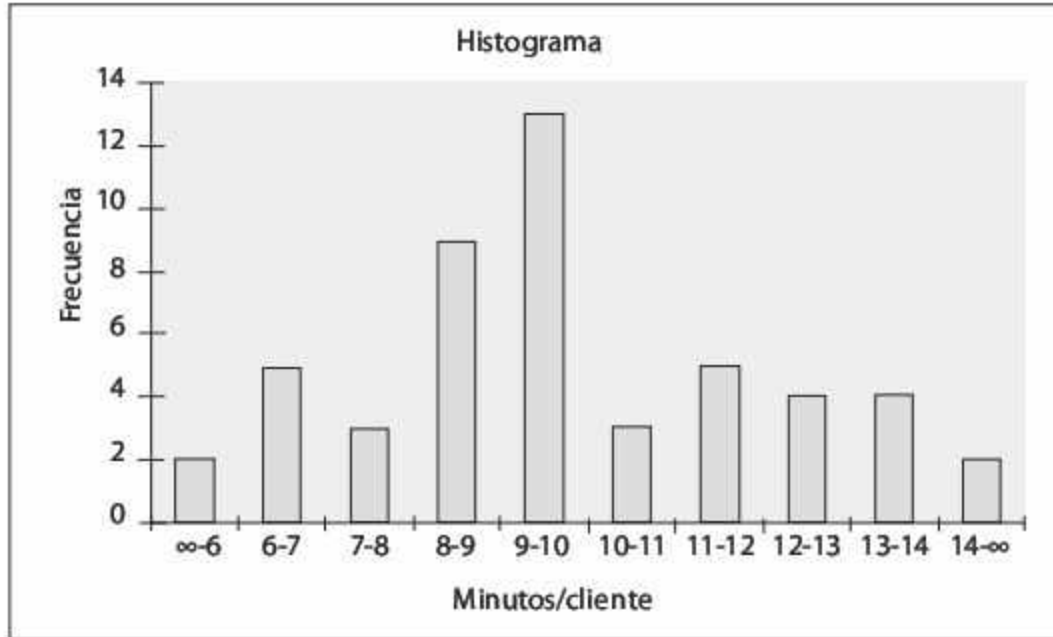


Figura 3.5
Histograma de frecuencias del tiempo de servicio en la florería.

En la tabla 3.4 se muestran los resultados de los cálculos de cada uno de los pasos del procedimiento siguiente:

- En la columna C1 se ordenaron los datos Y_i $i = 1, 2, \dots, 30$ en forma ascendente.
- En la columna C2 se organizaron los datos Y_{30+1-i} $i = 1, 2, \dots, 30$ en forma descendente.
- En la columna C3 se calcula la variable auxiliar $2i - 1$ $i = 1, 2, \dots, 30$ del estadístico de prueba.
- En las columnas C4 y C5 se calcula la probabilidad esperada acumulada para cada número Y_i ; $PEA(Y_i)$ de la columna C1, y el complemento de la probabilidad esperada acumulada para cada número Y_{n+1-i} : $1 - PEA(Y_{n+1-i})$ de la columna C2 a partir de la función de probabilidad propuesta, que en este ejemplo es una normal ($\mu = 10$, $\sigma = 2.0$).

Si se toma, por ejemplo, el renglón $i = 7$ con $Y_7 = 7.8266$ de la columna C1 y $Y_{24} = 10.0562$ de la columna C2, y se estandarizan ambos valores

$$z_{i=7} = \frac{7.8266 - 10}{2} = -1.0867$$

$$z_{i=24} = \frac{10.0562 - 10}{2} = 0.0281$$

se encuentra la probabilidad acumulada en la tabla normal estándar. De esta forma tenemos una $PEA_7 = 0.1385$ para $z = -1.0867$ y, para $z = 0.0285$ una $PEA_{24} = 0.5112$ y $1 - PEA_{24} = 0.4888$.

En las columnas C6, C7 y C8 se desglosan los cálculos del estadístico de prueba

$$A_n^2 = - \left[n + \frac{1}{n} \sum_{i=1}^n (2i-1) [\ln PEA(Y_i) + \ln(1-PEA(Y_{n+1-i}))] \right]$$

$$A_n^2 = - \left[30 + \frac{1}{30} \sum_{i=1}^{30} (2i-1) [\ln PEA(Y_i) + \ln(1-PEA(Y_{30+1-i}))] \right]$$

$$A_n^2 = - \left[30 + \frac{1}{30} [(1)[-4.2399 - 3.1812] + (3)[-4.0692 - 3.1812] + \dots + (59)[-0.0424 - 0.0146]] \right]$$

$$A_n^2 = 2.825724$$

Una vez calculado el estadístico de prueba es necesario ajustarlo de acuerdo con la tabla 3.3. En este caso, como tenemos $n \geq 5$ no se requiere ajuste.

Por último, al comparar el estadístico de prueba $A_n^2 = 2.825724$ con el valor crítico de la prueba y el nivel de significancia seleccionado, $a_{0.05,30} = 2.492$ (vea la tabla 3.3, en donde se dan los valores críticos de α), se rechaza la hipótesis H_0 .

Tabla 3.4 Cálculos de la hipótesis inicial en la prueba de Anderson-Darling.

	C1	C2	C3	C4	C5	C6	C7	C8
<i>i</i>	Y_i	Y_{30+1-i}	$2i-1$	$PEA(Y_i)$	$1-PEA(Y_{30+1-i})$	$LN(C4)$	$LN(C5)$	$(C3)*((C6)+C(7))$
1	5.63282	13.4663	1	0.014496	0.0415	-4.2339	-3.1812	-7.4151
2	5.76414	13.4663	3	0.01709	0.0415	-4.0692	-3.1812	-21.7513
3	5.76414	13.3229	5	0.01709	0.0483	-4.0692	-3.0301	-35.4967
4	6.95686	12.6853	7	0.064058	0.0897	-2.7480	-2.4114	-36.1152
5	7.11163	12.19	9	0.074343	0.1368	-2.5991	-1.9896	-41.2977
6	7.11163	10.2697	11	0.074343	0.4464	-2.5991	-0.8066	-37.4625
7	7.8266	10.0562	13	0.138585	0.4888	-1.9763	-0.7158	-34.9974
8	8.30552	9.93504	15	0.198431	0.5130	-1.6173	-0.6676	-34.2732
9	8.61959	9.83097	17	0.245033	0.5337	-1.4064	-0.6280	-34.5836
10	8.86415	9.82943	19	0.285043	0.5340	-1.2551	-0.6274	-35.7676
11	8.97359	9.62768	21	0.303903	0.5738	-1.1910	-0.5554	-36.6755
12	9.1919	9.53164	23	0.343089	0.5926	-1.0698	-0.5233	-36.6399
13	9.25952	9.39954	25	0.355602	0.6180	-1.0339	-0.4813	-37.8803
14	9.26901	9.34602	27	0.357372	0.6282	-1.0290	-0.4650	-40.3362
15	9.27425	9.34602	29	0.358348	0.6282	-1.0262	-0.4650	-43.2450
16	9.34602	9.27425	31	0.371838	0.6417	-0.9893	-0.4437	-44.4233
17	9.34602	9.26901	33	0.371838	0.6426	-0.9893	-0.4422	-47.2391
18	9.39954	9.25952	35	0.382	0.6444	-0.9623	-0.4394	-49.0620
19	9.53164	9.1919	37	0.407422	0.6569	-0.8979	-0.4202	-48.7701
20	9.62768	8.97359	39	0.426161	0.6961	-0.8529	-0.3623	-47.3930
21	9.82943	8.86415	41	0.466018	0.7150	-0.7635	-0.3355	-45.0617
22	9.83097	8.61959	43	0.466323	0.7550	-0.7629	-0.2811	-44.8902
23	9.93504	8.30552	45	0.487045	0.8016	-0.7194	-0.2212	-42.3263

24	10.0562	7.8266	47	0.511216	0.8614	-0.6710	-0.1492	-38.5467
25	10.2697	7.11163	49	0.553636	0.9257	-0.5912	-0.0773	-32.7565
26	12.19	7.11163	51	0.863247	0.9257	-0.1471	-0.0773	-11.4396
27	12.6853	6.95686	53	0.910306	0.9359	-0.0940	-0.0662	-8.4894
28	13.3229	5.76414	55	0.951689	0.9829	-0.0495	-0.0172	-3.6715
29	13.4663	5.76414	57	0.958464	0.9829	-0.0424	-0.0172	-3.4007
30	13.4663	5.63282	59	0.958464	0.9855	-0.0424	-0.0146	-3.3645

Replanteamiento de una nueva hipótesis:

H_0 : Normal ($\mu = 9.604$, $\sigma = 2.0$) minutos/cliente

H_1 : Otra distribución

En la tabla 3.5 se presenta el resumen de resultados de cada uno de los pasos del procedimiento; la modificación se ve reflejada en los cálculos de las probabilidades esperadas acumuladas. Por ejemplo, para el renglón $i = 7$ tenemos ahora:

$$z_{i=7} = \frac{7.8266 - 9.604}{2} = -0.8887$$

$$z_{i=24} = \frac{10.0562 - 9.604}{2} = 0.2261$$

Con estos valores se encuentra la probabilidad esperada acumulada en la tabla normal estándar: a $z = -0.8887$ le corresponde $PEA_7 = 0.18707$, y a $z = 0.2261$, $PEA_{24} = 0.5894$ y $1 - PEA_{24} = 0.4106$.

Al calcular de nuevo el estadístico de prueba se obtiene:

$$A_n^2 = - \left[n + \frac{1}{n} \sum_{i=1}^n (2i-1) [\ln PEA(Y_i) + \ln(1 - PEA(Y_{n+1-i}))] \right]$$

$$A_n^2 = - \left[30 + \frac{1}{30} \sum_{i=1}^{30} (2i-1) [\ln PEA(Y_i) + \ln(1 - PEA(Y_{n+1-i}))] \right]$$

$$A_n^2 = - \left[30 + \frac{1}{30} [(1)[-3.749 - 3.6218] + (3)[-3.5961 - 3.6218] + \dots + (59)[-0.0271 - 0.0238]] \right]$$

$$A_n^2 = 1.3516$$

Una vez calculado el estadístico de prueba, es necesario ajustarlo de acuerdo con la tabla 3.3. En este caso, como tenemos $n \geq 5$, no se requiere ajuste.

Por último, al comparar el estadístico de prueba $A_n^2 = 1.3516$ con el valor crítico y el nivel de significancia seleccionado, $a_{0.05,30} = 2.492$ (tabla 3.3 de valores críticos de α), vemos que no se puede rechazar la hipótesis H_0 .

Tabla 3.5 Cálculos de la segunda hipótesis en la prueba de Anderson-Darling.

	C1	C2	C3	C4	C5	C6	C7	C8
<i>i</i>	Y_i	Y_{30+1-i}	$2i - 1$	$PEA(Y_i)$	$1 - PEA(Y_{30+1-i})$	$LN(C4)$	$LN(C5)$	$(C3)*((C6) + C(7))$
1	5.63282	13.4663	1	0.02354	0.0267	-3.7491	-3.6218	-7.3709
2	5.76414	13.4663	3	0.02743	0.0267	-3.5961	-3.6218	-21.6535
3	5.76414	13.3229	5	0.02743	0.0315	-3.5961	-3.4583	-35.2718
4	6.95686	12.6853	7	0.09282	0.0617	-2.3771	-2.7854	-36.1375
5	7.11163	12.19	9	0.10634	0.0980	-2.2411	-2.3227	-41.0743
6	7.11163	10.2697	11	0.10634	0.3696	-2.2411	-0.9952	-35.5997
7	7.8266	10.0562	13	0.18707	0.4106	-1.6763	-0.8902	-33.3641
8	8.30552	9.93504	15	0.25808	0.4343	-1.3545	-0.8341	-32.8282
9	8.61959	9.83097	17	0.31128	0.4548	-1.1671	-0.7878	-33.2332
10	8.86415	9.82943	19	0.35571	0.4551	-1.0336	-0.7872	-34.5951
11	8.97359	9.62768	21	0.37629	0.4953	-0.9774	-0.7026	-35.2802
12	9.1919	9.53164	23	0.41837	0.5144	-0.8714	-0.6647	-35.3296
13	9.25952	9.39954	25	0.43161	0.5407	-0.8402	-0.6148	-36.3768
14	9.26901	9.34602	27	0.43348	0.5513	-0.8359	-0.5954	-38.6460
15	9.27425	9.34602	29	0.43451	0.5513	-0.8335	-0.5954	-41.4399
16	9.34602	9.27425	31	0.44867	0.5655	-0.8015	-0.5701	-42.5173
17	9.34602	9.26901	33	0.44867	0.5665	-0.8015	-0.5682	-45.2004
18	9.39954	9.25952	35	0.45927	0.5684	-0.7781	-0.5650	-47.0071
19	9.53164	9.1919	37	0.48556	0.5816	-0.7225	-0.5419	-46.7817
20	9.62768	8.97359	39	0.50471	0.6237	-0.6838	-0.4721	-45.0777
21	9.82943	8.86415	41	0.54486	0.6443	-0.6072	-0.4396	-42.9200
22	9.83097	8.61959	43	0.54516	0.6887	-0.6067	-0.3729	-42.1221
23	9.93504	8.30552	45	0.56572	0.7419	-0.5697	-0.2985	-39.0677
24	10.0562	7.8266	47	0.58943	0.8129	-0.5286	-0.2071	-34.5783
25	10.2697	7.11163	49	0.63037	0.8937	-0.4615	-0.1124	-28.1205
26	12.19	7.11163	51	0.90199	0.8937	-0.1031	-0.1124	-10.9946
27	12.6853	6.95686	53	0.93829	0.9072	-0.0637	-0.0974	-8.5385
28	13.3229	5.76414	55	0.96852	0.9726	-0.0320	-0.0278	-3.2892
29	13.4663	5.76414	57	0.97326	0.9726	-0.0271	-0.0278	-3.1301
30	13.4663	5.63282	59	0.97326	0.9765	-0.0271	-0.0238	-3.0042

3.3.4 Ajuste de datos con Stat::Fit

La herramienta Stat::Fit de ProModel se utiliza para analizar y determinar el tipo de distribución de probabilidad de un conjunto de datos. Esta utilidad permite comparar los resultados entre varias distribuciones analizadas mediante una calificación. Entre sus procedimientos emplea las pruebas Chi-cuadrada, de Kolmogorov-Smirnov y de Anderson-Darling. Además calcula los parámetros apropiados para cada tipo de distribución, e incluye información estadística adicional como media, moda, valor mínimo, valor máximo y varianza, entre otros datos. Stat::Fit se puede ejecutar desde la pantalla de inicio de ProModel, o bien desde el comando **Stat::Fit** del menú **Tools** (vea la figura 3.6).



Figura 3.6
Pantalla de inicio de ProModel.

Una vez que comience a ejecutarse el comando Stat::Fit, haga clic en el icono de la hoja en blanco de la barra de herramientas Estándar para abrir un nuevo documento (también puede abrir el menú **File** y hacer clic en **New**). Enseguida se desplegará una ventana con el nombre Data Table (vea la figura 3.7), en la que deberá introducir los datos de la variable que se analizará, ya sea con el teclado o mediante los comandos Copiar y Pegar (Copy / Paste) para llevar dichos datos desde otra aplicación, como puede ser Excel o el Bloc de notas de Windows.

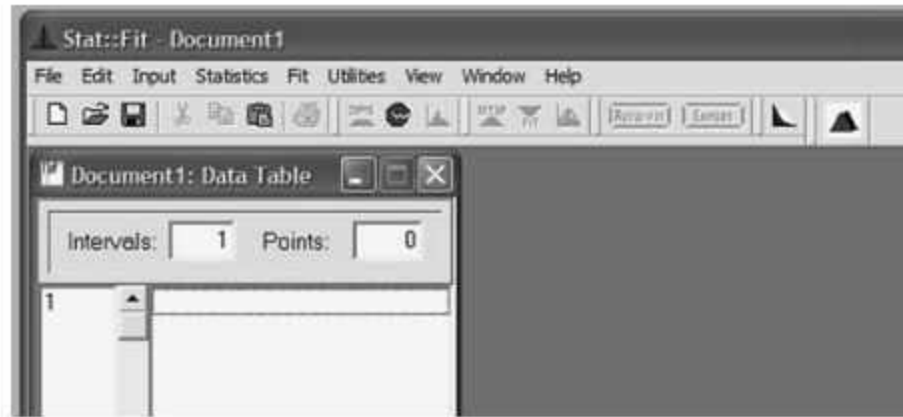


Figura 3.7
Introduzca los datos de la variable que desea analizar en esta ventana de Stat::Fit.

Una vez introducida la información es posible seleccionar una serie de opciones de análisis estadístico, entre ellas las de estadística descriptiva y las de pruebas de bondad de ajuste, de las cuales nos ocuparemos en los siguientes ejemplos.

Ejemplo 3.4

Los datos del número de automóviles que entran a una gasolinera por hora son:

14	7	13	16	16	13	14	17	15	16
13	15	10	15	16	14	12	17	14	12
13	20	8	17	19	11	12	17	9	18
20	10	18	15	13	16	24	18	16	18
12	14	20	15	10	13	21	23	15	18

Determine la distribución de probabilidad con un nivel de significancia α de 5 %.

Después de introducir estos datos en Stat::Fit, despliegue el menú **Statistics** y seleccione el comando **Descriptive**. Enseguida aparecerá una nueva ventana con el nombre de Descriptive Statistics, en donde se muestra el resumen estadístico de la variable (vea la figura 3.8).

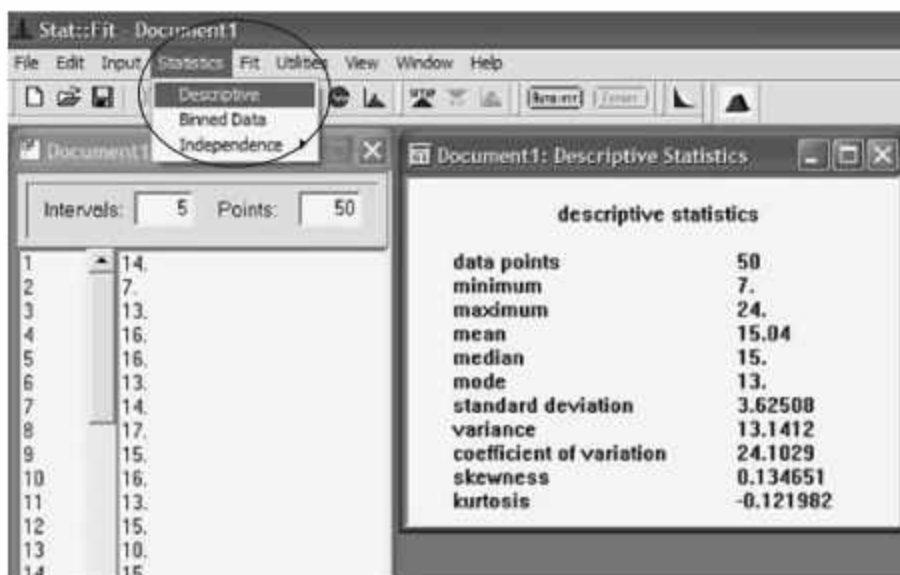


Figura 3.8
Ventana de resultados estadísticos de STAT::Fit.

Para determinar el tipo de distribución de probabilidad de los datos, seleccione el comando **AutoFit** del menú **Fit** en la pantalla principal de Stat::Fit. A continuación se desplegará un cuadro diálogo similar al que se ilustra en la figura 3.9, en el cual se tiene que seleccionar el tipo de distribución que se desea probar. Se debe seleccionar si dicha distribución es no acotada en ambos extremos (unbounded), o si el límite inferior está acotado —en este último caso se puede aceptar la propuesta de que la cota del límite inferior sea el dato más pequeño de la muestra (lower bound)—, o seleccionar explícitamente otro valor como límite inferior (assigned bound). Para este ejemplo seleccionamos una distribución de tipo discreto: **discrete distributions**, ya que los datos de la variable aleatoria [automóviles/hora] tienen esa característica.



Figura 3.9
Este cuadro de diálogo permite seleccionar el tipo de variable aleatoria.

Para que el proceso de ajuste se lleve a cabo, haga clic en el botón **OK**. El resultado se desplegará en la ventana **Automatic Fitting**, donde se describen las distribuciones de probabilidad analizadas, su posición de acuerdo con el ajuste, y si los datos siguen o no alguna de las distribuciones. En la figura 3.10 se observa el resultado del análisis de ajuste del ejemplo 3.4, el cual nos indica que no se puede rechazar la hipótesis de que los datos provengan de cualquiera de las siguientes dos distribuciones; Binomial, con $N = 104$ y $p = 0.145$, o de Poisson, con media 15.0 (esta última coincide con el resultado que obtuvimos en el ejemplo 3.1 mediante la prueba de bondad de ajuste Chi-cuadrada).

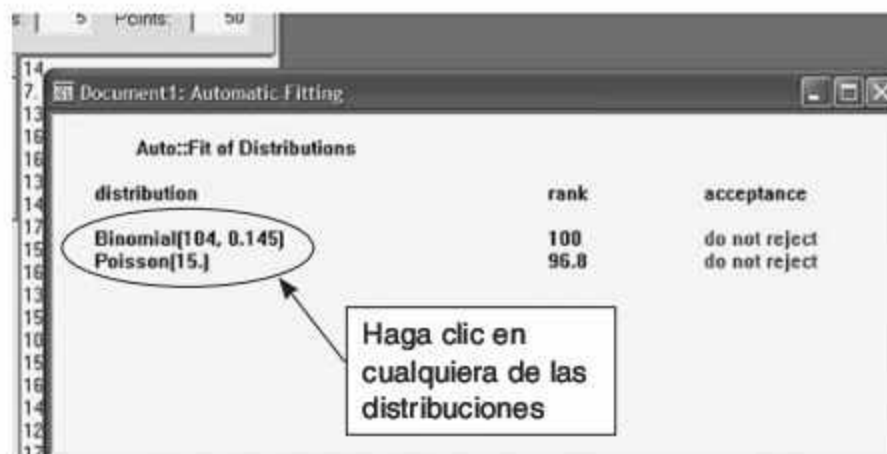


Figura 3.10
Ventana de resultados del análisis de la variable aleatoria.

Haga clic con el ratón en cualquiera de las dos distribuciones (vea la figura 3.10); enseguida se desplegará el histograma que se ilustra en la figura 3.11: las barras azules representan la frecuencia observada de los datos; la línea roja indica la frecuencia esperada de la distribución teórica.

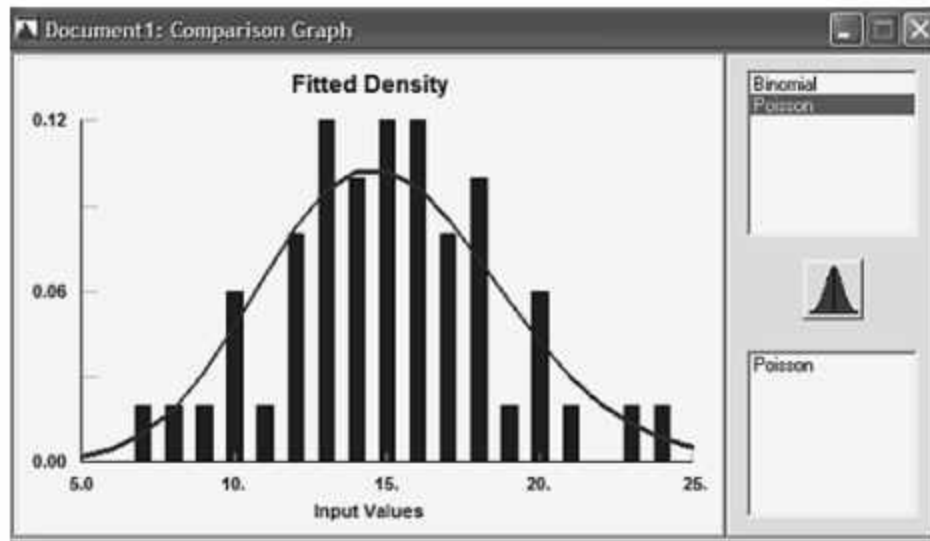


Figura 3.11
Histogramas teórico y real de la variable aleatoria.

El formato del histograma puede ser modificado mediante el comando **Graphics style** del menú **Graphics** (esta opción solamente está disponible cuando se tiene activa la ventana Comparison Graph; vea la figura 3.11).

Ejemplo 3.5

Éstos son los datos de un estudio del tiempo de atención a los clientes en una florería, medido en minutos/cliente:

9.400	8.620	9.346	13.323	7.112	13.466	5.764	8.974	9.831	10.056
7.445	6.619	9.260	6.775	8.306	5.633	8.864	13.944	8.952	9.355
10.489	6.306	12.685	11.078	6.957	9.532	9.192	11.731	11.350	14.389
12.553	8.045	9.829	11.804	9.274	12.190	10.270	14.751	9.237	6.515
12.397	8.453	9.628	13.838	9.935	7.827	9.269	8.690	11.515	8.527

Determine la distribución de probabilidad con un nivel de significancia α de 5 %.

Debido a las características de la variable aleatoria a analizar, al desplegarse el cuadro de diálogo Auto::Fit (vea la figura 3.9) debemos activar la opción **continuous distributions**. El resumen de resultados que se ilustra en la figura 3.12 indica que la muestra puede provenir de cualquiera de las cuatro distribuciones listadas, resultado que coincide con el análisis ejemplificado previamente en la prueba de Anderson-Darling acerca de la normalidad de los datos.

En la ventana **Comparison Graph** puede compararse la forma de la distribución normal (verde) y lognormal (roja) propuestas por Stat::Fit, y la diferencia respecto del histograma de frecuencias de la muestra.

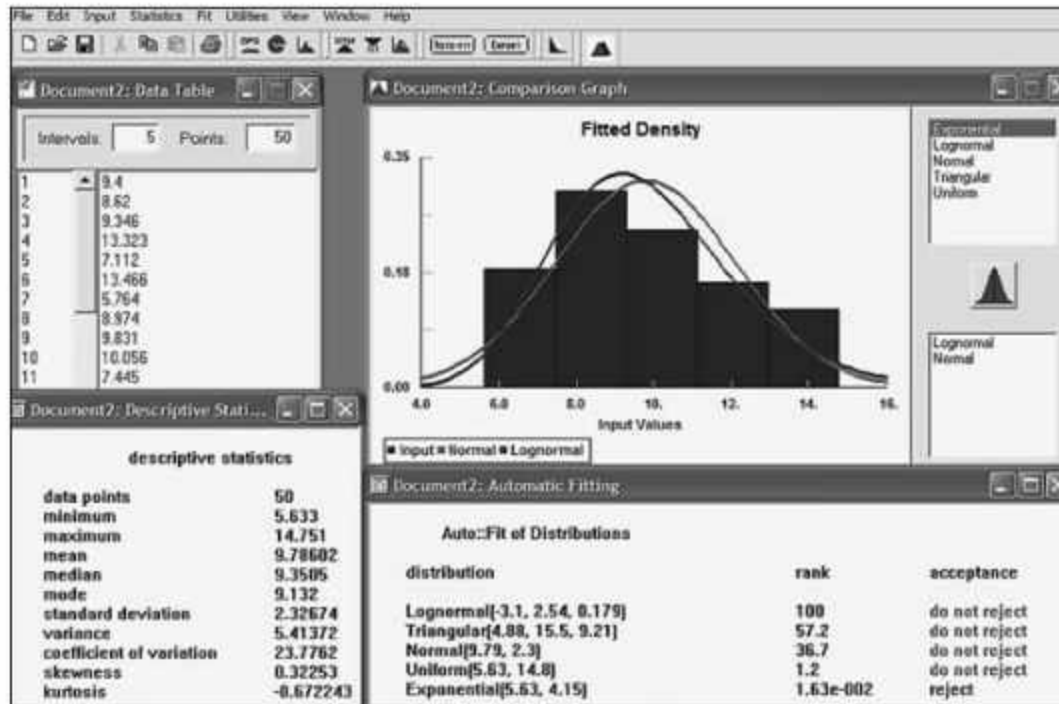


Figura 3.12
Resumen del análisis de la variable aleatoria del ejemplo 3.5.

3.4 Generación de variables aleatorias

La variabilidad de eventos y actividades se representa a través de funciones de densidad para fenómenos continuos, y mediante distribuciones de probabilidad para fenómenos de tipo discreto. La simulación de estos eventos o actividades se realiza con la ayuda de la generación de variables aleatorias.

Los principales métodos para generar las variables aleatorias son:

- Método de la transformada inversa
- Método de convolución
- Método de composición
- Método de transformación directa
- Método de aceptación y rechazo

En las siguientes secciones se describirán los primeros cuatro métodos; el lector interesado en el método de aceptación y rechazo puede consultar la bibliografía recomendada.

3.4.1 Método de la transformada inversa

El método de la transformada inversa puede utilizarse para simular variables aleatorias continuas, lo cual se logra mediante la función acumulada $F(x)$ y la generación de números pseudoaleatorios $r_i \sim U(0, 1)$. El método consiste en:

1. Definir la función de densidad $f(x)$ que represente la variable a modelar.
2. Calcular la función acumulada $F(x)$.
3. Despejar la variable aleatoria x y obtener la función acumulada inversa $F(x)^{-1}$.
4. Generar las variables aleatorias x , sustituyendo valores con números pseudoaleatorios $r_i \sim U(0,1)$ en la función acumulada inversa.

El método de la transformada inversa también puede emplearse para simular variables aleatorias de tipo discreto, como en las distribuciones de Poisson, de Bernoulli, binomial, geométrica, discreta general, etcétera. La generación se lleva a cabo a través de la probabilidad acumulada $P(x)$ y la generación de números pseudoaleatorios $r_i \sim U(0,1)$. El método consiste en:

1. Calcular todos los valores de la distribución de probabilidad $p(x)$ de la variable a modelar.
2. Calcular todos los valores de la distribución acumulada $P(x)$.
3. Generar números pseudoaleatorios $r_i \sim U(0,1)$.
4. Comparar con el valor de $P(x)$ y determinar qué valor de x corresponde a $P(x)$.

En la figura 3.13 se muestra gráficamente la metodología para generar variables aleatorias continuas:

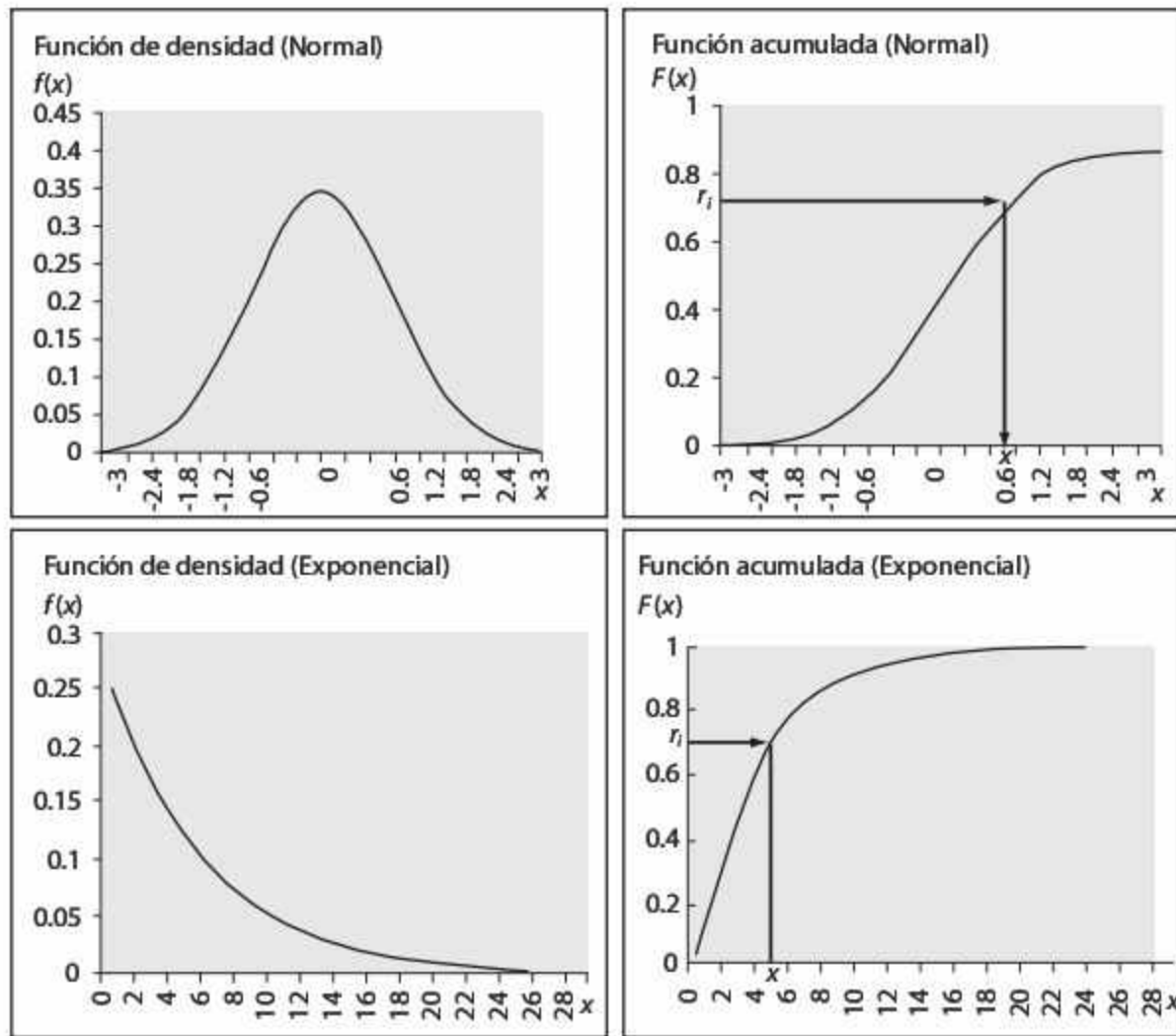


Figura 3.13 Esquemización del método de la transformada inversa para variables continuas.

Por su parte, la figura 3.14 muestra de manera gráfica la metodología para generar variables aleatorias discretas.

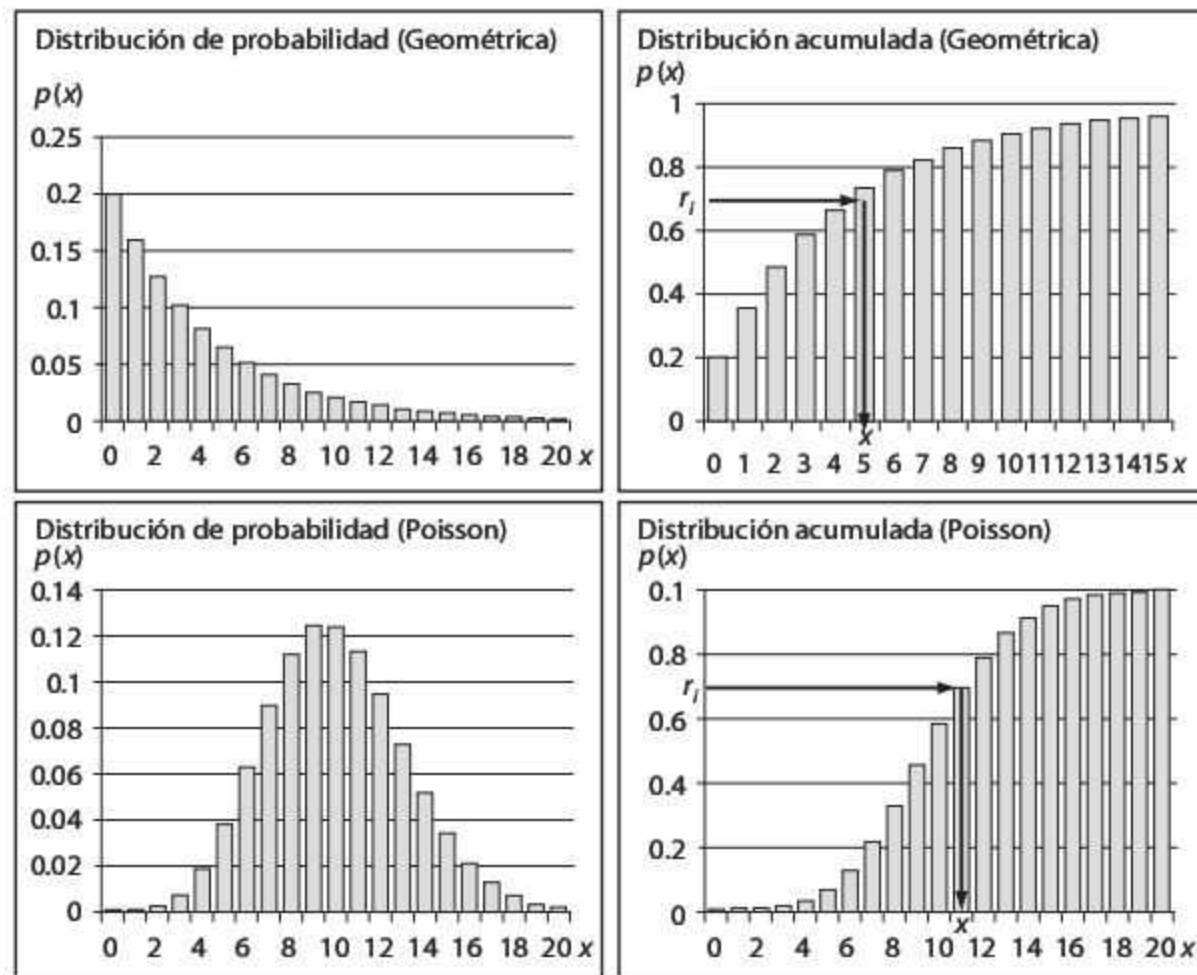


Figura 3.14 Esquemización del método de la transformada inversa para variables discretas.

Distribución uniforme

A partir de la función de densidad de las variables aleatorias uniformes entre a y b ,

$$F(x) = \frac{1}{b-a} \quad a \leq x \leq b$$

se obtiene la función acumulada

$$F(x) = \int_a^x \frac{1}{b-a} dx = \frac{x-a}{b-a} \quad a \leq x \leq b$$

Al igualar la función acumulada $F(x)$ con el número pseudoaleatorio $r_i \sim U(0, 1)$, y despejar x se obtiene:

$$x_i = a + (b - a) F(x)_i$$

$$x_i = a + (b - a) r_i$$

Ejemplo 3.6

La temperatura de una estufa se comporta de manera uniforme dentro del rango de 95 a 100°C. La lista de números pseudoaleatorios y la ecuación $x_i = 95 + 5r_i$ nos permiten modelar el comportamiento de la variable aleatoria que simula la temperatura de la estufa (vea la tabla 3.6).

Tabla 3.6 Simulación de las temperaturas de una estufa.

Medición	r_i	Temperatura °C
1	0.48	97.40
2	0.82	99.10
3	0.69	98.45
4	0.67	98.35
5	0.00	95.00

Distribución exponencial

A partir de la función de densidad de las variables aleatorias exponenciales con media $1/\lambda$,

$$f(x) = \lambda e^{-\lambda x} \quad \text{para} \quad x \geq 0$$

se obtiene la función acumulada

$$F(x) = \int_0^x \lambda e^{-\lambda x} dx = 1 - e^{-\lambda x} \quad \text{para} \quad x \geq 0$$

Si igualamos la función acumulada $F(x)$ con el número pseudoaleatorio $r_i \sim U(0, 1)$, y despejamos x se obtiene:

$$x_i = -\frac{1}{\lambda} \ln(1 - F(x)_i)$$

$$x_i = -\frac{1}{\lambda} \ln(1 - r_i)$$

Ejemplo 3.7

Los datos históricos del tiempo de servicio en la caja de un banco se comportan de forma exponencial con media de 3 minutos/cliente. Una lista de números pseudoaleatorios $r_i \sim U(0,1)$ y la ecuación generadora exponencial $x_i = -3 \ln(1 - r_i)$ nos permiten simular el comportamiento de la variable aleatoria (vea la tabla 3.7).

Tabla 3.7 Simulación del tiempo de servicio en la caja de un banco.

Cliente	r_i	Tiempo de servicio (min)
1	0.64	3.06
2	0.83	5.31
3	0.03	0.09
4	0.50	2.07
5	0.21	0.70

Distribución de Bernoulli

A partir de la distribución de probabilidad de las variables aleatorias de Bernoulli con media

$$p(x) = p^x(1 - p)^{1-x} \quad \text{para } x = 0, 1$$

se calculan las probabilidades para $x = 0$ y $x = 1$, para obtener

x	0	1
$p(x)$	$1 - p$	p

Si acumulamos los valores de $p(x)$ obtenemos:

x	0	1
$p(x)$	$1 - p$	1

Al generar los números pseudoaleatorios $r_i \sim U(0,1)$ se aplica la regla:

$$x_i = \begin{cases} 0 & \text{si } r_i \in (0, 1-p) \\ 1 & \text{si } r_i \in (1-p, 1) \end{cases}$$

Ejemplo 3.8

Los datos históricos sobre la frecuencia de paros de cierta máquina muestran que existe una probabilidad de 0.2 de que ésta falle ($x = 1$), y de 0.8 de que no falle ($x = 0$) en un día determinado. Genere una secuencia aleatoria que simule este comportamiento.

A partir de la distribución de probabilidad de la variable aleatoria de Bernoulli con media 0.8,

$$p(x) = (0.2)^x (0.8)^{1-x} \quad \text{para } x = [0, 1]$$

se calculan las probabilidades puntuales y las acumuladas para $x=0$ y $x=1$, y se obtienen los datos ilustrados en la tabla 3.8:

Tabla 3.8 Cálculo de las probabilidades acumuladas de las fallas de la máquina del ejemplo 3.8.

x	0	1
$P(x)$	0.8	0.2
$P(x)$	0.8	1

La regla para generar esta variable aleatoria estaría definida por:

$$x_i = \begin{cases} 0 & \text{si } r_i \in (0 - 0.8) \\ 1 & \text{si } r_i \in (0.8 - 1) \end{cases}$$

Con una lista de números pseudoaleatorios $r_i \sim U(0,1)$ y la regla anterior es posible simular el comportamiento de las fallas de la máquina a lo largo del tiempo, al considerar lo siguiente:

- Si el número pseudoaleatorio es menor que 0.8, la máquina no fallará.
- Si el número pseudoaleatorio es mayor que 0.8, ocurrirá la falla (vea la tabla 3.9).

Tabla 3.9 Simulación de las fallas de la máquina.

Día	r_i	x_i	Evento: la máquina
1	0.453	0	no falla
2	0.823	1	falla
3	0.034	0	no falla
4	0.503	0	no falla
5	0.891	1	falla

Ejemplo 3.9

El número de piezas que entran a un sistema de producción sigue una distribución de Poisson con media de 2 piezas/hr. Simule el comportamiento de la llegada de las piezas al sistema.

A partir de la distribución de probabilidad de la variable aleatoria de Poisson con media 2,

$$p(x) = \frac{\lambda^x e^{-\lambda}}{x!} \quad \text{para } x = 0, 1, 2, 3, \dots$$

$$p(x) = \frac{2^x e^{-2}}{x!} \quad \text{para } x = 0, 1, 2, 3, \dots$$

se calculan las probabilidades puntuales y las acumuladas para $x = 0, 1, 2, \dots$, y se obtienen los datos de la tabla 3.10.

Tabla 3.10 Cálculo de las probabilidades acumuladas para el ejemplo 3.9.

x	$p(x)$	$P(x)$
0	0.1353	0.1353
1	0.2706	0.4060
2	0.2706	0.6766
3	0.1804	0.8571
4	0.0902	0.9473
5	0.0360	0.9834
6	0.0120	0.9954
7	0.0034	0.9989
8	0.0008	0.9997
9	0.0001	0.9999
10	0.00003	0.9999

La regla para generar esta variable aleatoria estaría definida por:

$$x_i = \begin{cases} 0 & \text{si } r_i \in (0 - 0.1353) \\ 1 & \text{si } r_i \in (0.1353 - 0.4060) \\ 2 & \text{si } r_i \in (0.4060 - 0.6766) \\ 3 & \text{si } r_i \in (0.6766 - 0.8572) \\ 4 & \text{si } r_i \in (0.8571 - 0.9473) \\ 5 & \text{si } r_i \in (0.9473 - 0.9834) \\ 6 & \text{si } r_i \in (0.9834 - 0.9954) \\ 7 & \text{si } r_i \in (0.9954 - 0.9989) \\ 8 & \text{si } r_i \in (0.9989 - 0.9997) \\ 9 & \text{si } r_i \in (0.9997 - 0.9999) \quad \dots \end{cases}$$

Con una lista de números pseudoaleatorios $r_i \sim U(0,1)$ y la regla anterior es posible simular la llegada de las piezas al sistema de producción, con los resultados consignados en la tabla 3.11.

Tabla 3.11 Simulación de la llegada de piezas al sistema, con variables aleatorias de Poisson.

Hora	r_i	Piezas/hr
1	0.6754	2
2	0.0234	0
3	0.7892	3
4	0.5134	2
5	0.3331	1

Ejemplo 3.10

La tabla siguiente muestra la demanda diaria de cepillos dentales en un supermercado. Simule el comportamiento de la demanda mediante el método de la transformada inversa.

Día	1	2	3	4	5	6	7	8	9
Demanda	1	2	2	1	3	0	3	1	3

A partir de la información histórica se calculan las probabilidades puntuales y las acumuladas para $x = 0, 1, 2, 3$ (vea la tabla 3.12).

Tabla 3.12 Cálculo de la probabilidades acumuladas para el ejemplo 3.10.

x	$p(x)$	$p(x)$
0	0.1111	0.1111
1	0.2222	0.3333
2	0.3333	0.6666
3	0.3333	1

La regla para generar esta variable aleatoria estaría determinada por:

$$x_i = \begin{cases} 0 & \text{si } r_i \in (0 - 0.1111) \\ 1 & \text{si } r_i \in (0.1111 - 0.3333) \\ 2 & \text{si } r_i \in (0.3333 - 0.6666) \\ 3 & \text{si } r_i \in (0.6666 - 1) \end{cases}$$

Con una lista de números pseudoaleatorios $r_i \sim U(0,1)$ y la regla anterior es posible simular la demanda diaria de cepillos dentales, tal como se muestra en la tabla 3.13.

Tabla 3.13 Simulación de la demanda de cepillos dentales.

Día	r_i	Demanda diaria
1	0.213	1
2	0.345	2
3	0.021	0
4	0.987	3
5	0.543	2

3.4.2 Método de convolución

En algunas distribuciones de probabilidad la variable aleatoria a simular, Y , puede generarse mediante la suma de otras variables aleatorias X de manera más rápida que a través de otros métodos. Entonces, el método de convolución se puede expresar como:

$$Y = X_1 + X_2 + \dots + X_k$$

Las variables aleatorias de cuatro de las distribuciones más conocidas (de Erlang, normal, binomial y de Poisson) pueden generarse a través de este método, como se verá a continuación.

Distribución de Erlang

La variable aleatoria k -Erlang con media $1/\lambda$ puede producirse a partir de la generación de k variables exponenciales con media $1/k\lambda$:

$$Y = X_1 + X_2 + \dots + X_k$$

$$Y = -\frac{1}{k\lambda} \ln(1-r_1) - \frac{1}{k\lambda} \ln(1-r_2) - \dots - \frac{1}{k\lambda} \ln(1-r_k)$$

$$Y = -\frac{1}{k\lambda} [\ln(1-r_1) + \ln(1-r_2) + \dots + \ln(1-r_k)]$$

$$Y = -\frac{1}{k\lambda} [\ln((1-r_1)(1-r_2)\dots(1-r_k))]$$

$$Y = -\frac{1}{k\lambda} [\ln((1-r_1)(1-r_2)\dots(1-r_k))]$$

$$Y = ER_i = -\frac{1}{k\lambda} \ln \prod_{i=1}^k (1-r_i)$$

Ejemplo 3.11

El tiempo de proceso de cierta pieza sigue una distribución 3-Erlang con media $1/\lambda$ de 8 minutos/pieza. Una lista de números pseudoaleatorios $r_i \sim U(0,1)$ y la ecuación de generación de números Erlang permite obtener la tabla 3.14, que indica el comportamiento de la variable aleatoria.

$$Y = ER_i = -\frac{8}{3} \ln \prod_{i=1}^k (1-r_i)$$

$$Y = -\frac{8}{3} \ln[(1-r_1)(1-r_2)(1-r_3)]$$

Tabla 3.14 Simulación del tiempo de proceso para el ejemplo 3.11.

Pieza	$1-r_1$	$1-r_2$	$1-r_3$	Tiempo de proceso (min/pieza)
1	0.28	0.52	0.64	6.328
2	0.96	0.37	0.83	3.257
3	0.04	0.12	0.03	23.588
4	0.35	0.44	0.50	6.837
5	0.77	0.09	0.21	11.279

Distribución normal

La variable aleatoria normal con media μ y desviación estándar σ puede generarse mediante el *teorema del límite central*:

$$Y = X_1 + X_2 + \dots + X_k \sim N(k\mu_x, k\sigma_x^2)$$

Al sustituir X_i por números pseudoaleatorios r_i , se obtiene:

$$Y = r_1 + r_2 + \dots + r_k \sim N\left(k\frac{1}{2}, k\frac{1}{12}\right)$$

$$Y = r_1 + r_2 + \dots + r_{12} \sim N\left(\frac{12}{2}, \frac{12}{12}\right) \sim N(6, 1)$$

$$Y = Z = (r_1 + r_2 + \dots + r_{12}) - 6 \sim N(0, 1)$$

$$Z = \sum_{i=1}^{12} (r_i) - 6 = \frac{x - \mu}{\sigma} \sim N(0, 1).$$

Si despejamos X , tenemos que

$$x = N_i = \left[\sum_{i=1}^{12} (r_i) - 6 \right] \sigma + \mu$$

Ejemplo 3.12

El volumen de líquido de un refresco sigue una distribución normal con media de 12 onzas y desviación estándar de 0.4 onzas. Genere 5 variables aleatorias con esta distribución para simular el proceso de llenado.

$$N_i = \left[\sum_{i=1}^{12} (r_i) - 6 \right] \sigma + \mu$$

$$N_i = \left[\sum_{i=1}^{12} (r_i) - 6 \right] (0.4) + 12$$

Tabla 3.15 Simulación del volumen de llenado de refrescos (ejemplo 3.12).

Botella	$\sum_{i=1}^{12} (r_i)$	$\sum_{i=1}^{12} (r_i) - 6$	Volumen (onzas)
1	6.21	0.21	12.084
2	5.34	-0.66	11.736
3	6.03	0.03	12.012
4	6.97	0.97	12.038
5	4.81	-1.19	11.524

Distribución Binomial

La variable aleatoria Binomial con parámetros N y p puede ser generada a través de la suma de N variables aleatorias con distribución de Bernoulli con parámetro p .

$$Y = B_i = BE_1 + BE_2 + \dots + BE_N \sim BI(N, p)$$

Ejemplo 3.13

Al inspeccionar lotes de tamaño $N = 5$, la probabilidad de que una pieza sea defectuosa es 0.03. Simule el proceso de inspección para determinar el número de piezas defectuosas por lote.

Este proceso sigue una distribución Binomial con $N = 5$ y $p = 0.03$, y será simulado mediante la generación de variables aleatorias de Bernoulli con $p = 0.03$, de acuerdo con el procedimiento señalado en la sección anterior, donde $BE_i = 0$ representa una pieza en buen estado y $BE_i = 1$, una pieza defectuosa. (Observe los resultados en la tabla 3.16.)

$$BE_i = \begin{cases} 0 & \text{si } r_i \in (0 - 0.97) \\ 1 & \text{si } r_i \in (0.97 - 1) \end{cases}$$

$$B_i = BE_1 + BE_2 + \dots + BE_5$$

Tabla 3.16 Simulación del sistema de inspección del ejemplo 3.13.

Lote	r_1	BE_1	r_2	BE_2	r_3	BE_3	r_4	BE_4	r_5	BE_5	Piezas defectuosas
1	0.49	0	0.32	0	0.15	0	0.01	0	0.45	0	0
2	0.11	0	0.85	0	0.93	0	0.99	1	0.61	0	1
3	0.57	0	0.92	0	0.84	0	0.74	0	0.82	0	0
4	0.62	0	0.01	0	0.68	0	0.98	1	0.99	1	2
5	0.34	0	0.98	1	0.99	1	0.02	0	0.98	1	3

3.4.3 Método de composición

El método de composición —conocido también como método mixto— permite generar variables aleatorias x cuando éstas provienen de una función de densidad $f(x)$ que puede expresarse como la combinación convexa de m distribuciones de probabilidad $f_i(x)$. Entonces, la combinación convexa se puede expresar como:

$$f(x) = \sum_{i=1}^m f_i(x)I_A(x)$$

donde:

$$I_A(x) = \begin{cases} 1 & \text{si } x \in A \\ 0 & \text{si } x \notin A \end{cases}$$

Algunas de las distribuciones más conocidas que pueden expresarse como una combinación convexa son: la triangular, la de Laplace y la trapezoidal. El procedimiento general de generación es el siguiente:

1. Calcule la probabilidad de cada una de las distribuciones $f_i(x)$.
2. Asegúrese que cada función $f_i(x)$ sea función de densidad.
3. Obtenga, mediante el método de la transformada inversa, las expresiones para generar variables aleatorias de cada una de las distribuciones $f_i(x)$.
4. Genere un número pseudoaleatorio r_i que permita definir el valor de $I_A(x)$.
5. Seleccione la función generadora correspondiente a la función $f_i(x)$.
6. Genere un segundo número pseudoaleatorio r_j y sustitúyalo en la función generadora anterior para obtener Y .

Distribución triangular

A partir de la función de densidad triangular

$$f(x) = \begin{cases} \frac{2(x-a)}{(b-a)(c-a)} & a \leq x \leq c \\ \frac{2(b-x)}{(b-a)(b-c)} & c < x \leq b \end{cases}$$

calcule la probabilidad de cada uno de los segmentos de la función

$$p(x) = \begin{cases} \int_a^c \frac{2(x-a)}{(b-a)(c-a)} dx \\ \int_c^b \frac{2(b-x)}{(b-a)(b-c)} dx \end{cases}$$

$$p(x) = \begin{cases} \frac{(c-a)}{(b-a)} & a \leq x \leq c \\ \frac{(b-c)}{(b-a)} & c < x \leq b \end{cases}$$

Ya que los segmentos por separado no son funciones de densidad, se ajustan al dividir entre su correspondiente $p(x)$.

$$f(x) = \begin{cases} \frac{2(x-a)}{(b-a)(c-a)} \frac{(b-a)}{(c-a)} = \frac{2(x-a)}{(c-a)^2} & a \leq x \leq c \\ \frac{2(b-x)}{(b-a)(b-c)} \frac{(b-a)}{(b-c)} = \frac{2(b-x)}{(b-c)^2} & c < x \leq b \end{cases}$$

Al expresar la función como una combinación convexa se obtiene:

$$f(x) = \sum_{i=1}^m f_i(x) I_A(x) = \sum_{i=1}^2 f_i(x) I_A(x)$$

$$f(x) = \frac{2(x-a)}{(c-a)^2} I_{a \leq x \leq c}(x) + \frac{2(b-x)}{(b-c)^2} I_{c < x \leq b}(x)$$

donde

$$I_A(x) = \begin{cases} 1 & \text{si } x \in A \\ 0 & \text{si } x \notin A \end{cases}$$

Para aplicar el método de la transformada inversa, primero integramos a cada segmento de la función:

$$F(x) = \begin{cases} \int_a^x \frac{2(x-a)}{(c-a)^2} dx = \frac{(x-a)^2}{(c-a)^2} & a \leq x \leq c \\ \int_c^x \frac{2(b-x)}{(b-c)^2} dx = 1 - \frac{(b-x)^2}{(b-c)^2} & c < x \leq b \end{cases}$$

Luego, al despejar x y sustituir r_i en $F(x)$ obtenemos:

$$x = \begin{cases} a + (c - a)\sqrt{r_i} \\ b - [(b - c)\sqrt{1 - r_i}] \end{cases}$$

Por último, al expresar la ecuación anterior incluyendo la función indicadora $I_A(x)$ tenemos que:

$$x = \begin{cases} a + (c - a)\sqrt{r_i} & \text{si } r_i \leq \frac{(c - a)}{(b - a)} \\ b - [(b - c)\sqrt{1 - r_i}] & \text{si } r_i > \frac{(c - a)}{(b - a)} \end{cases}$$

Ejemplo 3.14

Genere una muestra de 5 variables aleatorias con distribución triangular a partir de los parámetros: valor mínimo 5, moda 10 y valor máximo 20.

$$x = \begin{cases} a + (c - a)\sqrt{r_i} & \text{si } r_i \leq \frac{(c - a)}{(b - a)} \\ b - [(b - c)\sqrt{1 - r_i}] & \text{si } r_i > \frac{(c - a)}{(b - a)} \end{cases}$$

Al sustituir $a = 5$, $c = 10$ y $b = 20$ obtenemos

$$x = \begin{cases} 5 + (5)\sqrt{r_i} & \text{si } r_i \leq \frac{5}{15} \\ 20 - [(10)\sqrt{1 - r_i}] & \text{si } r_i > \frac{5}{15} \end{cases}$$

Al generar una secuencia de números pseudoaleatorios se obtiene la secuencia de variables triangulares que se lista en la tabla 3.17:

Tabla 3.17 Simulación de variables aleatorias triangulares.

Variable	r_j	r_i	$x = 5 + 5\sqrt{r_i}$ si $r_i \leq .33$	$x = 20 - 10\sqrt{1 - r_i}$ si $r_i > 0.33$
1	0.231	0.456	8.37	-
2	0.421	0.967	-	18.18
3	0.853	0.982	-	18.65
4	0.048	0.134	6.83	-
5	0.675	0.536	-	13.18

3.4.4 Método de transformación directa

Este método se basa en el teorema de Pitágoras, y se usa para generar variables aleatorias normales. En la figura 3.15 se muestra la relación entre las variables involucradas en él.

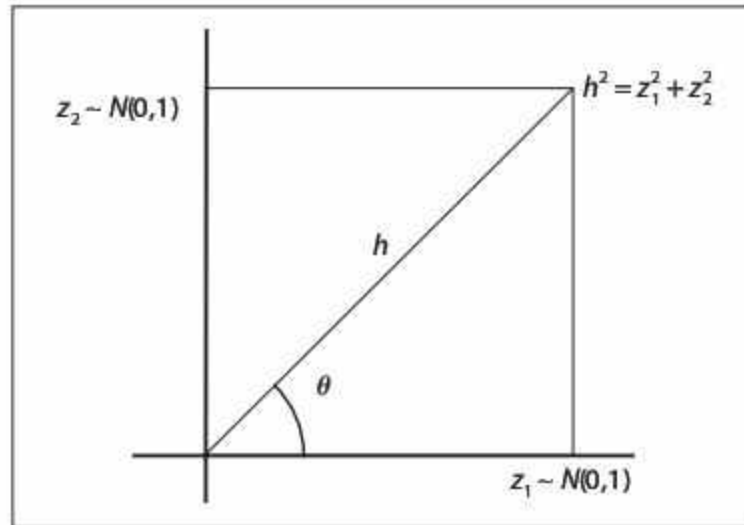


Figura 3.15
Generación de variables aleatorias $z \sim N(0, 1)$.

Geoméricamente,

$$z_2 = h \operatorname{sen} \theta$$

$$z_2 = \sqrt{z_1^2 + z_2^2} \operatorname{sen} \theta$$

La suma de v variables aleatorias normales estándar sigue una distribución Chi-cuadrada con v grados de libertad:

$$z_2 = \sqrt{\chi_{v=2}^2} \operatorname{sen} \theta$$

La función de densidad de una variable aleatoria Chi-cuadrada con 2 grados de libertad es la misma de una distribución exponencial con media igual a 2. En consecuencia, a través de la ecuación obtenida por el método de la transformada inversa para generar variables aleatorias exponenciales, y sustituyéndola en la ecuación anterior se obtiene:

$$z_2 = \sqrt{-2 \ln(1 - r_j)} \operatorname{sen} \theta$$

Se generan valores aleatorios uniformes del ángulo θ entre 0 y 2π mediante el método de la transformada inversa:

$$\theta = a + (b - a)r_j$$

$$\theta = (2\pi)r_j$$

Y al sustituir obtenemos

$$z_2 = \sqrt{-2\ln(1-r_1)}\text{sen}(2\pi r_1)$$

Para cualquier variable aleatoria normal N ,

$$z = \frac{N - \mu}{\sigma}$$

Al despejar N y sustituir el valor de z previamente desarrollado, se llega a la expresión final para la generación de variables aleatorias normales:

$$N_i = \left[\left(\sqrt{-2\ln(1-r_1)} \right) \text{sen}(2\pi r_1) \right] \sigma + \mu$$

Este procedimiento debe iniciarse también a través de la generación de la variable aleatoria z_1 , lo cual dará lugar a la ecuación final

$$N_i = \left[\left(\sqrt{-2\ln(1-r_1)} \right) \text{cos}(2\pi r_1) \right] \sigma + \mu$$

Cualquiera de las dos últimas ecuaciones puede utilizarse para resolver el ejemplo 3.12, en donde el generador de variables aleatorias que simulan el volumen de las botellas sería:

$$N_i = \left[\left(\sqrt{-2\ln(1-r_1)} \right) \text{sen}(2\pi r_1) \right] (0.4) + 12$$

De manera que si producimos los números pseudoaleatorios uniformes 0.43 y 0.75, el volumen del líquido generado para alguna de las botellas sería

$$N_i = \left[\left(\sqrt{-2\ln(1-0.43)} \right) \text{sen}(2\pi(0.75)) \right] (0.4) + 12 = 11.575 \text{ onzas}$$

3.5 Expresiones comunes de algunos generadores de variables aleatorias

En la tabla 3.18 se presentan los generadores de variables aleatorias de las distribuciones de probabilidad más usuales.

Tabla 3.18 Generadores de variables aleatorias.

Distribución	Generador	Parámetros
Uniforme U_i	$U_i = a + (b - a)r_i$	a = Límite inferior de la distribución uniforme. b = Límite superior de la distribución uniforme. r_i = Número aleatorio con distribución uniforme entre 0 y 1.
Triangular T_i	$T_i = \begin{cases} a + \sqrt{(b-a)(c-a)r_i} & \text{si } r_i \leq \frac{(c-a)}{(b-a)} \\ b - \sqrt{(b-a)(b-c)(1-r_i)} & \text{si } r_i > \frac{(c-a)}{(b-a)} \end{cases}$	a = Límite inferior de la distribución triangular. c = Moda de la distribución triangular. b = Límite superior de la distribución triangular.
Triangular T_i	$T_i = \begin{cases} a + (c-a)\sqrt{r_i} & \text{si } r_i \leq \frac{(c-a)}{(b-a)} \\ b - [(b-c)\sqrt{1-r_i}] & \text{si } r_i > \frac{(c-a)}{(b-a)} \end{cases}$	a = Límite inferior de la distribución triangular. c = Moda de la distribución triangular. b = Límite superior de la distribución triangular.
χ^2	$\chi_i^2 = \sum_{j=1}^n Z_j^2$	Z_j = Números aleatorios con distribución normal estándar. n = Grados de libertad.
Erlang ER_i	$ER_i = -\frac{1}{k\lambda} \ln \prod_{i=1}^k (1-r_i)$	$1/\lambda$ = Valor esperado. k = Parámetro de forma.
Normal N_i	$N_i = \left[\left(\sqrt{-2\ln(1-r_i)} \right) \cos(2\pi r_i) \right] \sigma + \mu$ $N_i = \left[\left(\sqrt{-2\ln(1-r_i)} \right) \text{sen}(2\pi r_i) \right] \sigma + \mu$	μ = Media de la distribución normal σ = Desviación estándar de la distribución normal.
Normal N_i	$N_i = \left(\sum_{i=1}^{12} r_i - 6 \right) \sigma + \mu$	μ = Media de la distribución normal. σ = Desviación estándar de la distribución normal.

Exponencial E_i	$E_i = -\frac{1}{\lambda} \ln(1-r_i)$	$1/\lambda =$ Media de la distribución exponencial.
Weibull W_i	$W_i = \gamma + \beta \sqrt{-\ln(1-r_i)}$	$\beta =$ Parámetro de escala. $\alpha =$ Parámetro de forma. $\gamma =$ Parámetro de localización.
Gamma G_i	$G_i = -\frac{1}{k\lambda} \ln \prod_{i=1}^k (1-r_i)$	$1/\lambda =$ Valor esperado. $k =$ Parámetro de forma.
Lognormal LN_i	$LN_i = e^{N_i}$ donde: $N_i = \left(\sum_{i=1}^{12} r_i - 6 \right) \cdot \left(\ln \left(1 + \frac{\sigma^2}{\mu^2} \right) \right)^{1/2} + \left(\ln \frac{\mu^2}{\sqrt{\mu^2 + \sigma^2}} \right)$	$\mu =$ Valor esperado. $\sigma^2 =$ Varianza.
Bernoulli BE_i	$BE_i = \begin{cases} 0 & \text{si } r_i \in (0, 1-p) \\ 1 & \text{si } r_i \in (1-p, 1) \end{cases}$	$p =$ Probabilidad de ocurrencia del evento $x = 1$. $1 - p =$ Probabilidad de ocurrencia del evento $x = 0$.
Binomial B_i	$B_i = \sum_{j=1}^N BE_j$	$BE_j =$ Números aleatorios con distribución de Bernoulli. $N =$ Número del evento máximo de la distribución binomial. $p =$ Probabilidad de éxito de la distribución binomial que se involucra al generar los Bernoulli.
Poisson P_i	<i>Inicialización.</i> Hacer $N = 0, T = 1$ y generar un aleatorio r_i <i>Paso 1.</i> Calcular $T' = (T)(r_i)$ <i>Paso 2.</i> Si la $T' \geq e^{-\lambda}$, entonces hacer $N = N + 1, T = T'$, calcular otro r_i y regresar al paso 1. Si $T' < e^{-\lambda}$, entonces la variable generada esta dada por: $P_i = N$. Para generar la siguiente variable de Poisson, regresar a la fase de inicialización.	$\lambda =$ Media de la distribución de Poisson. $N =$ Contador. $T =$ Contador.

3.6 Problemas

1. Utilice la prueba Chi-cuadrada para determinar, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos.

17.392	8.110	4.078	3.151	3.528	2.440	5.924	3.461	2.052	10.369
3.690	10.870	4.793	2.498	0.569	8.281	0.154	5.959	3.384	12.877
13.602	5.244	16.677	5.977	4.313	4.767	2.381	6.443	1.392	1.578
8.115	4.891	6.720	7.728	2.717	10.451	5.901	0.818	7.088	2.637
4.714	3.032	1.495	15.733	7.768	2.333	7.822	3.708	6.412	1.290
3.957	5.285	7.094	3.078	1.264	2.630	10.177	2.155	2.945	7.552
11.094	4.772	7.281	14.344	19.867	0.119	2.072	1.486	3.791	4.214
1.611	1.781	1.530	3.280	4.301	0.202	7.489	1.422	1.453	0.022
6.001	9.269	8.477	3.043	0.877	6.966	2.103	1.816	0.433	2.547
0.843	1.182	8.121	2.007	1.395	4.661	7.378	5.300	17.066	12.171

2. A partir de la prueba Chi-cuadrada determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos.

18.799	14.889	20.977	25.106	24.793	26.933	11.266	19.063	24.380	15.653
17.239	13.238	12.612	16.089	16.906	11.528	17.728	18.384	20.539	18.538
18.692	18.519	25.371	19.659	19.255	17.947	27.889	23.463	29.503	17.380
26.646	13.550	22.156	23.609	27.676	19.662	17.905	22.701	18.475	23.030
14.223	16.611	13.914	18.548	19.870	20.112	18.709	28.778	13.030	17.054
9.690	25.791	14.881	17.386	23.031	21.867	23.498	22.383	14.513	15.537
22.776	21.291	16.241	19.036	20.526	22.231	20.555	16.356	27.539	21.949
20.289	23.319	23.448	17.454	16.307	24.445	15.195	13.764	22.845	22.554
28.823	25.775	25.216	20.452	20.008	21.815	19.898	15.781	12.901	3.313
21.777	22.472	20.854	15.892	24.953	18.755	16.640	16.715	18.284	18.187

3. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos.

12.656	11.664	11.855	11.399	11.845	9.766	11.866	10.671	12.157	12.503
13.317	11.381	11.252	12.146	11.769	11.792	13.577	12.038	11.854	13.830
11.369	13.271	11.985	11.936	13.610	12.363	12.437	11.765	12.683	11.931
11.264	10.902	12.204	11.019	13.940	11.873	10.412	11.665	12.957	11.617
11.346	10.634	12.316	11.836	12.571	11.363	11.654	12.286	11.669	12.212
9.526	11.931	12.247	14.116	10.475	10.441	9.695	13.172	14.374	11.610
10.999	12.548	12.659	11.148	12.809	12.660	11.793	10.452	13.013	12.763
11.650	11.309	12.863	12.347	12.556	14.086	12.273	10.893	12.480	10.771
12.566	11.843	12.299	12.357	12.131	11.728	10.653	14.121	13.598	13.049
10.522	10.883	12.533	12.074	11.991	12.161	10.118	11.743	11.062	11.002

4. Emplee la prueba Chi-cuadrada para determinar, con un nivel de confianza de 95%, qué tipo de distribución siguen los datos. Compruebe con la herramienta Stat::Fit de ProModel.

1.679	1.187	0.234	1.780	1.458	2.628	0.504	0.951	1.383	0.486
0.561	0.494	4.923	0.635	0.504	2.606	0.382	1.380	2.700	0.468
2.771	3.141	1.019	2.516	1.182	2.258	0.161	8.055	0.464	2.312
2.327	0.761	1.876	1.506	2.451	0.831	5.715	0.699	1.450	3.582
0.684	3.192	1.427	0.518	2.198	0.922	1.597	2.660	2.933	4.518

5. Determine, con un nivel de confianza de 95%, qué tipo de distribución siguen los datos; emplee la prueba Chi-Cuadrada.

12.561	2.695	12.082	10.335	13.260	2.549	4.594	2.500	24.930	7.805
8.322	7.422	11.143	20.599	7.508	4.367	1.544	3.706	8.185	14.405
4.057	15.584	9.049	6.265	10.663	10.257	11.475	4.688	16.256	4.688
11.963	5.599	19.204	1.784	25.998	12.299	10.317	3.779	18.993	7.419
15.154	9.579	8.423	6.934	2.005	13.234	5.542	5271	12.831	8.231
15.330	7.958	7.103	16.134	0.189	10.165	14.624	15.696	10.212	0.891
3.186	9.051	11.118	4.449	17.901	15.497	6.645	5.078	11.555	3.724
21.500	7.160	13.528	3.372	15.334	7.603	31.066	1.992	21.127	10.784
3.643	27.334	3.178	1.313	10.962	6.936	3.140	16.877	19.171	6.620
3.775	16.675	1.368	17.583	1.669	11.157	16.432	2.831	7.844	10.745

6. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos; emplee la prueba de Kolmogorov-Smirnov. Compruebe con la herramienta Stat::Fit de ProModel.

16.032	24.076	16.463	21.151	14.817	14.702	27.014	12.165	16.597	21.404
18.825	19.364	18.515	14.240	24.154	19.916	16.238	20.795	25.924	18.874
17.532	16.713	16.677	18.739	14.206	19.501	18.590	18.587	19.929	25.354
12.858	16.452	17.487	22.658	22.240	17.471	16.537	23.960	14.417	18.338
28.501	16.939	17.926	24.477	17.673	22.422	13.373	21.971	20.549	24.509

7. Emplee la prueba Chi-cuadrada para determinar, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos. Compruebe con la herramienta Stat::Fit de ProModel.

0.904	0.598	0.081	2.756	0.151	1.662	0.223	0.531	1.229	0.347
1.228	0.235	2.060	1.182	0.280	7.860	0.664	2.898	2.815	0.121
2.294	2.087	1.424	1.525	0.754	7.145	0.754	1.962	1.613	0.003
1.337	3.399	1.639	3.591	2.393	0.412	3.258	0.256	1.419	0.156
2.775	0.355	0.046	1.243	0.776	0.585	0.667	0.123	1.202	6.985

8. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos mediante la prueba de Kolmogorov-Smirnov. Compruebe con Stat:Fit.

4.548	3.136	5.366	1.979	6.097	3.823	5.520	4.203	4.972	8.429
3.242	4.705	5.919	5.530	6.891	5.997	6.640	6.376	6.860	5.991
6.303	6.476	8.503	3.863	1.738	2.913	5.171	6.856	5.665	3.396
5.225	5.966	4.743	7.228	6.030	6.184	7.600	5.716	5.781	4.465
5.307	8.546	6.093	4.720	5.771	4.521	3.715	5.368	1.871	1.629

9. A partir de la prueba Chi-cuadrada determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos. Compruebe con Stat:Fit

6.536	8.441	3.822	6.176	5.059	5.325	6.476	4.229	5.619	4.062
4.769	4.484	2.938	6.459	3.083	6.199	2.590	7.407	7.001	8.501
3.154	3.546	6.316	4.364	8.986	4.195	2.952	3.590	7.356	6.269
5.427	3.431	6.532	6.101	2.625	4.463	7.900	3.715	4.881	7.410
3.404	5.769	2.917	6.739	7.049	5.743	5.448	3.958	6.632	7.036

10. Utilice la prueba de Anderson-Darling para determinar, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos. Compruebe con Stat:Fit.

1.018	0.283	1.672	-0.289	-1.343	-0.418	1.317	0.249	0.937	-0.670
-1.322	-0.296	-1.638	1.970	-0.541	1.567	-1.717	0.125	-0.608	1.027
2.295	-0.952	0.431	2.210	-0.477	0.913	-0.697	-0.145	-1.088	0.137
-1.108	-0.281	0.564	0.683	-0.691	0.010	-0.429	-1.420	-0.070	1.517
0.095	-0.104	1.240	-0.354	-1.525	1.077	0.200	-0.959	-0.144	-1.169

11. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos; utilice la prueba de Kolmogorov-Smirnov.

7.982	40.122	5.862	21.920	7.902	10.824	22.258	13.343	11.045	23.603
18.951	12.348	8.725	11.536	10.187	11.442	13.396	13.070	13.668	7.954
6.361	6.405	34.450	24.956	5.442	12.996	5.073	13.620	11.020	11.729
3.382	14.387	10.037	5.481	2.969	7.503	4.159	23.466	5.219	11.713
37.134	21.099	9.021	6.080	9.053	5.178	18.700	9.056	6.647	5.767
17.684	8.814	22.939	2.491	10.123	3.244	9.433	11.774	3.271	10.390
6.839	7.073	10.708	25.237	7.568	1.152	8.059	26.399	29.285	22.350
3.274	7.325	10.046	9.888	13.798	15.255	20.507	11.147	19.691	7.711
22.836	11.811	14.650	2.898	20.041	10.228	9.553	19.870	8.520	26.182
12.427	14.432	24.699	6.848	7.197	12.156	1.674	8.582	16.293	16.126

12. A partir de la prueba Chi-cuadrada, determine con un nivel de confianza de 90% qué tipo de distribución siguen los datos.

5.091	11.319	3.274	3.366	12.233	5.725	9.186	8.232	6.545	6.481
6.752	10.640	7.242	2.910	8.391	2.288	4.582	6.114	9.965	10.643
11.584	13.333	10.081	11.892	14.542	9.851	11.088	6.301	5.350	3.465
9.595	13.784	4.867	3.171	7.782	5.682	9.587	12.519	9.964	1.298
7.556	8.120	6.451	10.263	5.367	3.059	6.341	3.613	3.068	7.291
4.179	10.035	5.599	5.582	4.836	8.663	6.975	8.441	2.064	5.147
13.470	1.512	11.317	9.799	7.825	9.464	7.799	6.929	8.915	8.007
9.710	5.259	8.086	4.141	2.972	14.575	2.248	6.565	13.418	5.238
14.135	5.937	2.954	9.264	14.970	6.742	5.551	5.313	6.348	5.723
12.436	8.153	5.418	4.028	6.515	9.474	6.817	10.190	4.961	13.263

13. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos; utilice la prueba de Kolmogorov-Smirnov.

1.338	0.530	0.580	0.102	0.285	0.725	5.567	6.773	0.101	5.549
1.198	0.049	0.294	3.661	3.072	5.193	0.329	2.721	0.988	0.716
1.852	1.426	1.586	0.664	6.032	0.093	3.856	1.779	1.729	1.456
3.050	4.688	4.117	2.350	2.954	0.883	1.790	3.847	2.659	3.622
1.003	0.460	1.645	2.342	3.983	1.517	0.695	3.564	0.573	0.204
0.286	1.581	0.395	3.986	2.416	0.577	0.617	1.494	0.468	1.037
2.283	0.774	0.483	0.852	4.342	0.064	0.299	0.214	3.294	0.345
0.388	0.429	2.156	4.276	0.371	4.520	0.408	0.113	0.240	2.923
0.095	2.267	1.355	2.859	0.799	4.718	8.664	0.339	1.892	1.262
0.265	2.032	7.487	2.852	0.040	1.860	0.716	3.551	0.493	0.269

14. Determine, con un nivel de confianza de 95%, qué tipo de distribución siguen los datos; emplee la prueba de Kolmogorov-Smirnov. Compruebe con Stat::Fit.

20.931	18.562	26.714	25.275	24.580	22.090	19.608	15.447	29.631	28.821
28.013	26.693	29.751	22.189	20.807	27.339	22.556	24.069	15.724	21.614
25.570	18.746	16.818	29.122	27.190	26.915	26.844	19.573	26.853	17.053
22.518	18.883	26.128	24.007	28.127	25.213	19.964	27.141	25.458	26.060
29.791	17.890	15.515	24.985	17.717	19.063	29.986	24.074	23.517	20.733

15. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos mediante la prueba de Anderson-Darling. Compruebe con Stat::Fit.

7.717	7.971	5.261	5.994	7.215	6.100	6.876	7.514	6.409	6.679
6.377	6.155	7.512	7.936	7.960	6.157	5.796	7.579	6.450	6.719
6.150	5.983	5.091	7.492	6.974	5.386	6.347	5.053	5.129	5.922
6.174	5.962	5.153	6.838	5.741	5.478	5.471	7.745	5.057	5.548
7.814	6.238	7.484	6.150	7.561	7.734	5.595	7.587	5.235	7.872

16. Utilice la prueba Chi-cuadrada para determinar, con un nivel de confianza de 95%, qué tipo de distribución siguen los datos. Compruebe con Stat::Fit.

71.680	81.149	97.491	68.180	95.076	63.185	75.425	80.150	68.181	97.844
86.478	91.051	95.882	95.804	62.614	92.978	97.926	69.716	70.205	73.864
89.360	97.891	83.945	60.747	75.734	83.704	93.645	84.366	64.310	86.950
95.181	91.325	72.550	63.391	96.829	90.108	75.107	68.775	92.229	77.148
90.591	83.537	91.262	69.235	69.346	74.473	80.042	68.510	63.499	89.607

17. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos mediante la prueba Chi-Cuadrada. Compruebe con Stat::Fit.

2.076	3.724	5.187	3.775	1.443	4.098	2.919	4.043	1.677	3.501
3.137	1.806	4.968	2.370	3.009	3.964	4.057	4.035	2.499	3.200
3.983	4.102	4.574	5.331	3.309	3.351	3.722	6.430	3.587	5.006
1.556	4.993	4.328	2.749	4.374	3.018	4.606	2.682	4.870	2.727
3.899	5.828	1.874	1.316	3.361	2.639	2.246	4.463	4.164	6.351

18. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos. Emplee la prueba Chi-cuadrada y compruebe con Stat::Fit.

2.076	3.724	5.187	3.775	1.443	4.098	2.919	4.043	1.677	3.501
3.137	1.806	4.968	2.370	3.009	3.964	4.057	4.035	2.499	3.200
3.983	4.102	4.574	5.331	3.309	3.351	3.722	6.430	3.587	5.006
1.556	4.993	4.328	2.749	4.374	3.018	4.606	2.682	4.870	2.727
3.899	5.828	1.874	1.316	3.361	2.639	2.246	4.463	4.164	6.351

19. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos; utilice la prueba de Anderson-Darling. Compruebe con Stat::Fit.

191.088	178.781	199.534	191.382	173.618	193.244	185.665	192.927	175.524	189.795
176.583	187.396	198.267	181.183	186.406	192.473	193.006	192.881	182.241	187.850
193.267	192.581	195.991	200.364	188.597	188.870	191.077	206.708	190.292	198.489
198.414	174.540	194.575	184.283	194.842	186.476	196.176	183.730	197.700	184.097
203.231	192.099	177.140	172.582	188.939	183.386	180.174	195.355	193.626	206.255

20. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos. Emplee la prueba Chi-cuadrada y compruebe con Stat::Fit.

17.264	18.129	15.235	15.281	16.162	18.089	14.601	12.091	17.516	12.258
14.938	17.223	19.787	13.435	16.082	13.025	18.482	19.439	15.032	10.816
16.751	16.805	18.580	15.663	13.163	16.930	18.475	18.064	16.390	15.989
14.257	14.772	17.520	15.959	17.579	18.917	18.737	13.028	13.534	14.233
16.651	17.175	17.510	14.658	15.845	17.552	18.398	10.877	13.538	13.217

21. Utilice la prueba de Kolmogorov-Smirnov para determinar, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos. Compruebe con Stat::Fit.

2.865	4.419	3.681	6.502	1.141	2.773	2.299	4.589	7.142	1.783
2.336	2.201	1.186	3.610	0.753	2.653	3.574	3.588	3.128	3.100
3.420	1.123	3.264	2.219	1.962	2.915	4.282	4.835	3.057	1.000
1.242	3.725	4.317	1.694	3.286	3.698	3.208	1.628	3.704	1.020
3.117	1.283	3.821	0.943	1.713	4.715	1.740	2.769	2.877	3.956

22. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos. Utilice la prueba de Kolmogorov-Smirnov y compruebe con Stat::Fit.

82.842	208.481	261.858	54.809	989.300	64.625	334.560	53.719	75.188	59.444
44.148	102.905	185.161	32.573	1231.860	1176.460	175.667	53.632	105.450	125.164
647.174	183.747	113.870	78.092	64.731	332.889	139.595	153.120	57.566	83.781
114.911	431.138	274.864	877.946	85.668	222.815	47.822	114.754	416.533	337.344
169.609	251.844	210.369	106.671	56.651	98.984	168.127	476.397	86.082	215.320

23. A partir de la prueba de Kolmogorov-Smirnov, determine con un nivel de confianza de 90% qué tipo de distribución siguen los datos. Compruebe con Stat::Fit.

0.488	0.116	0.731	0.094	0.684	0.093	0.368	0.090	0.761	0.420
0.995	0.908	0.183	0.146	0.633	0.567	0.058	0.507	0.780	0.139
0.088	0.382	0.707	0.413	0.581	0.254	0.440	0.447	0.251	0.870
0.149	0.427	0.743	0.434	0.260	0.738	0.300	0.302	0.314	0.423
0.731	0.313	0.908	0.845	0.937	0.607	0.025	0.302	0.608	0.078

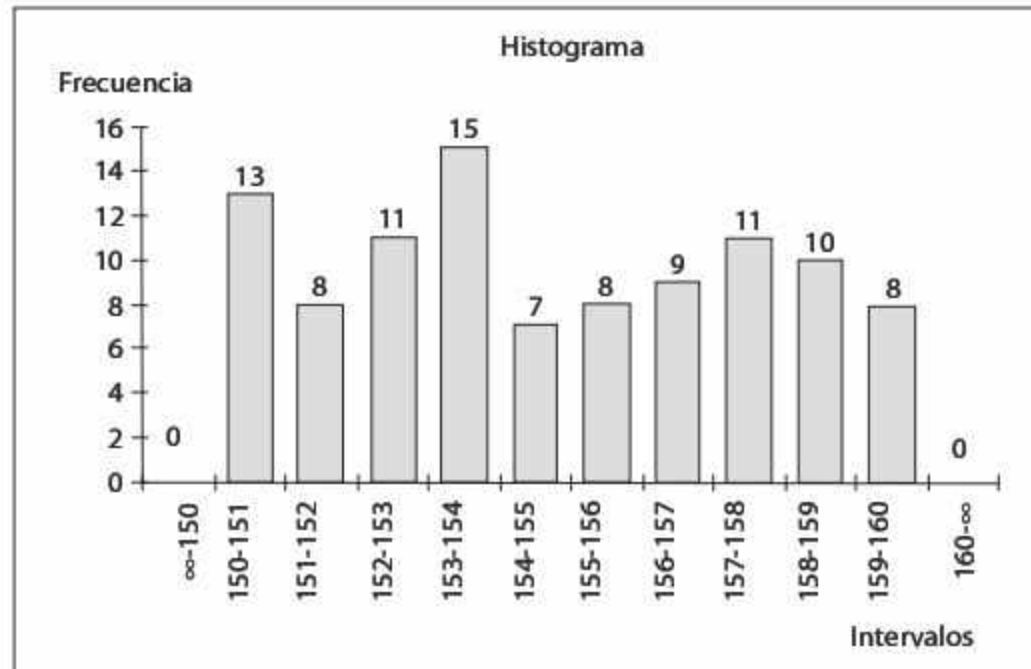
24. Determine, con un nivel de confianza de 90%, si la variable aleatoria representada por la siguiente tabla de frecuencia sigue una distribución exponencial con media 1.

Intervalo	Frecuencia
0-1	29
1-2	18
2-3	18
3-4	12
4-5	8
5-6	3
6-7	2
7-8	5
8-9	3
9-10	0
10-∞	2

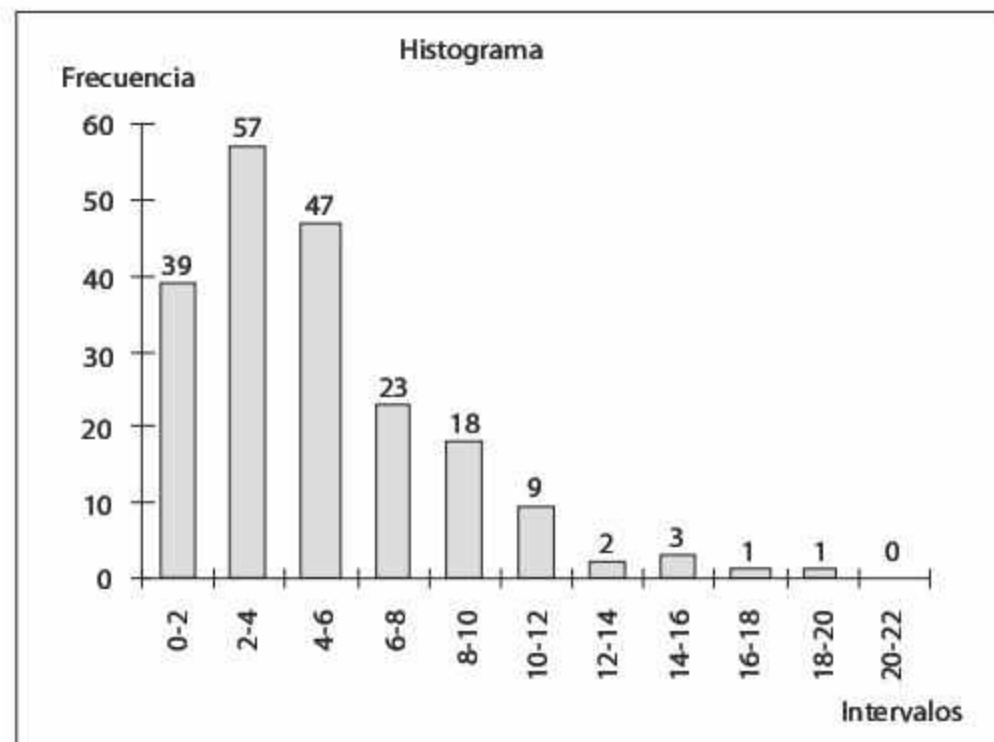
25. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos. *a)* Emplee la prueba de Anderson-Darling, *b)* emplee la prueba Chi-cuadrada.

Intervalo	Frecuencia
-∞-12	3
12.0-12.5	4
12.5-13.0	3
13.0-13.5	6
13.5-14.0	8
14.0-14.5	15
14.5-15.0	12
15.0-15.5	13
15.5-16.0	7
16.0-16.5	16
16.5-17.0	7
17.0-∞	6

26. Determine, con un nivel de confianza de 95%, qué tipo de distribución siguen los datos. Utilice la prueba Chi-cuadrada.



27. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos; emplee la prueba Chi-cuadrada.



28. Utilice la prueba Chi-cuadrada para determinar con un nivel de confianza de 95% qué tipo de distribución siguen los datos.

4	5	3	5	5	4	3	2	4	3
4	6	4	3	5	2	2	3	3	4
4	4	3	3	3	2	2	3	2	3
3	4	5	2	3	4	3	3	5	3
2	5	3	4	4	1	4	5	4	5
7	2	4	4	2	4	1	5	4	4
5	5	5	2	4	4	5	4	4	1
6	4	6	6	2	4	4	2	2	2
3	3	2	5	3	5	1	3	2	4
1	2	5	2	3	1	5	3	2	5

29. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos. Use la prueba de Kolmogorov-Smirnov.

1	0	0	1	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0	0
0	0	1	0	0	0	0	0	1	1
1	0	1	2	1	1	1	0	0	0
1	0	1	2	0	0	1	1	0	1
0	1	1	0	0	1	1	2	1	0
1	2	0	1	1	1	0	1	0	1
1	0	1	2	0	0	1	2	1	0
1	0	0	2	0	0	0	0	1	0
0	0	0	1	0	1	2	1	0	0

30. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos; use la prueba Chi-cuadrada y compruebe con Stat::Fit.

5	3	2	4	6	4	3	2	2	0
3	2	4	6	2	8	6	3	3	3
11	5	9	3	7	4	7	2	7	3
4	7	7	3	3	4	8	4	2	7
5	4	5	3	6	4	4	3	1	3

31. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos. Compruebe con la herramienta Stat::Fit.

10	9	8	10	9	8	13	10	16	13
14	15	11	6	12	10	5	10	9	12
8	7	8	9	10	8	10	5	3	13
9	12	15	7	5	9	14	12	11	10
4	12	13	7	7	7	18	5	9	10

32. A partir de la prueba Chi-cuadrada, determine con un nivel de confianza de 90% qué tipo de distribución siguen los datos.

2	1	2	0	1	1	2	1	0	1
1	2	2	1	0	0	0	0	1	1
1	1	0	2	2	0	0	2	1	0
2	0	0	1	2	1	0	0	2	2
1	1	1	1	2	1	1	1	2	0
2	0	3	2	0	0	1	2	1	1
1	2	3	0	0	0	0	1	1	0
1	1	1	0	2	0	0	0	3	1
0	1	0	1	0	1	1	2	3	2
1	2	0	0	1	1	0	0	2	0

33. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos; emplee la prueba Chi-cuadrada.

0	0	0	0	0	0	0	1	0	0
0	0	1	0	1	0	1	0	1	0
0	1	1	0	0	1	1	1	0	1
0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	1	1	0
0	0	0	0	0	1	0	0	1	1
0	1	1	1	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0
1	1	0	0	0	1	0	0	0	0
0	1	0	0	0	0	0	1	0	0

34. Determine, con un nivel de confianza de 95%, qué tipo de distribución siguen los datos; emplee la prueba Chi-cuadrada.

8	8	6	6	4	7	5	4	6	6
8	8	5	4	6	5	4	4	6	4
6	8	6	8	7	5	4	6	5	6
7	5	8	7	6	8	5	5	4	7
6	8	8	5	6	7	4	4	7	6
4	5	6	5	8	4	5	8	4	8
6	5	7	4	7	8	8	4	5	4
5	7	6	6	5	5	6	5	5	5
5	6	6	5	8	4	8	5	6	7
4	4	4	4	7	7	8	8	5	7

35. Utilice la prueba Chi-cuadrada para determinar, con un nivel de confianza de 95%, qué tipo de distribución siguen los datos. Compruebe con Stat::Fit.

1	2	0	1	2	1	0	0	1	2
2	1	1	1	1	2	0	2	2	1
1	1	2	1	1	0	0	0	1	1
0	1	0	0	1	0	2	0	1	2
0	1	1	1	0	0	0	0	1	0

36. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos. Utilice la prueba de Kolmogorov-Smirnov y compruebe con Stat::Fit.

4	5	4	4	4	4	5	4	5	3
5	5	4	4	3	5	5	4	4	5
3	3	4	5	3	5	4	4	4	5
3	5	5	5	5	3	5	5	3	3
5	5	4	5	4	5	5	4	4	4

37. Determine, con un nivel de confianza de 95%, qué tipo de distribución siguen los datos; emplee la prueba Chi-cuadrada y compruebe con Stat::Fit.

x	Frecuencia
0	11
1	29
2	31
3	14
4	9
5-∞	6

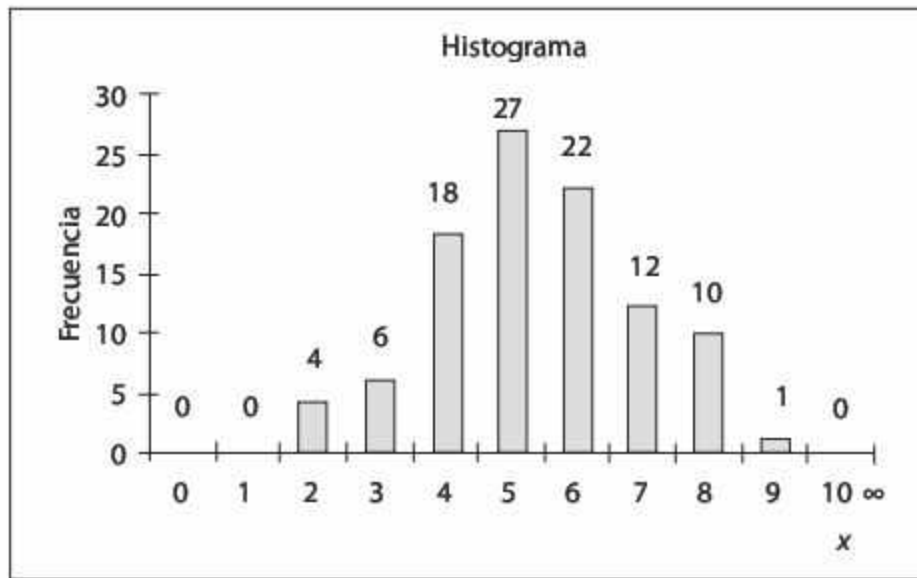
38. Determine, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos; use la prueba Chi-cuadrada.

0	1	0	2	2	1	0	2	5	0
6	0	0	0	3	1	4	0	0	2
1	2	1	0	1	4	0	0	0	1
2	2	2	0	1	0	0	0	4	3
0	1	0	0	2	1	0	0	9	3
4	2	0	1	8	0	2	0	6	1
1	1	0	3	2	0	0	0	3	1
0	5	0	0	1	1	1	0	1	0
7	0	0	2	0	2	0	2	0	2
0	0	5	9	1	2	4	0	0	0

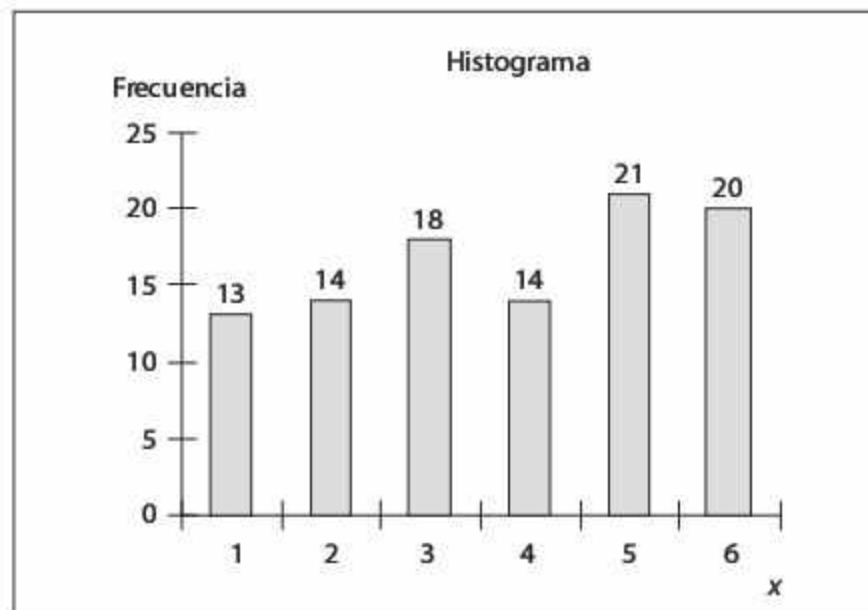
39. Emplee la prueba Chi-cuadrada para determinar, con un nivel de confianza de 90%, qué tipo de distribución siguen los datos.

x	Frecuencia
0	21
1	17
2	14
3	9
4	8
5	6
6	6
7	1
8	4
9	4
10	1
11	0
12	1
13	2
14-∞	6

40. Determine, con un nivel de confianza de 90%, si los datos se distribuyen de acuerdo con una distribución binomial con $N = 10$ y $p = 0.5$.



41. Determine, con un nivel de confianza de 90%, si los datos se distribuyen de acuerdo con una distribución uniforme discreta (1-6).



42. Genere, mediante el método de la transformada inversa, 100 números aleatorios para las siguientes distribuciones de probabilidad.

a) $p(x) = \frac{1}{5} \left(\frac{4}{5}\right)^{x-1}$ para $x = 1, 2, 3, \dots$

b) $p(x) = \frac{1}{2} \left(\frac{1}{2}\right)^x$ para $x = 0, 1, 2, 3, \dots$

$$c) p(x) = \frac{20+10x}{200} \quad \text{para } x = 0, 1, 2, 3, 4$$

$$d) p(x) = \frac{18-4x}{50} \quad \text{para } x = 0, 1, 2, 3, 4$$

$$e) p(x) = \frac{3x+8}{100} \quad \text{para } x = 2, 3, 4, 5, 6$$

$$f) p(x) = \frac{1}{6} \quad \text{para } x = 6, 7, 8, 9, 10, 11$$

$$g) p(x) = \frac{p(1-p)^{x-1}}{1-(1-p)^n} \quad \text{para } x = 1, 2, \dots, n$$

43. Obtenga, con el método de la transformada inversa, la expresión matemática para generar variables aleatorias que sigan las funciones de densidad indicadas.

$$a) f(x) = \frac{x}{5} e^{-\frac{x^2}{10}} \quad \text{para } x \geq 0$$

$$b) f(x) = 36x^2 e^{-12x^3} \quad \text{para } x \geq 0$$

$$c) f(x) = 4x^3 e^{-x^4} \quad \text{para } x \geq 0$$

44. En los siguientes incisos genere variables aleatorias con densidad $f(x)$ mediante los números aleatorios 0.123, 0.765, 0.893, 0.563, 0.642, 0.225, 0.334, 0.076, 0.984 y 0.445. Use el método de la transformada inversa.

$$a) f(x) = \begin{cases} 3/4 & 0 \leq x \leq 1 \\ 1/4 & 1 \leq x \leq 2 \end{cases}$$

$$b) f(x) = \begin{cases} \frac{1}{12}(x-7) & 7 \leq x \leq 10 \\ \frac{1}{20}(15-x) & 10 \leq x \leq 15 \end{cases}$$

$$c) f(x) = \begin{cases} \frac{1}{12}(x-8) & 8 \leq x \leq 11 \\ \frac{5}{8} & 11 \leq x \leq 12 \end{cases}$$

$$d) f(x) = \begin{cases} \frac{1}{6} & 8 \leq x \leq 10 \\ \frac{1}{12} & 10 \leq x \leq 14 \\ \frac{1}{3} & 14 \leq x \leq 15 \end{cases}$$

45. Obtenga las funciones generadoras mediante el método de composición de las siguientes funciones de densidad:

$$a) f(x) = \begin{cases} 3/4 & 0 \leq x \leq 1 \\ 1/4 & 1 \leq x \leq 2 \end{cases}$$

$$b) f(x) = \begin{cases} \frac{1}{12}(x-7) & 7 \leq x \leq 10 \\ \frac{1}{20}(15-x) & 10 \leq x \leq 15 \end{cases}$$

$$c) f(x) = \begin{cases} \frac{1}{12}(x-8) & 8 \leq x \leq 11 \\ \frac{5}{8} & 11 \leq x \leq 12 \end{cases}$$

$$d) f(x) = \begin{cases} \frac{1}{6} & 2 \leq x \leq 4 \\ \frac{1}{3} & 4 \leq x \leq 5 \\ \frac{1}{12} & 5 \leq x \leq 9 \end{cases}$$

$$e) f(x) = \begin{cases} \frac{1}{2}e^x & -\infty \leq x \leq 0 \\ \frac{1}{2}e^{-x} & 0 \leq x \leq \infty \end{cases}$$

46. Genere 100 variables aleatorias para las siguientes distribuciones de probabilidad; utilice el método de la transformada inversa.

$$a) f(x) = \frac{(x-4)^2}{72} \quad \text{para } 0 \leq x \leq 4 + \sqrt[3]{152}$$

$$b) f(x) = \frac{(x-4)^2}{72} \quad \text{para } 4 \leq x \leq 10$$

$$c) f(x) = \frac{1}{5} \quad \text{para } 2 \leq x \leq 7$$

$$d) f(x) = \frac{x}{2} \quad \text{para } 0 \leq x \leq 2$$

$$e) f(x) = \frac{x^2}{4} \quad \text{para } 1 \leq x \leq \sqrt[3]{13}$$

$$f) f(x) = \frac{(x-14)}{200} \quad \text{para } 14 \leq x \leq 34$$

47. Genere 50 variables aleatorias para las siguientes distribuciones de probabilidad; emplee el método de la transformada inversa.

$$a) f(x) = \frac{(x-1)}{4} \quad \text{para } 1 \leq x \leq 1 + \sqrt{8}$$

$$b) f(x) = \frac{1}{6} \quad \text{para } 7 \leq x \leq 13$$

$$c) f(x) = \frac{3}{2}x \quad \text{para } 1 \leq x \leq \sqrt{7/3}$$

$$d) f(x) = \frac{1}{0.3}x^2 \quad \text{para } 2 \leq x \leq \sqrt[3]{8.9}$$

$$e) f(x) = \frac{1}{0.8}x^2 \quad \text{para } 0 \leq x \leq \sqrt[3]{2.4}$$

48. Obtenga las expresiones para generar variables aleatorias con la distribución de probabilidad:

$$a) f(x) = \frac{3}{28}(x+1)^2 \quad \text{para } -2 \leq x \leq 2$$

$$b) f(x) = \frac{2}{9}(x-9) \quad \text{para } 9 \leq x \leq 12$$

$$c) f(x) = \frac{2}{9}(8-x) \quad \text{para } 5 \leq x \leq 8$$

$$d) f(x) = \frac{1}{8}(15-x) \quad \text{para } 11 \leq x \leq 15$$

49. Determine los valores de X para los números aleatorios 0.0234, 0.2456, 0.7867, 0.426, 0.9845, 0.124, 0.862, 0.555, 0.3345 y 0.6735, puesto que X son variables aleatorias con las siguientes distribuciones de probabilidad.

$$a) f(x) = \frac{1}{5}e^{-\frac{x}{5}} \quad \text{para } x \geq 0$$

$$b) f(x) = \frac{1}{6}(x-)e^{-\frac{x-3}{6}} \quad \text{para } x \geq$$

50. Mediante cualquier hoja de cálculo, genere 50 variables aleatorias.

a) Distribuidas de manera exponencial con $\lambda = 5$.

b) Distribuidas de forma normal con media 50 y varianza 36.

c) Distribuidas de manera uniforme con límite inferior igual a 20 y límite superior igual a 100.

- d) Distribuidas triangularmente con límite inferior = 5, valor más probable = 15 y límite superior = 25.
- e) Con distribución binomial y parámetros $N = 5$, $p = 0.3$, $q = 0.7$.
- f) Con distribución de Poisson, con $\lambda = 3$.
- g) Con distribución weibull con parámetro de localización 100, escala 30 y forma 4
- h) Con distribución erlang con parámetro de forma 4 y media 20.

Compruebe con Stat::Fit si las variables aleatorias generadas siguen la distribución de probabilidad que se esperaría de ellas.

- 51. Mediante una hoja de cálculo, genere 200 variables aleatorias distribuidas normalmente con media 30 y varianza 25. Use la fórmula $x = 5z + 30$ donde el valor de z representa una variable aleatoria con distribución normal estándar obtenida a partir de la función = NORMSINV(RAND())
- 52. Mediante cualquier hoja de cálculo, genere 50 variables aleatorias distribuidas de manera triangular con límite inferior = 12, valor más probable = 18 y límite superior = 25, use:
 - a) El método de la transformada inversa.
 - b) El método de composición.

Referencias

- [1] Azarang, M. y García, E. (1996). *Simulación y Análisis de Modelos Estocásticos* (1a ed.): McGraw Hill.
- [2] Banks, J., Carson, J.S., Nelson, B.L., y Nicol, D.M. (2005). *Discrete-Event System Simulation* (4th ed.): Prentice Hall N.J.
- [3] Law, A.M. y Kelton, W.D. (2000). *Simulation Modeling and Analysis* (3th ed.): McGraw Hill.



Capítulo 4

Simulación de variables aleatorias

- 4.1** Verificación y validación de los modelos de simulación
- 4.2** Simulaciones no terminales o de estado estable
- 4.3** Modelos de simulación
- 4.4** Selección de lenguajes de simulación
- 4.5** Caso de estudio 1
- 4.6** Caso de estudio 2
- 4.7** Problemas

4.1 Verificación y validación de los modelos de simulación

Gracias al avance tecnológico, en la actualidad existen en el mercado aplicaciones con interfaces gráficas tan poderosas que permiten a muchos usuarios con inclinaciones técnicas desarrollar modelos en el área de la simulación. Por desgracia, en general, dichos usuarios, aunque aprenden a usar el lenguaje relacionado y manejan algunos de los conceptos básicos, ponen muy poca atención al análisis correcto de los resultados. Así, muchos estudios son interpretados de manera errónea y es muy probable que conduzcan, en consecuencia, a malas decisiones.

El fenómeno que acabamos de describir ocurre por razones como éstas: en primer lugar, el falso sentido de seguridad que desarrolla el usuario por el simple hecho de conocer el lenguaje utilizado en el área; la facilidad de uso del software de simulación actual y su capacidad para desarrollar gráficos y animaciones y, sobre todo, la dificultad implícita en el análisis estadístico de la información. Es muy común encontrar personas que después de simular un sistema estocástico aseguran de manera bastante ingenua que el resultado de la variable de respuesta es un valor único —por ejemplo, que el número de piezas que se acumulan ante una máquina es tan sólo el promedio de la variable—, y dejan de lado un completo análisis estadístico de dicha variable. Para evitar que el lector se convierta en uno de esos usuarios, aquí se discutirán los aspectos mínimos que deben cuidarse en el análisis de las variables de salida.

Para empezar, debemos distinguir dos categorías entre los modelos de simulación: modelos de categoría terminal y modelos no terminales o de estado estable. A continuación se explica esta clasificación con más detalle.

4.1.1 Simulaciones terminales

Los modelos de tipo terminal tienen como característica principal la ocurrencia de un evento que da por terminada la simulación. Un ejemplo sería el siguiente: digamos que nos interesa conocer el tiempo que llevaría procesar un lote de 10 piezas, el tiempo requerido para vender 100 periódicos, o el número de clientes que se atiende en una cafetería entre las 8:00 y las 9:00 a.m. El análisis estadístico recomendado para este tipo de simulaciones involucra la utilización de intervalos de confianza y la determinación de la distribución de probabilidad de la variable de salida.

4.1.1.1 Intervalos de confianza

Debido a la naturaleza aleatoria de los resultados de este tipo de modelos, es necesario determinar su distribución de probabilidad y su intervalo de confianza en las diferentes réplicas. En la sección 3.3 del capítulo anterior se discute cómo obtener la distribución de probabilidad de una variable aleatoria; por lo tanto, aquí nos ocuparemos de los intervalos de confianza.

Si la variable aleatoria sigue una distribución normal, el intervalo de confianza está dado por:

$$IC = \left[\bar{x} - \frac{s}{\sqrt{r}}(t_{\alpha/2, r-1}) \quad , \quad \bar{x} + \frac{s}{\sqrt{r}}(t_{\alpha/2, r-1}) \right]$$

En caso de que la variable aleatoria siga otro tipo de distribución, el intervalo de confianza es relativamente más amplio, y se calcula como:

$$IC = \left[\bar{x} - \frac{s}{\sqrt{r\alpha}} \quad , \quad \bar{x} + \frac{s}{\sqrt{r\alpha}} \right]$$

En ambas ecuaciones:

r = Número de réplicas

α = Nivel de rechazo

$$\bar{x} = \frac{1}{r} \sum_{i=1}^r x_i$$

$$s = \left(\frac{1}{r-1} \sum_{i=1}^r (x_i - \bar{x})^2 \right)^{1/2}$$

Ejemplo 4.1

Los resultados de 10 réplicas de la simulación de un sistema de inventario promedio en un almacén son 190.3, 184.2, 182.4, 195.6, 193.2, 190.5, 191.7, 188.5, 189.3, 188.4. Determine el intervalo de confianza con un nivel de aceptación de 95%.

La media y la desviación estándar de la muestra son:

$$\bar{x} = 189.41$$

$$s = 3.916$$

Bajo la premisa de que el inventario promedio sigue una distribución normal, el intervalo de confianza se calcularía de la siguiente forma:

$$IC = \left[\bar{x} - \frac{s}{\sqrt{r}} (t_{\alpha/2, r-1}) \quad , \quad \bar{x} + \frac{s}{\sqrt{r}} (t_{\alpha/2, r-1}) \right]$$

$$IC = \left[189.41 - \frac{3.916}{\sqrt{10}} (t_{0.025, 9}) \quad , \quad 189.41 + \frac{3.916}{\sqrt{10}} (t_{0.025, 9}) \right],$$

el factor $t_{0.025, 9}$ se obtiene de la tabla de distribución t que se encuentra en los anexos.

$$IC = \left[189.41 - \frac{3.916}{\sqrt{10}} (2.262) \quad , \quad 189.41 + \frac{3.916}{\sqrt{10}} (2.262) \right]$$

$$IC = [186.60, 192.21] \text{ piezas promedio en el almacén.}$$

Si la variable aleatoria no fuera normal, o si la suposición de normalidad se considerara inadecuada, el intervalo de confianza se calcularía así:

$$IC = \left[\bar{x} - \frac{s}{\sqrt{r\alpha}} , \bar{x} + \frac{s}{\sqrt{r\alpha}} \right]$$

$$IC = \left[189.41 - \frac{3.916}{\sqrt{10(0.05)}} , 189.41 + \frac{3.916}{\sqrt{10(0.05)}} \right]$$

$$IC = [183.87, 194.94] \text{ piezas promedio en el almacén.}$$

En consecuencia, si la variable aleatoria sigue una distribución normal y realizamos 100 réplicas del experimento, esperamos que el resultado de 95% de las mismas se encuentre entre 186.08 y 192.73, a diferencia de lo que ocurriría en una variable no normal, en donde después de 100 réplicas del experimento esperamos que 95% de ellas se encuentren entre 183.87 y 194.94; lo cual implica mucho menos exactitud.

Ejemplo 4.2

Un cliente ha solicitado la entrega de un pedido de 100 artículos. Como analistas, deseamos conocer un valor estimado del tiempo promedio de entrega del pedido, y un intervalo de confianza con un nivel de 90%.

Por el hecho de que este problema exige una simulación de tipo terminal, debemos desarrollar un modelo y realizar suficientes réplicas independientes para obtener la distribución de probabilidad y el intervalo de confianza que nos interesan. Con este propósito realizamos 30 réplicas, y obtuvimos los siguientes resultados del tiempo de entrega del producto, medido en días transcurridos desde la formulación del pedido.

9.57	4.81	7.33	5.72	7.90	12.03	6.30	5.83	8.39	9.74
6.55	2.92	10.65	7.85	8.04	4.60	10.20	7.27	6.32	7.08
4.69	6.77	4.85	5.72	9.75	6.33	8.35	8.89	5.83	3.77

En la figura 4.1 se muestran el histograma del tiempo de entrega y el resultado de la prueba de bondad de ajuste.

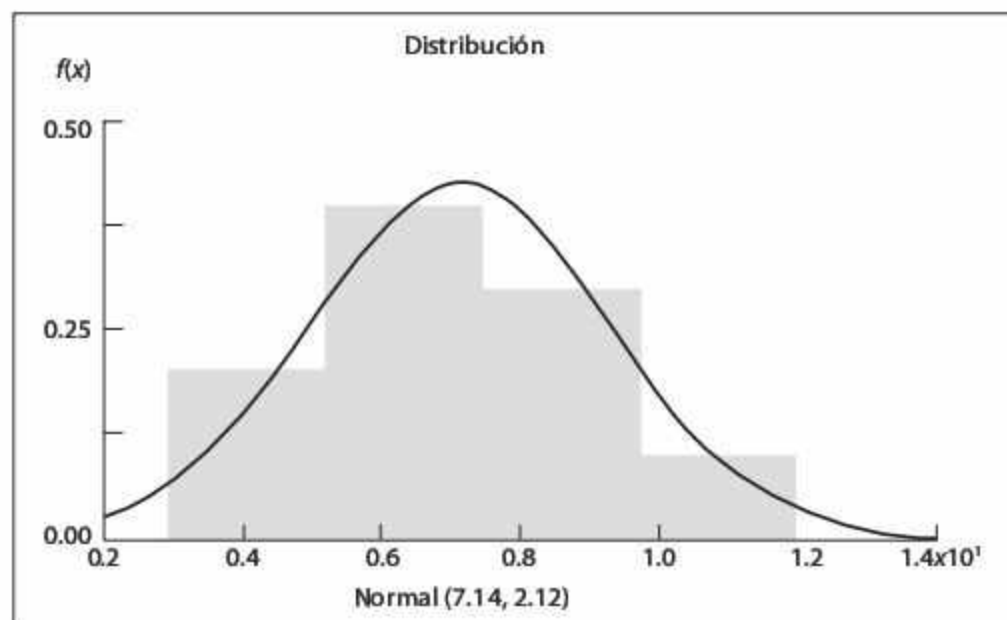


Figura 4.1 Histograma de las réplicas del modelo de tiempo de entrega del producto (ejemplo 4.2).

Si tomamos en cuenta la evidencia estadística de normalidad, así como la media y la desviación estándar de la muestra, calculamos el intervalo de confianza así:

$$\bar{x} = 7.14$$

$$s = 2.12$$

$$IC = \left[\bar{x} - \frac{s}{\sqrt{r}} (t_{\alpha/2, r-1}) \quad , \quad \bar{x} + \frac{s}{\sqrt{r}} (t_{\alpha/2, r-1}) \right]$$

$$IC = \left[7.14 - \frac{2.12}{\sqrt{30}} (t_{0.05, 30}) \quad , \quad 7.14 + \frac{2.12}{\sqrt{30}} (t_{0.05, 30}) \right]$$

nuevamente el factor $t_{0.05, 30}$ se obtiene de la tabla de distribución t que se encuentra en los anexos.

$$IC = \left[7.14 - \frac{2.12}{\sqrt{30}} (1.697) \quad , \quad 7.14 + \frac{2.12}{\sqrt{30}} (1.697) \right]$$

$$IC = [6.46, 7.79] \text{ días}$$

4.2 Simulaciones no terminales o de estado estable

A diferencia de los modelos anteriores, las simulaciones no terminales o de estado estable no involucran una ocurrencia en el tiempo en que tengan que finalizar. Por ejemplo, si deseáramos conocer el número de máquinas que deben instalarse en un sistema de producción cuya operación tiene que mantenerse activa continuamente durante todo el año, podríamos modelar el sistema hasta que la variable de interés llegara a un estado estable. En este caso surge la necesidad de determinar la longitud de la corrida para asegurar la estabilización de los resultados del modelo. Veamos cómo satisfacer dicho requisito.

4.2.1 Longitud de las réplicas

Para que el resultado de una variable aleatoria llegue al estado estable en una simulación no terminal, es necesario garantizar que la longitud de la réplica, n , sea lo suficientemente grande para que la variación entre réplicas no difiera de cierta exactitud, ϵ , el $100(1 - \alpha)\%$ de las veces.

En caso de normalidad, el tamaño de corrida de la simulación se calcula como:

$$n = \left(\frac{\sigma Z_{\alpha/2}}{\epsilon} \right)^2$$

Ejemplo 4.3

Determine la longitud de la réplica para estimar, dentro de un rango de ± 2 , el valor de una variable normal con desviación estándar 4 y un nivel de aceptación de 95% (nivel de rechazo de 5%).

Solución:

$$n = \left(\frac{\sigma Z_{\alpha/2}}{\epsilon} \right)^2$$

$$n = \left(\frac{4 Z_{0.025}}{2} \right)^2 = ((2)(1.96))^2 = 15.36$$

$$n = 16$$

La figura 4.2 representa los resultados de la simulación de seis réplicas independientes de longitud $n = 16$ de una variable aleatoria con distribución normal y desviación estándar 4. Como puede ver, los resultados finales de la variable aleatoria se encuentran dentro de la exactitud deseada. Dado el nivel de aceptación de 95% (nivel de rechazo de 5%), si realizáramos 100 réplicas cabría esperar que cinco de ellas estuvieran fuera de la exactitud deseada.

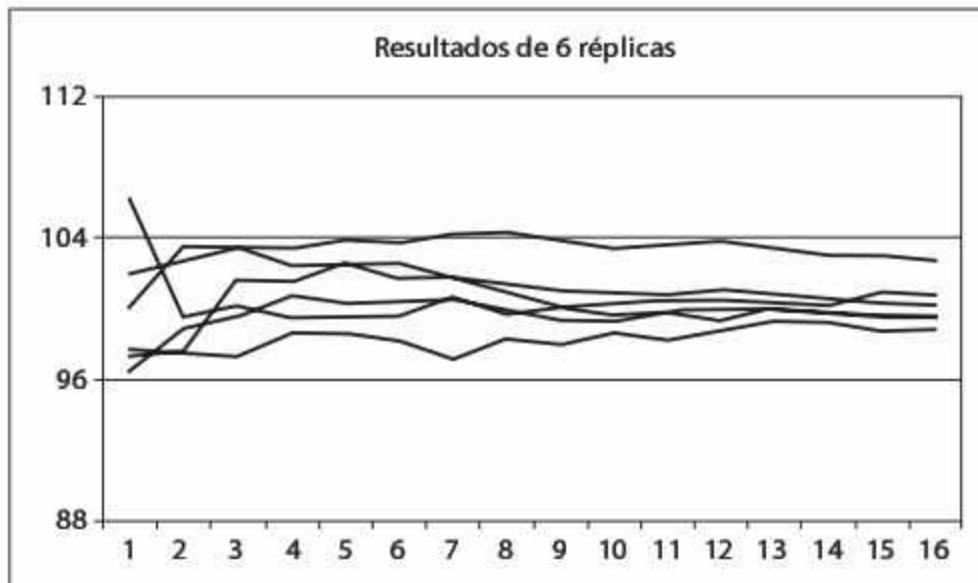


Figura 4.2
Comportamiento del promedio de la variable aleatoria normal con seis réplicas; exactitud deseada de 1σ .

Si se tiene la certeza de normalidad pero se desconoce el valor de la desviación estándar, será necesario realizar una corrida inicial de tamaño n' para determinar un estimador de la desviación. En este caso la longitud de la réplica se determina mediante

$$n = \left(\frac{s}{\epsilon} (t_{\alpha/2, n'-1}) \right)^2$$

Ejemplo 4.4

Se realizó una corrida inicial de tamaño $n' = 10$ para estimar la desviación estándar $s = 13.21$ de una variable normal. Determine la longitud de la réplica para estimar el valor medio dentro de un rango de ± 0.3 con un nivel de aceptación de 95% (nivel de rechazo 5%).

Solución (vea la figura 4.3):

$$n = \left(\frac{s}{\epsilon} (t_{\alpha/2, n'-1}) \right)^2$$

$$n = \left(\frac{13.21}{0.3} (t_{0.025, 9}) \right)^2 = ((44.033)(2.262))^2 = 9920.83$$

$$n = 9920$$

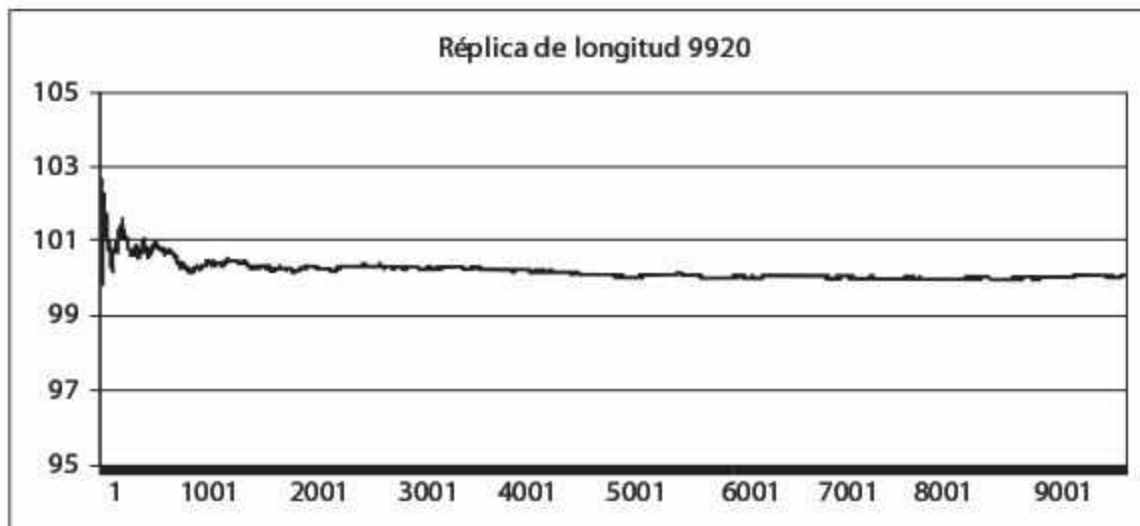


Figura 4.3
Comportamiento del promedio de la variable aleatoria normal; exactitud deseada de 0.045σ .

Cuando se desconoce el tipo de distribución de la variable aleatoria a simular o bien la suposición de normalidad no existe, es preciso hacer uso del teorema de Tchebycheff para calcular la longitud de la réplica. En este caso se utiliza

$$n = \frac{1}{\alpha} \left(\frac{\sigma}{\epsilon} \right)^2$$

Ejemplo 4.5

Supongamos que la suposición de normalidad del ejemplo anterior no es válida, y consideremos la misma información con $n' = 10$ y $s = 1321$. La longitud de la réplica para estimar el valor de la variable dentro de un rango de ± 0.3 con un nivel de aceptación de 90% (nivel de rechazo 10%) es:

$$n = \frac{1}{\alpha} \left(\frac{\sigma}{\epsilon} \right)^2 = \frac{1}{\alpha} \left(\frac{s}{\epsilon} \right)^2$$

$$n = \frac{1}{0.1} \left(\frac{13.21}{0.3} \right)^2 = 19389.3$$

$$n = 19390$$

Este resultado concuerda con el concepto expuesto en el ejemplo 4.1: para distribuciones diferentes, aun con los mismos rangos deseados, los requerimientos de simulación cambian, en este caso respecto de la n . Ahora bien, veamos qué sucede al momento de simular la variable aleatoria exponencial. La figura 4.4 muestra el comportamiento promedio de la variable; a pesar de haberla simulado en 9920 ocasiones, dicha variable está lejos de lograr la exactitud que se observa en la figura 4.3, en donde la variable simulada sigue una distribución normal. Para que el comportamiento de la variable llegue a la zona de estabilización con la exactitud deseada de 0.045σ , es preciso generar 19390 variables exponenciales (al simular solamente 9920 variables exponenciales la exactitud es de 0.063σ).

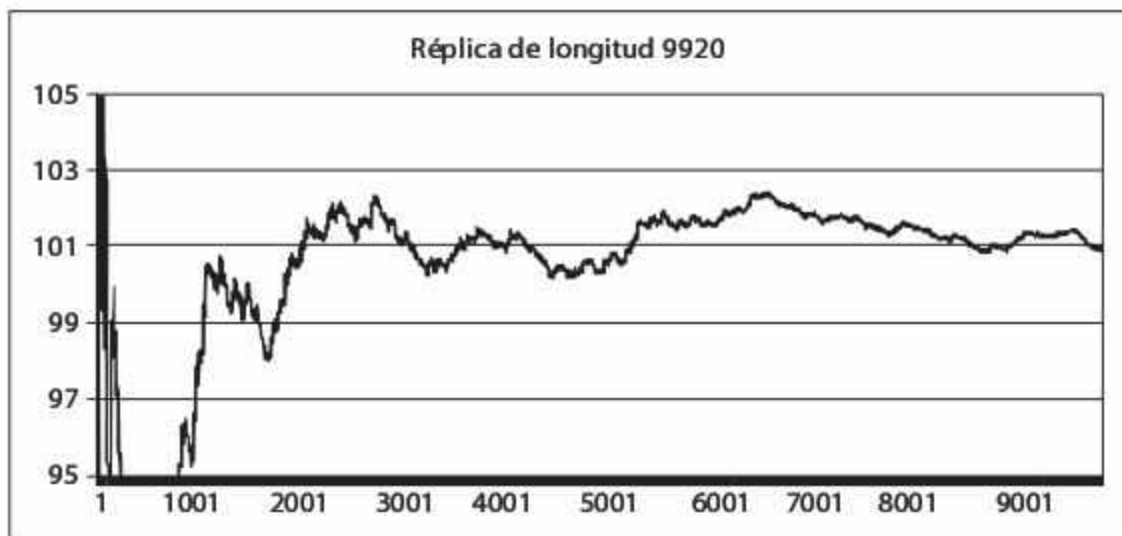


Figura 4.4
Comportamiento del promedio de la variable aleatoria exponencial; exactitud lograda de 0.063σ .

Ejemplo 4.6

Determinar la longitud de la réplica para estimar, dentro de un rango de ± 0.5 con un nivel de aceptación de 90% (nivel de rechazo 10%), el valor promedio de una variable con $\sigma = 8$. Se desconoce la distribución de probabilidad de la variable.

Solución:

$$n = \frac{1}{\alpha} \left(\frac{\sigma}{\epsilon} \right)^2$$

$$n = \frac{1}{0.1} \left(\frac{8}{0.5} \right)^2 = 2560$$

Ejemplo 4.7

Determine la longitud de la réplica, de manera que el estimado del valor promedio de una variable con distribución de Weibull y desviación estándar σ no difiera en más de $\frac{1}{4}$ de la desviación estándar, con un nivel de rechazo de 5% (nivel de aceptación 95%).

Solución:

$$n = \frac{1}{\alpha} \left(\frac{\sigma}{\epsilon} \right)^2$$

$$n = \frac{1}{\alpha} \left(\frac{\sigma}{\frac{1}{4}\sigma} \right)^2 = \frac{1}{\alpha} (4)^2 = \frac{16}{0.05} = 320$$

En la figura 4.5 se muestran los resultados de 10 réplicas de longitud 320; cada una representa el promedio móvil de una variable aleatoria de Weibull con σ de 1.6. De acuerdo con los cálculos del ejemplo, esta longitud de la réplica debería asegurar la estabilización de la variable aleatoria —lo cual aparentemente ocurre, según la gráfica—, con una exactitud de 0.25σ . Los resultados confirman los cálculos, ya que el valor final de las réplicas estuvo entre 54.41 y 54.57. La diferencia de 0.16 equivale a una exactitud relativa a la desviación estándar de apenas 0.1σ .

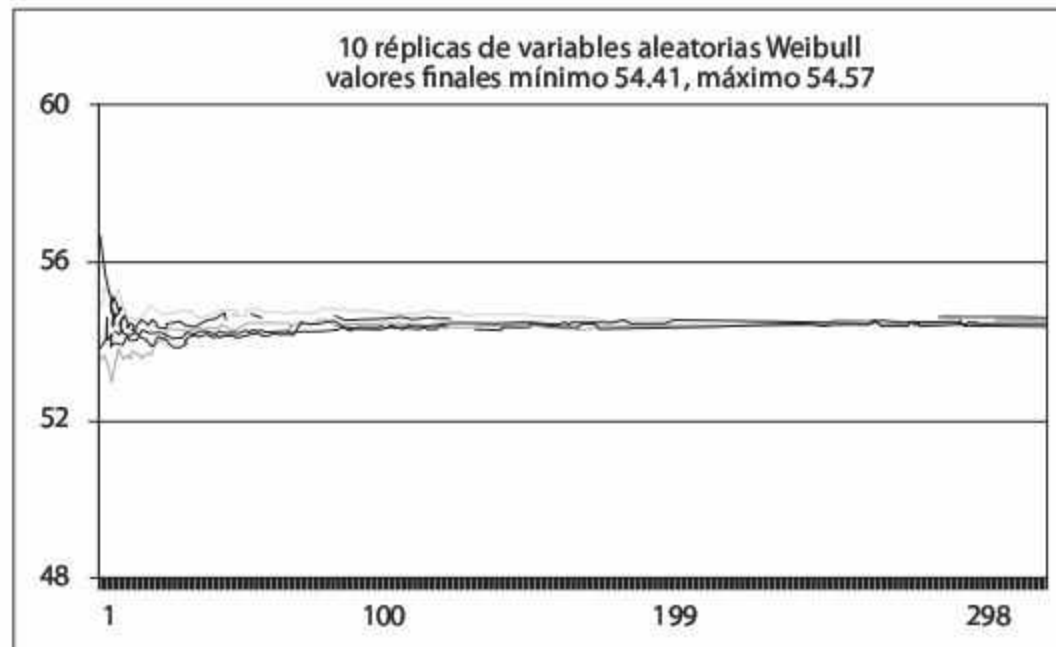


Figura 4.5
Comportamiento del promedio de la variable aleatoria de Weibull; exactitud deseada de 0.25σ .

4.3 Modelos de simulación

Con el propósito de dar una idea de cómo desarrollar un modelo de simulación y de qué manera emplear los conceptos expuestos a lo largo del presente capítulo, a continuación se presentan algunos ejemplos programados en un programa de hoja de cálculo.

4.3.1 Modelo de una línea de espera con un servidor

Ejemplo 4.8

El tiempo que transcurre entre la llegada de ciertas piezas a una estación de inspección sigue una distribución exponencial con media de 5 minutos/pieza. El proceso está a cargo de un operario, y la duración de la inspección sigue una distribución normal con media de 4.0 y desviación estándar de 0.5 minutos/pieza. Calcule el tiempo promedio de permanencia de las piezas en el proceso de inspección.

Para solucionar el problema anterior se debe:

1. Construir una *tabla de eventos* en la que se describa la relación entre las variables involucradas en el proceso. Para la construcción de dicha tabla es preciso identificar los elementos que se listan a continuación.

Variable de estado	Tiempo en el sistema de inspección (7)
Entidades	Piezas
Eventos	Tiempo de llegada (2) Fin de la inspección (5)
Evento secundario	Inicio de la inspección (3)
Actividades	Tiempo entre llegadas (1) Tiempo de inspección (4)

Los números entre paréntesis indican la columna que ocupa cada elemento en las tablas 4.1 y 4.2.

2. Defina las relaciones lógico-matemáticas entre los elementos; en la tabla 4.1 se describen, por ejemplo, las siguientes relaciones:
 - a) El *tiempo entre llegadas* es una variable aleatoria, que se simula por medio del generador RAND() o ALEATORIO() de la hoja de cálculo de Excel y la función generadora de variables exponenciales $E_i = -5 \ln(1 - r_i)$.
 - b) El evento *tiempo de llegada* de la pieza corresponde al valor acumulado de la columna (1).
 - c) Si se toma en cuenta que solamente existe un operario encargado de la tarea, el *inicio de la inspección* puede ocurrir cuando la pieza entra al sistema, esto, en caso de que el operario esté ocioso (2), o bien cuando termina de inspeccionar la pieza anterior (5).
 - d) El *tiempo de inspección* es una variable aleatoria normal con media 4 y desviación estándar 0.5, generada mediante la función interna normal acumulada inversa (NORMINV O DISTNORMINV) y como probabilidad el generador de números aleatorios RAND() o ALEATORIO().

- e) El fin de la inspección se calcula sumando el tiempo de inspección (4) al tiempo de inicio de la inspección (3).
- f) La variable *Tiempo en inspección* se calcula, finalmente, como la diferencia entre el tiempo de llegada (2) y el fin de la inspección (5).
- g) Si bien no forma parte del objetivo del ejemplo, también es posible determinar el *tiempo de espera* de una pieza antes de ser inspeccionada, ya que es igual a la diferencia entre el tiempo de inicio de inspección (3) y el tiempo de llegada de la pieza (2).
- h) Esta última columna permite calcular el tiempo promedio de inspección como promedio móvil: cada vez que una nueva pieza es simulada, el tiempo promedio de inspección se recalcula.

Tabla 4.1 Relación entre los eventos y actividades involucradas en el proceso (ejemplo 4.8).

	C	D	E	F	G	H	I	J	K
	Pieza	Tiempo entre llegadas (1)	Tiempo de llegada (2)	Inicio de la inspección (3)	Tiempo de inspección (4)	Fin de la inspección (5)	Tiempo en inspección (6)	Tiempo en espera (7)	Tiempo promedio en inspección (8)
5	1	=-5*LN(1-RAND())	=D5	=E5	=NORMINV(RAND(),4,0.5)	=F5+G5	=H5-E5	=F5-E5	=AVERAGE(\$I\$5:I5)
6	2	=-5*LN(1-RAND())	=D6+E5	=MAX(E6,H5)	=NORMINV(RAND(),4,0.5)	=F6+G6	=H6-E6	=F6-E6	=AVERAGE(\$I\$5:I6)
7	3	=-5*LN(1-RAND())	=D7+E6	=MAX(E7,H6)	=NORMINV(RAND(),4,0.5)	=F7+G7	=H7-E7	=F7-E7	=AVERAGE(\$I\$5:I7)
8	4	=-5*LN(1-RAND())	=D8+E7	=MAX(E8,H7)	=NORMINV(RAND(),4,0.5)	=F8+G8	=H8-E8	=F8-E8	=AVERAGE(\$I\$5:I8)

- Una vez definidas las relaciones se simula el proceso, al hacerlo, se debe cuidar que el tamaño de la réplica o experimento sea lo suficientemente grande para asegurar la estabilidad del resultado final. La réplica cuyos resultados se ilustran en la tabla 4.2 se realizó con 1500 piezas; la información nos indica que el tiempo promedio de espera es de 15.05 minutos/pieza. Además de este resultado, la columna 8 permite visualizar la estabilización del sistema mediante una gráfica de líneas.

Tabla 4.2 Simulación del proceso de inspección (en una hoja de cálculo de Excel).

Pieza	Tiempo entre llegadas (1)	Tiempo de llegada (2)	Inicio de la inspección (3)	Tiempo de inspección (4)	Fin de la inspección (5)	Tiempo en inspección (6)	Tiempo en espera (7)	Tiempo promedio en inspección (8)
1	4.82	4.82	4.82	4.03	8.84	4.03	0.00	4.03
2	2.77	7.59	8.84	4.77	13.61	6.02	1.25	5.02
3	3.72	11.32	13.61	3.97	17.59	6.27	2.30	5.44
4	3.39	14.71	17.59	3.71	21.30	6.59	2.88	5.73
5	2.59	17.30	21.30	3.10	24.40	7.10	4.00	6.00

6	0.19	17.49	24.40	2.64	27.04	9.56	6.91	6.59
7	1.85	19.33	27.04	4.34	31.38	12.05	7.71	7.37
8	0.93	20.26	31.38	4.16	35.54	15.28	11.12	8.36
9	4.65	24.92	35.54	4.41	39.95	15.03	10.63	9.10
10	6.02	30.94	39.95	4.14	44.09	13.15	9.01	9.51
...
1495	0.04	7380.97	7390.99	3.68	7394.66	13.70	10.02	15.02
1496	1.46	7382.42	7394.66	4.45	7399.11	16.69	12.24	15.02
1497	1.56	7383.99	7399.11	4.32	7403.43	19.45	15.13	15.03
1498	1.98	7385.97	7403.43	3.66	7407.09	21.13	17.47	15.03
1499	0.37	7386.34	7407.09	3.48	7410.57	24.23	20.75	15.04
1500	1.86	7388.20	7410.57	4.57	7415.15	26.94	22.37	15.05

En la figura 4.6 se muestra la gráfica de estabilización que se obtuvo a partir de la columna 8 (Tiempo promedio en inspección). Dicha gráfica nos indica que el tamaño de la réplica es lo suficientemente grande para asegurar la convergencia del resultado. Cabe señalar que esta gráfica de estabilización corresponde a una réplica diferente a la de la tabla de eventos.



Figura 4.6 Periodo de estabilización de tiempo promedio de permanencia en el sistema.

Al trabajar con procesos donde se involucran variables, actividades y eventos aleatorios, las variables de estado o variables de respuesta serán, en consecuencia, aleatorias. La figura 4.7 muestra las gráficas de estabilización de 5 diferentes réplicas del mismo modelo. Si bien la estabilización está asegurada, el resultado final nunca es el mismo; es evidente que replicar el experimento debe ser una práctica común en cualquier simulación.

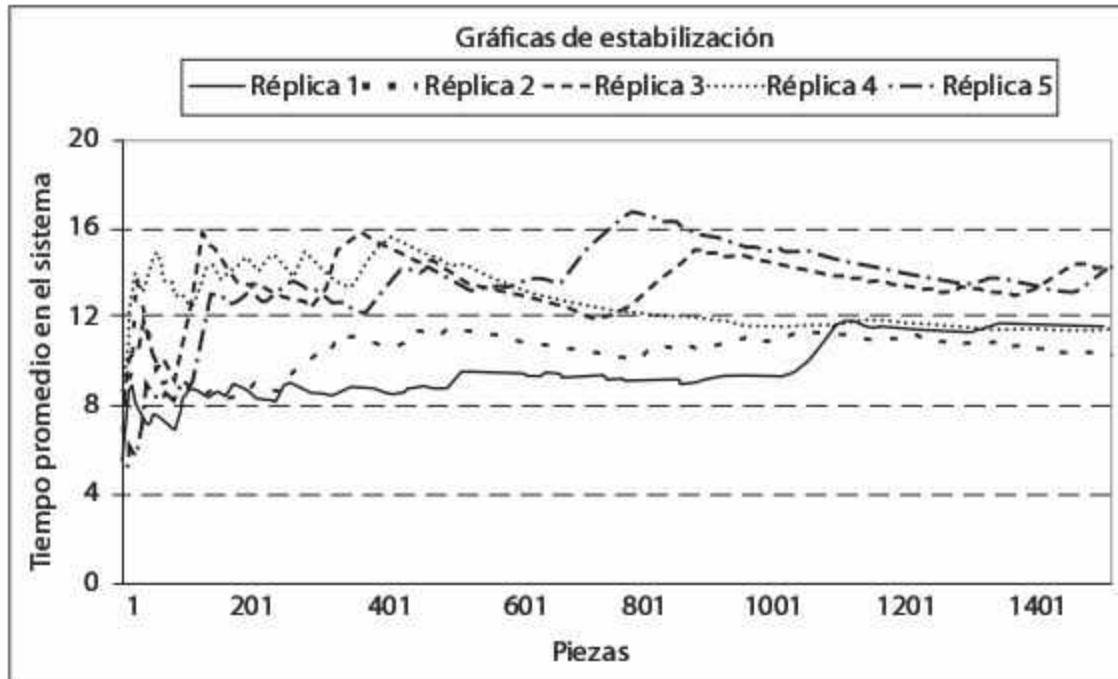


Figura 4.7
Gráficas de estabilización de 5 réplicas independientes.

Al replicar el experimento 50 veces se obtienen los resultados que se listan en la tabla 4.3. Para comprender el comportamiento de la variable es necesario analizar estadísticamente esta información.

Tabla 4.3 Resultados de 50 réplicas del experimento.

10.44	11.53	14.58	11.43	14.27
12.63	15.22	10.66	12.00	10.23
11.06	10.22	17.09	14.62	13.90
12.01	11.30	12.55	12.25	16.34
13.49	12.52	11.55	11.73	15.31
10.57	10.47	10.05	11.85	13.12
10.97	10.41	11.66	10.48	10.59
13.75	11.88	15.09	10.12	9.69
12.97	12.34	11.21	13.17	13.66
12.36	11.63	10.19	11.56	10.59

El análisis estadístico de las réplicas —realizado en este caso con la herramienta Stat:Fit de ProModel— permite concluir, a través de una prueba de bondad de ajuste, que el tiempo promedio de espera en el proceso de inspección sigue una distribución de Erlang con los siguientes parámetros: localización 9, forma 3 y escala 1.06 (vea la figura 4.8); además, tenemos los siguientes estadísticos básicos:

- Media: 12.18 minutos/pieza
- Desviación estándar: 1.76 minutos/pieza
- Intervalo de confianza con $1 - \alpha = .95$: [11.66,12.69] min/pieza

- Valor mínimo en la muestra: 9.69 minutos/pieza
- Valor máximo en la muestra: 17.09 minutos/pieza
- Coeficiente de asimetría (skewness): 0.8177
- Curtosis: -0.071

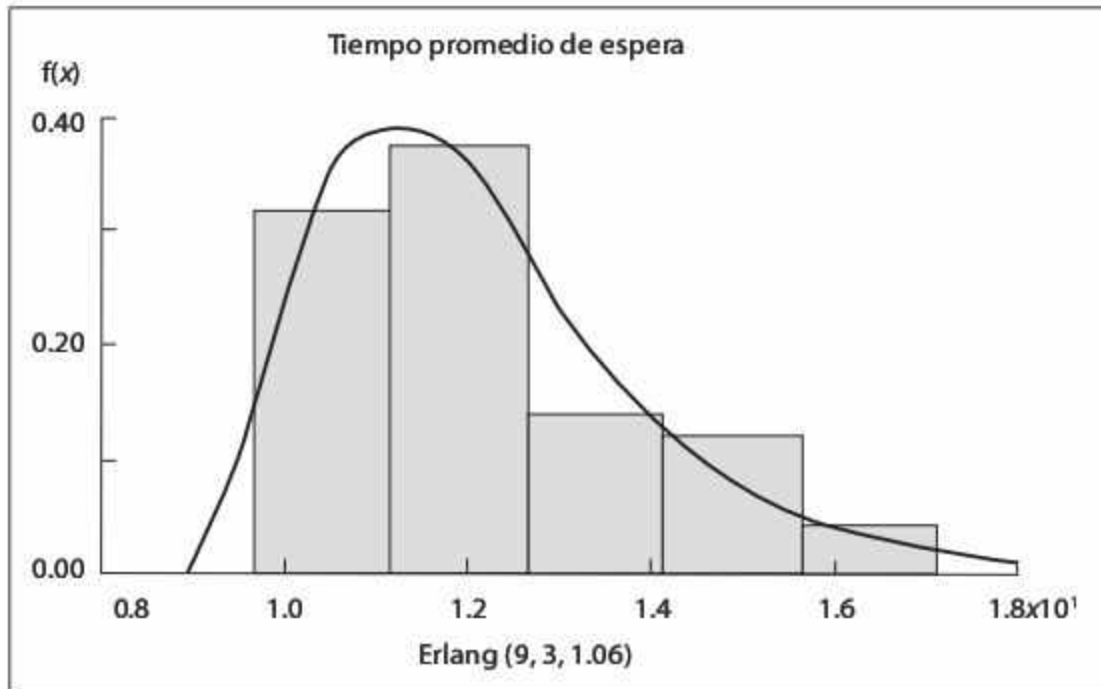


Figura 4.8
Histograma de las réplicas (obtenido con Stat:Fit).

4.3.2 Modelo de un proceso de ensamble e inspección

Ejemplo 4.9

Dos barras metálicas de diferente longitud son unidas mediante un proceso de soldadura para formar una barra de mayor longitud. La longitud del primer tipo de barra sigue una distribución uniforme entre 45 y 55 cm. La longitud del segundo tipo de barra sigue una distribución 4-Erlang con media de 30 cm. Las especificaciones del producto final son de 80 ± 10 cm. Determine el porcentaje de barras fuera de especificación.

Para la solución del ejemplo se requiere:

- Identificación de los elementos:

Variable de estado	Cantidad de barras fuera de especificación
Entidades	Barras
Evento	Comparación contra especificaciones 0: Dentro de especificaciones 1: Fuera de especificaciones
Actividades	Medición de la longitud de la barra 1 Medición de la longitud de la barra 2 Soldadura de las barras 1 y 2

- Construcción de la tabla de eventos:

Tabla 4.4 Relación entre los elementos (ejemplo 4.9).

G	D	E	F	G	H	I	J
Ensamble	Longitud barra 1 [cm] (1)	Longitud barra 2 [cm] (2)	Longitud total [cm] (3)	Ei (4)	Es (5)	Estado de la barra (6)	Probabilidad de estar fuera de especificaciones (7)
1	= (55-45)*RAND()+45	= -(30/4)*LN(RAND())	= D5+E5	70	90	= IF(F5<G5,1,IF(F5>H5,1,0))	= SUM(\$I\$5:I5)/C5
2	= (55-45)*RAND()+45	= -(30/4)*LN(RAND())	= D6+E6	70	90	= IF(F6<G6,1,IF(F6>H6,1,0))	= SUM(\$I\$5:I6)/C6
3	= (55-45)*RAND()+45	= -(30/4)*LN(RAND())	= D7+E7	70	90	= IF(F7<G7,1,IF(F7>H7,1,0))	= SUM(\$I\$5:I7)/C7
4	= (55-45)*RAND()+45	= -(30/4)*LN(RAND())	= D8+E8	70	90	= IF(F8<G8,1,IF(F8>H8,1,0))	= SUM(\$I\$5:I8)/C8

La tabla 4.4 muestra la relación matemática entre las diferentes variables o elementos del sistema; fue desarrollada en una hoja de cálculo y el significado de cada columna es el siguiente:

- (1) La *longitud de la barra 1* es una variable aleatoria con distribución uniforme entre 45 y 55 cm. Fue simulada con el generador RAND() o ALEATORIO() de la hoja de cálculo, y CON la ecuación generadora de variables uniformes $U_i = a + (b - a)r_i$.
- (2) La *longitud de la barra 2* es una variable aleatoria simulada con la función RAND() o ALEATORIO(), y CON la ecuación generadora de eventos Erlang:

$$ER_i = -\frac{30}{4} \ln\left(\prod_{i=1}^4 r_i\right).$$
- (3) *Longitud total*: Esta columna representa el proceso de soldadura, y se obtiene sumando las longitudes de las barras pequeñas de las columnas (1) y (2).
- (4) La variable E_i simula el límite inferior de las especificaciones.
- (5) La variable Es simula el límite superior de las especificaciones.
- (6) Se asigna el atributo de calidad a cada pieza, denominado *Estado de la barra*, mediante la comparación de la longitud total de la barra y los límites de especificación.
- (7) Para determinar la *Probabilidad de estar fuera de especificaciones* se divide el número de piezas defectuosas entre el número de piezas totales. Esto permite obtener la probabilidad como promedio móvil, de manera que cada vez que es simulado un nuevo ensamble la probabilidad se recalcula.

- Simulación de sistema.

En la tabla 4.5 se muestra una réplica con los resultados numéricos de las ecuaciones de la tabla 4.4.

Tabla 4.5 Simulación del proceso (en Excel).

Ensamble	Longitud barra 1 [cm] (1)	Longitud barra 2 [cm] (2)	Longitud total [cm] (3)	Ei (4)	Es (5)	Estado de la barra (6)	Probabilidad (7)
1	47.25	22.50	69.75	70.00	90.00	1.00	1.00
2	48.65	43.32	91.96	70.00	90.00	1.00	1.00
3	53.59	16.32	69.92	70.00	90.00	1.00	1.00
4	47.79	35.74	83.53	70.00	90.00	0.00	0.75
5	52.08	25.62	77.70	70.00	90.00	0.00	0.60
6	51.93	37.48	89.41	70.00	90.00	0.00	0.50
7	51.31	19.86	71.17	70.00	90.00	0.00	0.43
8	53.17	39.58	92.75	70.00	90.00	1.00	0.50
9	45.58	32.53	78.11	70.00	90.00	0.00	0.44
10	49.10	25.01	74.12	70.00	90.00	0.00	0.40
...
995	46.83	12.19	59.02	70.00	90.00	1.00	0.50
996	54.31	8.04	62.35	70.00	90.00	1.00	0.50
997	52.36	16.18	68.54	70.00	90.00	1.00	0.50
998	49.32	48.40	97.72	70.00	90.00	1.00	0.50
999	51.30	6.82	58.11	70.00	90.00	1.00	0.50
1000	49.88	38.76	88.64	70.00	90.00	0.00	0.50

A partir de la información de la variable aleatoria *Estado de la barra* (columna 6) y mediante una prueba de bondad de ajuste, es posible demostrar que esa variable sigue una distribución de probabilidad de Bernoulli con media 0.5.

- Construcción de la gráfica de estabilización.

La gráfica de estabilización (vea la figura 4.9) de la información de la *Probabilidad* (columna 7) de la tabla 4.5 permite visualizar que la réplica entra a la zona de estado estable después de 200 ensambles, y se mantiene oscilando alrededor de 0.5 hasta el final de la simulación. Ésta nos permite comprobar visualmente que el experimento tiene las dimensiones suficientes para asegurar la convergencia del resultado.

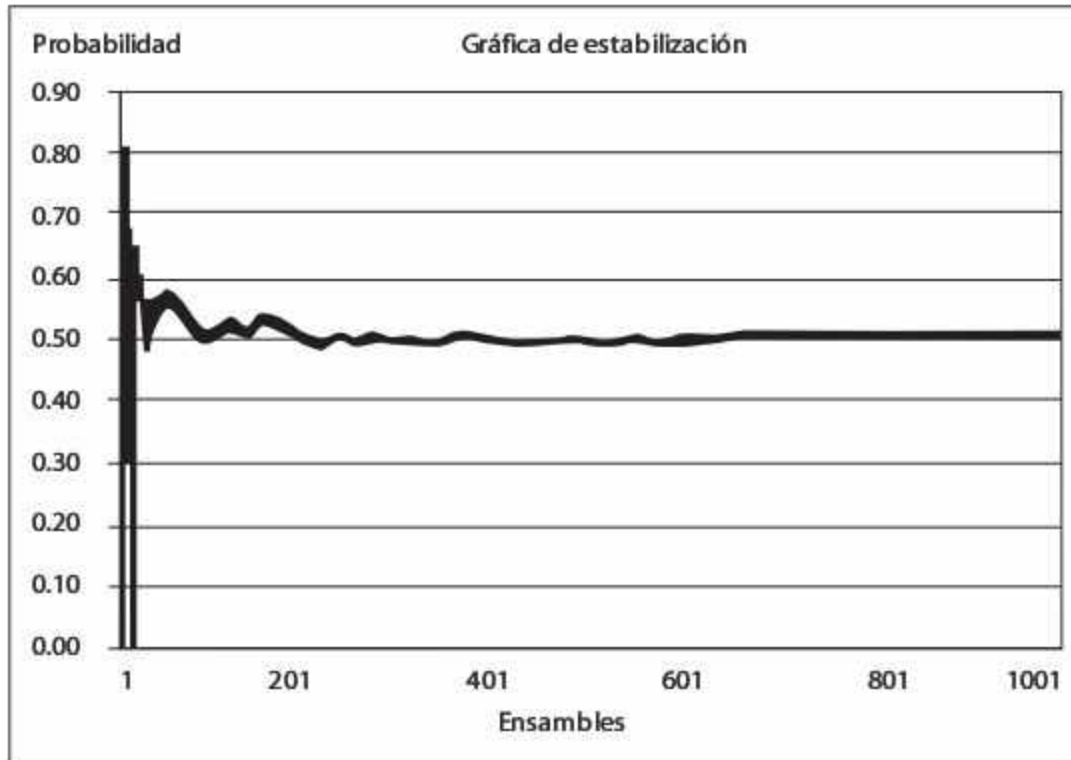


Figura 4.9
Gráfica de estabilización de la probabilidad de que una barra esté fuera de especificaciones.

- Réplicas.

Si se replica el experimento 42 veces, y se modifica sólo la secuencia de números pseudoaleatorios, se obtienen los resultados de la tabla 4.6.

Tabla 4.6 Resultados de 42 réplicas del experimento.

0.50	0.49	0.50	0.51	0.50	0.54
0.52	0.51	0.48	0.49	0.51	0.54
0.50	0.53	0.53	0.51	0.49	0.49
0.51	0.51	0.52	0.51	0.52	0.50
0.51	0.46	0.50	0.52	0.52	0.51
0.52	0.49	0.50	0.50	0.50	0.53
0.54	0.54	0.49	0.50	0.51	0.53

- Análisis estadístico de la variable de estado:

El análisis del resultado de las réplicas de la tabla 4.6 —realizado con ayuda de la herramienta Stat::Fit de ProModel— permite concluir, a través de una prueba de bondad de ajuste, que la *Probabilidad de que un ensamble esté fuera de especificaciones* sigue una distribución de Erlang con estos parámetros: localización, 0, forma 31.5, y escala, 0.517 (vea la figura 4.10).

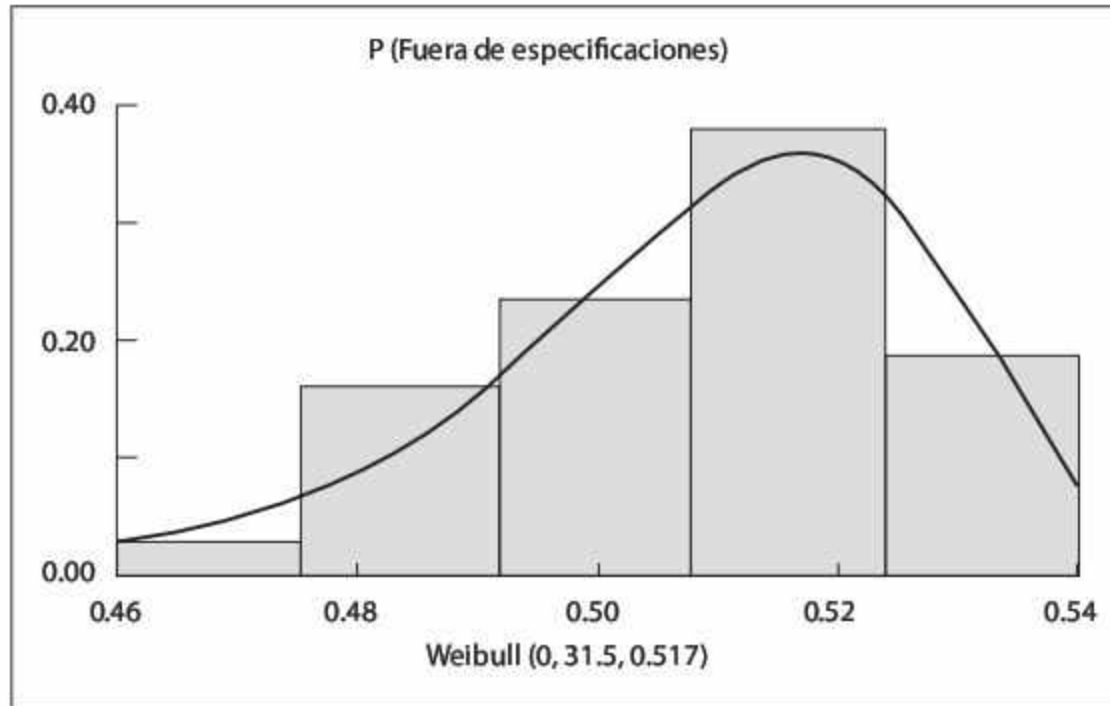


Figura 4.10
Histograma de las réplicas (obtenido con Stat::Fit).

Además, la variable de respuesta tiene los siguientes estadísticos:

- Media: 0.509
- Desviación estándar: 0.0175
- Intervalo de confianza con $1 - \alpha = .95$: [0.504,0.514] min/pieza
- Valor mínimo en la muestra: 0.46
- Valor máximo en la muestra: 0.54
- Coeficiente de asimetría: -0.124
- Curtosis: 0.014

4.3.3 Modelo de un sistema de inventarios

Ejemplo 4.10

La demanda de azúcar en una tienda sigue una distribución exponencial con media de 100 kg/día. El dueño de la tienda revisa el inventario cada 7 días, y hace un pedido a la planta igual a la capacidad de la bodega menos la cantidad de azúcar que tiene disponible en ese momento; la entrega es inmediata. La demanda no surtida por falta de existencias representa ventas perdidas. La capacidad de almacenamiento de la bodega es de 700 kg. El costo de ordenar es de \$1,000/orden. El costo de faltante es de \$6/kg, y el costo de llevar el inventario es de \$1/kg. Determine el comportamiento del inventario a lo largo del tiempo y el costo promedio/día para un horizonte de dos meses.

Para la solución del ejemplo se requiere:

- Identificación de los elementos

Variable de estado	Inventario en el almacén
Entidades	Clientes
Evento	Demanda Ventas Entrega de material por parte del proveedor
Actividades	Cálculo de los costos

- Construcción de la tabla de eventos

Tabla 4.7 Tablas de eventos para el ejemplo 4.10.

9	B	C	D	E	F	G
10	Día	Entregas del proveedor	Inventario inicial	Demanda	Ventas	Inventario final
11	0	700	=C11	=-100*LN(1-RAND())	=IF(D11>=E11,E11,D11)	=MAX(0,\$D11-\$E11)
12	=B11+1	=IF(MOD(B12,7)=0,700-G11,0)	=G11+C12	=-100*LN(1-RAND())	=IF(D12>=E12,E12,D12)	=MAX(0,\$D12-\$E12)
13	=B12+1	=IF(MOD(B13,7)=0,700-G12,0)	=G12+C13	=-100*LN(1-RAND())	=IF(D13>=E13,E13,D13)	=MAX(0,\$D13-\$E13)
14	=B13+1	=IF(MOD(B14,7)=0,700-G13,0)	=G13+C14	=-100*LN(1-RAND())	=IF(D14>=E14,E14,D14)	=MAX(0,\$D14-\$E14)
15	=B14+1	=IF(MOD(B15,7)=0,700-G14,0)	=G14+C15	=-100*LN(1-RAND())	=IF(D15>=E15,E15,D15)	=MAX(0,\$D15-\$E15)

(a) Relación entre los elementos

9	H	I	J	K	L
10	Costo de ordenar	Costo de llevar inventario	Costo de faltante	Costo total	Costo promedio
11	=IF(MOD(B11,7)=0,1000,0)	=1*(D11+G11)/2	=IF(D11<=E11,6*(E11-D11),0)	=SUM(J11:L11)	=AVERAGE(\$M\$11:M11)
12	=IF(MOD(B12,7)=0,1000,0)	=1*(D12+G12)/2	=IF(D12<=E12,6*(E12-D12),0)	=SUM(J12:L12)	=AVERAGE(\$M\$11:M12)
13	=IF(MOD(B13,7)=0,1000,0)	=1*(D13+G13)/2	=IF(D13<=E13,6*(E13-D13),0)	=SUM(J13:L13)	=AVERAGE(\$M\$11:M13)
14	=IF(MOD(B14,7)=0,1000,0)	=1*(D14+G14)/2	=IF(D14<=E14,6*(E14-D14),0)	=SUM(J14:L14)	=AVERAGE(\$M\$11:M14)
15	=IF(MOD(B15,7)=0,1000,0)	=1*(D15+G15)/2	=IF(D15<=E15,6*(E15-D15),0)	=SUM(J15:L15)	=AVERAGE(\$M\$11:M15)

(b) Relación entre los costos

Las tablas 4.7(a) y 4.7(b) muestran la relación matemática entre las diferentes variables o elementos del sistema; la tabla está desarrollada en un programa de hoja de cálculo, y el significado de cada columna es el siguiente:

- **B:** Contador de los *Días* transcurridos.
- **C:** En esta columna se simulan las *Entregas de material*: cada siete días se restablece el inventario en un nivel de 700 kg. Los valores se calculan como la diferencia entre la *Capacidad del almacén* y el *Inventario final* del día anterior. El uso de la función *residuo* o *módulo* (MOD) permite controlar que la entrega se realice cada vez que el *Día* (columna B) sea múltiplo de siete.
- **D:** El *Inventario al inicio* del día se calcula mediante la suma del *Inventario final* del día anterior y las *Entregas de material* por parte del proveedor.
- **E:** La *Demanda* es una variable aleatoria con distribución exponencial y media de 100 kg. Se simula mediante el generador RAND()o ALEATORIO() de la hoja de cálculo y la ecuación generadora $E_i = -\frac{1}{\lambda} \ln(1-r_i)$.

- **F:** Las *Ventas* representan la cantidad que le fue entregada al cliente, y se calcula como el valor mínimo entre el *Inventario al inicio* del día y la *Demanda*.
- **G:** El *Inventario al final* del día se calcula mediante la resta de las *Ventas* (columna F) del *Inventario inicial* del día (columna D), se verifica previamente que no exista faltante.
- **H:** En esta columna la función *residuo* o *módulo* (MOD) permite incrementar en \$1,000 el *Costo de ordenar* cada vez que llegue una orden a la tienda.
- **I:** Se calcula el inventario promedio durante el día, y el resultado se multiplica por \$1/kg.
- **J:** En caso de no cubrir la *Demanda*, el *Costo de faltante* se calcula multiplicando la demanda no surtida en ese día por el costo de faltante por unidad, que en el ejemplo es de \$6/kg.
- **K:** El *Costo total* se determina mediante la suma de las columnas *Costos de inventario*, *Faltante* y *Ordenar*.
- **L:** La forma de calcular esta columna permite tener el *Costo total* como promedio móvil: cada vez que se simula un nuevo día, el costo se recalcula. Con esta columna se analiza la estabilidad de la variable inventario promedio.

- Simulación de sistema.

La tabla 4.8 muestra los resultados, es una réplica de 14 días de la simulación del sistema, utiliza las ecuaciones de las tablas 4.7(a) y 4.7(b).

Tabla 4.8 Tabla de eventos del sistema de inventarios (realizada en Excel).

Día	Entregas del proveedor	Inventario inicial	Demanda	Ventas	Inventario final	Costo de ordenar	Costo de llevar inventario	Costo de faltante	Costo total	Costo promedio
0	700.00	700.00	181.23	181.23	518.77	1000.00	609.39		1609.39	1609.39
1		518.77	29.40	29.40	489.37		504.07		504.07	1056.73
2		489.37	205.65	205.65	283.72		386.55		386.55	833.33
3		283.72	112.36	112.36	171.36		227.54		227.54	681.89
4		171.36	107.42	107.42	63.94		117.65		117.65	569.04
5		63.94	77.26	63.94	0.00		31.97	79.96	111.93	492.85
6		0.00	43.34	0.00	0.00		0.00	260.06	260.06	459.60
7	700.00	700.00	42.02	42.02	657.98	1000.00	678.99		1678.99	612.02
8		657.98	121.36	121.36	536.61		597.30		597.30	610.38
9		536.61	83.89	83.89	452.73		494.67		494.67	598.81
10		452.73	139.21	139.21	313.52		383.12		383.12	579.20
11		313.52	185.00	185.00	128.51		221.01		221.01	549.36
12		128.51	46.62	46.62	81.89		105.20		105.20	515.19
13		81.89	329.79	81.89	0.00		40.94	1487.43	1528.37	587.56
14	700.00	700.00	150.99	150.99	549.01	1000.00	624.51		1624.51	656.69

- Resultados:

La figura 4.11 muestra el comportamiento del inventario al inicio del día (columna C) a lo largo del tiempo, para el periodo simulado de 60 días.

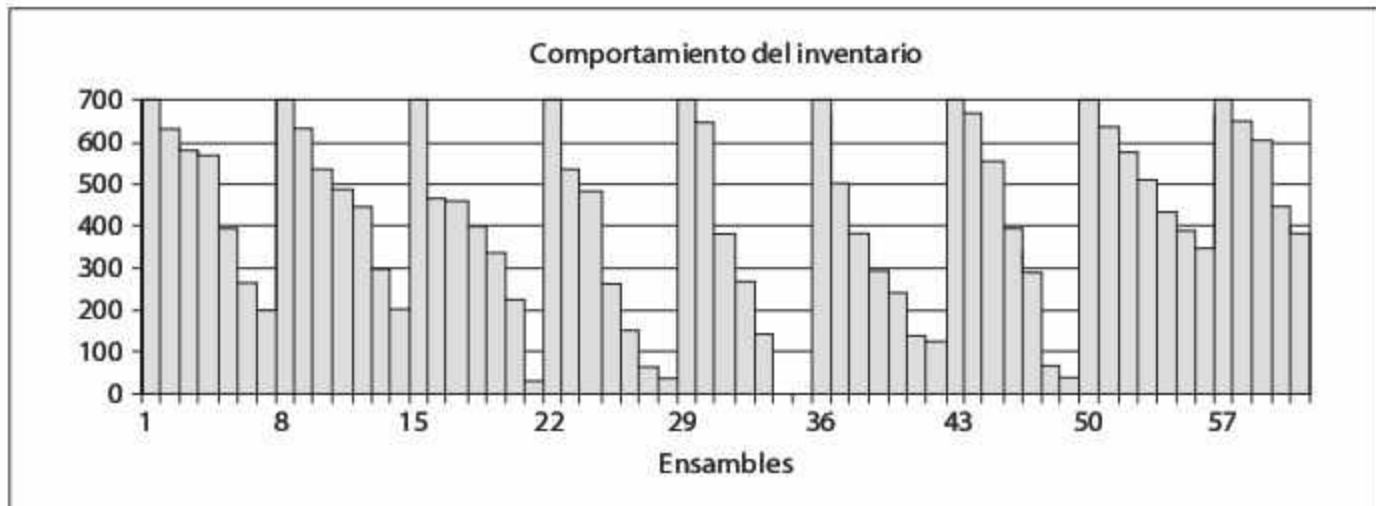


Figura 4.11 Simulación del sistema de inventarios (realizada en Excel).

El análisis del costo promedio de operación de la tienda incluye en primera instancia las gráficas de estado estable (figura 4.12) de cinco réplicas independientes de los resultados de la tabla 4.8. El resultado nos permite observar la convergencia del costo respecto del tiempo.

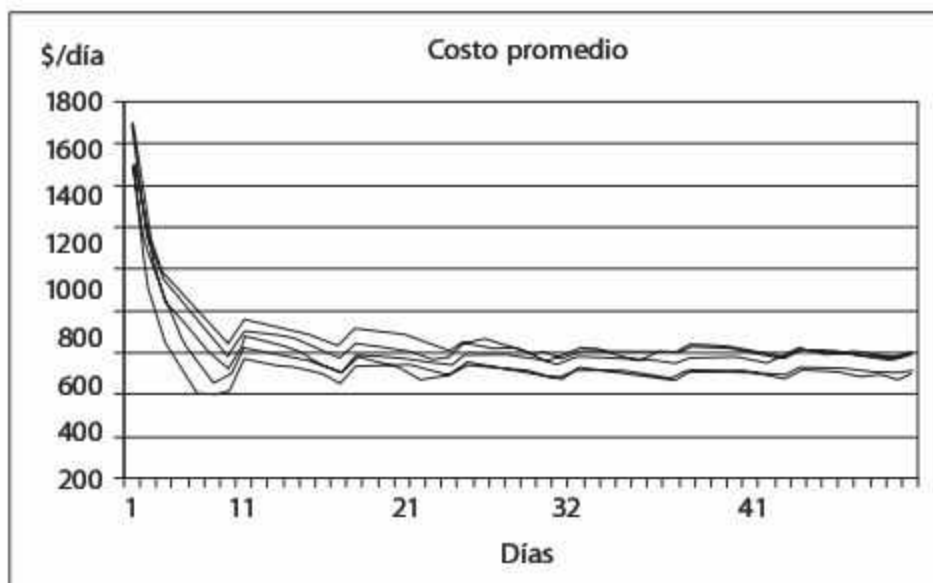


Figura 4.12
Costo promedio por día de cinco réplicas independientes.

Los valores finales del costo de operación de estas cinco réplicas son 592.55, 527.45, 506.13, 605.59 y 597.85. Con esta información calculamos un valor promedio de 565.9 y una desviación estándar de 45.7. Debido a que esta información es insuficiente para

demostrar la normalidad de los datos, el cálculo del intervalo de confianza con un nivel de aceptación de 95% (nivel de rechazo 5%) se realiza mediante el teorema de Tchebycheff:

$$IC = \left[\bar{x} - \frac{s}{\sqrt{r\alpha}} \quad , \quad \bar{x} + \frac{s}{\sqrt{r\alpha}} \right]$$

$$IC = \left[565.9 - \frac{45.7}{\sqrt{(5)(0.05)}} \quad , \quad 565.9 + \frac{45.7}{\sqrt{(5)(0.05)}} \right]$$

$$IC = [474.49 \quad , \quad 657.33] \$/\text{día}$$

4.4 Selección de lenguajes de simulación

En un principio, los programas de simulación se elaboraban mediante algún lenguaje de propósito general, como ASSEMBLER, FORTRAN, ALGOL o LP/1. A partir de la década de 1960 hacen su aparición los lenguajes específicos para simulación que permiten a analistas y programadores desarrollar modelos de una forma más rápida, gracias a módulos estandarizados. En aquella época surgieron lenguajes como GPSS, GASP, SIMSCRIPT, SLAM, SIMAN y SSED. En la última década del siglo pasado la aparición de las interfaces gráficas revolucionaron el campo de las aplicaciones en esta área, y ocasionaron el nacimiento de los simuladores, con los cuales se ha facilitado enormemente la programación de los modelos.

En el terreno práctico, es importante utilizar la aplicación que mejor se adecue al tipo de sistema a simular, ya que de la selección del lenguaje o simulador dependerá el tiempo de desarrollo del modelo de simulación. Las opciones van desde las hojas de cálculo, lenguajes de tipo general (como Visual Basic, C++ o FORTRAN), lenguajes específicos de simulación (como GPSS, SLAM, SIMAN, SIMSCRIPT, GAS y SSED), hasta simuladores específicamente desarrollados para diferentes objetivos (como SIMPROCESS, ProModel, Witness, Taylor II, Crystal Ball, Delmia).

En la actualidad la selección del lenguaje o simulador depende de los siguientes factores:

- (1) Los mercados primarios a los que atenderá la simulación, así como las aplicaciones típicas en que se le utilizará, por ejemplo: administración estratégica, logística, telecomunicaciones, manufactura, sistemas militares, sistemas de salud, manejo de materiales, análisis de riesgo, simulación continua o discreta, etcétera.
- (2) Los requerimientos de equipo, como plataforma o sistema operativo, memoria RAM y utilización de disco duro.
- (3) La capacidad de construcción y programación del modelo a través de iconos o mediante procesos de tipo "arrastrar y colocar" (*drag and drop*), así como acceso a programación estándar. A este respecto también es importante considerar el

tiempo y la velocidad en la detección de errores, así como la posibilidad de reutilizar partes de código, objetos o plantillas (*templates*).

- (4) La inclusión de herramientas complementarias para la realización de pruebas de bondad de ajuste en forma automática, el análisis de las variables de respuesta, la posibilidad de crear diseño de experimentos, y la optimización del sistema simulado.
- (5) La animación del sistema, considerando aspectos como velocidad, uso de diferentes vistas, facilidad de exportación, compatibilidad con otras aplicaciones, y la posibilidad de poder prescindir del uso de la animación.
- (6) El costo y el tipo de licencia otorgada, así como el soporte técnico y la facilidad de entrenamiento y uso de manuales y ayudas en línea.
- (7) Otras consideraciones, como la capacidad de empaquetamiento de los modelos, la distribución a otros usuarios, y la capacidad que tenga la compañía para actualizar su producto.

Algunas aplicaciones en el área de simulación disponibles actualmente en el mercado son:

aGPSS	Analytica 4.4	Arena Simulation Software	Bluesss Simulation System
Capacity Planning Simulator	CSIM 20	Enrmgmsuite	Enterprise Dynamics
ExtendSim	Flexsim	GoldSim	LABAMS
MAST	MedModel Optimization Suite	Micro Saint Sharp	Oracle Crystal Ball Suite
Patient Flow Simulator	Portfolio Simulator	Process Simulator	Project Simulator
ProModel Optimization Suite	SAIL	SAS Simulation Studio	ServiceModel Optimization Suite
Simcad Pro	Simio	SIMPROCESS	SIMSCRIPT III
SimTrack	SIMUL8	SLIM	Vanguard Business Analytics

4.5 Caso de estudio 1

La compañía PELICRE se dedica a la fabricación de shampoo para el crecimiento de cabello. La empresa cuenta con 2 líneas de producción donde se llenan, tapan y etiquetan los frascos.

El índice de producción anual es de cinco millones de frascos en la línea 1 y tres millones de frascos en la línea 2.

La empresa ha padecido un largo historial de dificultades en cuanto a la colocación de las tapas. Cuando ocurre una de estas fallas se debe detener la línea hasta que se corrija el imperfecto. Se ha descubierto que la distribución del tiempo entre fallas es el siguiente:

Tabla 4.9 Tiempo entre fallas de la línea de producción 1 (años/falla).

0.002603	0.039543	0.007795	0.002774	0.019015
0.007731	0.020012	0.000224	0.001501	0.001067
0.001784	0.006437	0.011479	0.050093	0.000667
0.006141	0.015217	0.001125	0.003704	0.003504
0.002996	0.011127	0.002341	0.005903	0.005369
0.0022	0.000258	0.009986	0.012724	0.014449
0.002533	0.013669	0.005693	0.013389	0.013057
0.014287	0.009593	0.019847	0.003017	0.006602
0.027379	0.01309	0.01223	0.002126	0.008752
0.010181	0.015025	0.007304	0.01711	0.049539

Tabla 4.10 Tiempo entre fallas de la línea de producción 2 (años/falla).

0.004286	0.004294	0.005361	0.008621	0.00891
0.008475	0.006741	0.003123	0.008298	0.011369
0.00732	0.008715	0.005589	0.012689	0.006035
0.009587	0.008716	0.003464	0.00747	0.006335
0.008301	0.009675	0.005695	0.009542	0.007926
0.006663	0.008711	0.008972	0.007954	0.012864
0.008863	0.008215	0.004428	0.008506	0.007099
0.007903	0.009177	0.007846	0.006076	0.008713
0.009643	0.007276	0.011721	0.009558	0.008087
0.00954	0.006018	0.007876	0.009368	0.008413

El tiempo de reparación sigue una distribución exponencial con una media de $(0.001/n)$ años, donde n denota el número de trabajadores en el equipo de reparaciones. Cada vez que ocurre una falla se destruyen cierta cantidad de frascos, lo que equivale a un costo de \$ 7.0 cada uno. Cada frasco se vende a \$ 8.0.

Un muestreo sobre el número de frascos que se destruyen cuando ocurre una falla arroja la siguiente información.

Tabla 4.11 Frascos rotos/falla.

8	10	12	10	9
10	17	10	11	14
13	10	7	8	8
8	10	10	13	8
7	7	7	7	10
14	12	10	12	11
11	15	7	8	12
6	8	15	14	9
10	7	12	19	8
6	10	4	4	11

El costo anual por trabajador es de \$50,000/año.

- a) La empresa desea determinar el tamaño óptimo del equipo de reparaciones, esto es, el número de trabajadores que deben estar asignados para la reparación de las líneas de producción con la finalidad de obtener la mayor utilidad posible.
- b) La solución obtenida en el inciso (a) seguirá siendo óptima si se cumple cuál de las siguientes opciones:
 1. El tiempo promedio entre fallas de la línea 1 disminuye en un 80%
 2. El tiempo promedio entre fallas de la línea 1 aumenta en un 100%
 3. El tiempo promedio entre fallas de la línea 2 disminuye en un 70%
 4. El tiempo promedio entre fallas de la línea 2 aumenta en un 80%
 5. El número de botellas rotas por falla aumenta en un 200%

4.6 Caso de estudio 2

La empresa DRE produce piezas electrónicas y las ensambla en una línea semiautomática. Una de las piezas es producida en lotes de tamaño $N = 10000$ unidades por semana. Al final de cada semana, se inspecciona una muestra aleatoria de $n = 25$ unidades para determinar si la producción de la semana cumple con los estándares de calidad. Si como resultado de la inspección, el número de piezas defectuosas obtenidas " d " es menor o igual a un valor de rechazo " c ", el lote será aceptado y enviado a la línea de ensamble y por cada pieza defectuosa que entre a la línea de ensamble, ocurrirá un falla que detendrá la línea de producción y ocasionará un costo de \$ 25/falla. Por otro lado, si el número de piezas defectuosas " d " es mayor que el valor de rechazo " c ", el lote será rechazado y enviado a desperdicio, en este caso, para que la línea de ensamble no se detenga, el lote deberá ser repuesto con un lote libre de defectos comprado a un proveedor a un costo de \$30,000. El costo de producción de cada pieza es \$ 2. Las unidades inspeccionadas se destruyen y el costo de mano de obra y equipo usado en la inspección es \$ 0.5. Por lo tanto, el costo total de las piezas inspeccionadas es de \$ 2.50.

La siguiente tabla muestra datos históricos de la fracción de piezas defectuosas en 100 lotes.

0.068	0.012	0.145	0.027	0.034	0.146	0.007	0.041	0.100	0.003
0.003	0.081	0.061	0.093	0.012	0.024	0.005	0.011	0.022	0.069
0.076	0.025	0.032	0.057	0.123	0.046	0.129	0.139	0.002	0.013
0.158	0.012	0.016	0.072	0.040	0.029	0.036	0.008	0.013	0.069
0.025	0.001	0.014	0.006	0.021	0.016	0.054	0.036	0.007	0.001
0.005	0.072	0.018	0.106	0.078	0.049	0.007	0.012	0.026	0.115
0.055	0.130	0.075	0.029	0.043	0.038	0.006	0.068	0.061	0.008
0.033	0.109	0.036	0.011	0.006	0.055	0.002	0.061	0.047	0.011
0.004	0.120	0.044	0.001	0.047	0.027	0.213	0.012	0.007	0.001
0.098	0.063	0.050	0.016	0.018	0.036	0.090	0.006	0.118	0.040

- Desarrolle un modelo de simulación en una hoja de cálculo para el valor óptimo de determinar el valor de rechazo “ c ” para este plan de muestreo simple, que asegure el mínimo costo.
- Si la fracción de piezas defectuosas de los lotes fuera tres veces más grande que los datos históricos, ¿cambiaría el valor de “ c ”?

4.7 Problemas

- En un restaurante de comida rápida se venden hamburguesas a \$6 cada una, con un costo de producción por unidad de \$3.5. Después de un estudio se encontró que la demanda por hora en este local se distribuye de acuerdo con la siguiente función de probabilidad:

Demanda	0	1	2	3	4	5	6
Probabilidades	0.10	0.15	0.25	0.20	0.15	0.08	0.07

Simule la utilidad promedio por hora que se obtendría en 100 horas de trabajo. Realice 5 corridas y construya la gráfica de estabilización de la utilidad promedio para cada corrida, incluyendo su respectivo intervalo de confianza a 95%. ¿Considera que las conclusiones obtenidas son estadísticamente válidas? ¿Por qué? ¿Cuál es la diferencia de concluir mediante los intervalos de confianza de cada réplica y emplear el intervalo de confianza global para las 5 réplicas?

- Después de realizar una simulación de 5 réplicas se obtuvieron los siguientes valores en estado estable para el nivel de ingresos promedio mensual de una compañía: 1236, 1324, 1289, 1302 y 1265. Determine el intervalo de confianza para establecer el verdadero valor del nivel de ingresos promedio mensual de la compañía.

3. Un modelo simula el número de botellas rotas por año en una línea de producción. Los resultados de 6 años de esta variable son: 11540, 10870, 12520, 13750, 10550 y 9850. No se tiene la certeza que el resultado de esta variable siga una distribución Normal.
 - a) Calcule la exactitud actual del modelo con un nivel de aceptación del 95%.
 - b) Determine el número de años que es necesario simular para obtener una exactitud en el resultado de ± 300 botellas rotas con un nivel de aceptación del 90%.

4. Una simulación predice el precio por barril de petróleo a nivel mundial para finales de 2015 en función de ciertos parámetros macroeconómicos que tienen variabilidad. Se realizaron 5 réplicas de 1 año cada una y el precio al final en cada una de las 5 réplicas fue: 125.50, 132.75, 120.80, 138.20 y 127.50 dólares por barril. Suponga normalidad en esta variable para lo siguiente:
 - a) Determinar la exactitud lograda con este número de réplicas y con un nivel de aceptación del 95%.
 - b) Calcular el número de réplicas que se deben realizar para lograr una exactitud de ± 0.35 con un nivel de aceptación de 90%.

5. Se desea conocer el número de productos a simular en un modelo de llenado de botes de mermelada para lograr una exactitud del volumen promedio de llenado de ± 10 mililitros con un nivel de aceptación de 95%. Se realizaron observaciones del volumen (en ml) de 10 botes obteniendo los siguientes resultados: 556, 557, 572, 561, 559, 558, 552, 558, 560 y 558.

6. El tiempo de reparación de un avión se comporta normalmente con media de 5 días y desviación estándar de 1 día. ¿Cuántas reparaciones se tendrían que realizar para que el resultado promedio del tiempo tuviera una exactitud de ± 0.2 días con un nivel de aceptación de 95%?

7. Determine el número de cajas de cereal que es necesario simular en un proceso de llenado para que la exactitud del peso promedio de las cajas no difiera en más de $\pm 0.33\sigma$ con un nivel de aceptación de 98%, considere que la máquina de llenado introduce hojuelas en cada caja con una distribución de weibull.

8. A un operario le llegan ciertas piezas para que las inspeccione; la revisión se desarrolla de acuerdo con la distribución de tiempo $t = 3r_i^2$. Si el operario recibe un lote de 10 piezas, simule cuánto tiempo tardará en revisar el lote. Utilice los siguientes números aleatorios: 0.6251, 0.5948, 0.6674, 0.2807, 0.9359, 0.1655, 0.1189, 0.7857, 0.4783, 0.9987.

Simule ahora 100 lotes mediante una hoja de cálculo y el generador de números pseudoaleatorios MINSTD y determine el tiempo de promedio de revisión por lote.

9. Se tiene un proceso de fabricación de refrigeradores. La demanda diaria de este producto está distribuida de manera normal. La demanda promedio es de 80 refrigeradores por día, con una desviación estándar de 10 refrigeradores diarios. Se desea

- saber cuál es la mejor política de producción, considere 60, 70, 80, 90 y 100 refrigeradores por día. El costo por faltante es de \$8/refrigerador por día, y el costo de tener un refrigerador en el inventario es de \$5/refrigerador por día.
- Se le pide realizar 5 corridas de 100 días para cada política.
 - Obtenga el costo promedio por día de cada política, y un intervalo de confianza para ese promedio diario.
 - Determine, con base en sus resultados, cuál de las políticas seleccionadas es la que debe implementar la empresa.
- Un cilindro con diámetro x_1 será insertado en un agujero con diámetro x_2 . Si x_1 sigue una distribución normal con media de 1.5 cm y varianza de 0.0016, y x_2 , una distribución 2-Erlang y una media de 2.5 cm, simule en una hoja de cálculo la inserción de 500 cilindros y determine mediante el estimador la probabilidad de que haya interferencia (es decir, que cilindro pequeño no entre en el agujero).
 - Una barra de longitud x_1 será unida mediante soldadura a otra de longitud x_2 . Si x_1 sigue una distribución normal con media de 30 cm y varianza de 0.81, y x_2 , una distribución Erlang con $k = 10$ y una media de 14.5 cm, simule la soldadura de 300 barras, tome en cuenta que las especificaciones de diseño son de 45 ± 5 cm. y determine el estimador de la probabilidad de que una barra esté fuera de especificaciones. También calcule los coeficientes de capacidad C_p y C_{p_k} de este proceso. ¿Considera que está bajo control?
 - Un proceso consta de 2 etapas: la primera tiene una duración de t_1 minutos y la segunda dura t_2 minutos. t_1 sigue una distribución normal con media de 30 min y varianza de 10 min, y t_2 , una distribución 3-Erlang y una media de 20 min, el tiempo máximo de producción permitido de este proceso es de 55 min, simule en una hoja de cálculo la producción de 1000 piezas y estime la probabilidad de que una pieza consuma más tiempo del permitido.
 - Un tirador de flecha se encuentra entrenando para las próximas olimpiadas. El blanco a que le dispara consiste en un cuadrado de 10 cm \times 10 cm. Un equipo de investigadores midió el comportamiento histórico de su pulso, y llegaron a la conclusión de que la desviación de cada disparo respecto del centro es normal ($\sigma = 3$) cm en el eje "y", y uniforme $(-12, +12)$ en el eje "x". Simule en una hoja de cálculo 500 disparos (por réplica) del tirador y calcule:
 - La probabilidad de dar en el blanco simulando sólo una réplica.
 - El intervalo de confianza a un nivel de 90% de la distancia entre el punto donde pegó el disparo y el centro del blanco, usando 10 réplicas.
 - En el cierre de la novena entrada del 7º juego del Clásico de Otoño en el Yankee Stadium, Kevin Brown, lanzador de los Padres de San Diego, se enfrenta a Chili Davis, bateador del equipo contrario. Los Padres de San Diego mantienen una mínima ventaja de 1 a 0. Con casa llena, 2 outs y 1 bola en la cuenta del bateador, Davis recibe la señal de esperar. El árbitro ha estado manteniendo una zona de strike de

40 cm \times 40 cm. Kevin Brown lanza la pelota respecto del centro de la zona de strike con una desviación normal ($\sigma = 10$) cm en el eje "y", y uniforme ($-30, +30$) en el eje "x". Simule, en una hoja de cálculo, hasta que el bateador quede eliminado, y calcule lo siguiente:

- a) El porcentaje de strikes al simular sólo una réplica.
- b) El intervalo de confianza a un nivel de 95% de la distancia entre el punto por donde pasó el lanzamiento y el centro de la zona de strike, use 100 réplicas.

Nota: Con 3 strikes (strike = lanzamiento en la zona de 40 \times 40) acumulados, el bateador pierde; con 4 bolas (bola = lanzamiento fuera de la zona) el bateador gana.

15. La llegada de clientes a un banco con 2 cajeras y una fila tiene una distribución de Poisson con media de 40 personas/hr. En una hoja de cálculo simule este proceso durante 8 hrs y determine el tiempo promedio en el sistema, tome en cuenta que el proceso de servicio es exponencial con media 4.4 minutos/cliente.
16. La llegada de clientes a un banco con 1 cajera y una fila tiene una distribución de Poisson con media de 15 personas/hr. El 40% de los clientes hacen 1 transacción, el 30%, 2 transacciones, el 20%, 3 transacciones y 10% el restante, 4 transacciones. El proceso de servicio es exponencial con media 2.2 minutos/transacción. Simule en una hoja de cálculo este proceso durante 8 hrs y replique durante 300 días para determinar el tiempo promedio en la fila.
17. En una fábrica las piezas pasan por un proceso de inspección donde un operario las revisa, tardan 6 ± 2 minutos por pieza. El porcentaje de rechazos que se tiene es de 15%, y las piezas defectuosas son eliminadas. Suponga que el inspector siempre tiene piezas disponibles para revisar, simule este sistema durante 100 piezas.
 - a) Realice 5 corridas y determine, por medio de un intervalo de confianza, el valor promedio de piezas defectuosas que se generarán en el sistema.
 - b) ¿Considera que los resultados obtenidos de la simulación son confiables? ¿Por qué?
18. Un centro de maquinado recibe diversas piezas para ser procesadas. Cada una se trabaja bajo los siguientes tiempos: el 30% tarda 2 minutos con distribución exponencial, el 35% tarda 3 ± 1 min; el 20% tarda 4 minutos de manera constante, el 15% se distribuye de acuerdo con una distribución normal con media de 5 min, y con una desviación estándar de 1 min. Por otra parte, el 5% del total de las piezas maquinadas son retiradas como producto no conforme, y enviadas al área de reproceso.
 - a) Simule el sistema hasta obtener 100 piezas buenas.
 - b) Realice 5 replicas y calcule un intervalo de confianza para el tiempo necesario para completar las 100 piezas.
 - c) ¿Considera que los resultados obtenidos en el inciso c tienen validez? En caso de que no los considere válidos, ¿qué sugiere para mejorar sus conclusiones?

19. Un camión de reparto tarda 30 ± 10 minutos en ser cargado, 20 ± 5 minutos en ser descargado, y 40 minutos con distribución exponencial en trasladarse, ya sea de su base al lugar de entrega, o del lugar de entrega a su base.
 - a) Simule el sistema por 10 horas y realice 5 réplicas.
 - b) Calcule un intervalo de confianza para el número de viajes que se pueden hacer en un día.
 - c) Sólo hay espacio para cargar un camión a la vez. Si la empresa necesita realizar al menos 10 entregas por día, ¿qué recomendaciones daría para lograrlo? Justifique su respuesta y establezca sus supuestos.

20. Una estación de gasolina que cuenta con una sola bomba recibe 10 clientes por hora con distribución exponencial. Estos clientes son atendidos por el operador de la bomba que les da el servicio y les cobra. El tiempo de servicio se distribuye exponencialmente con media de 4 minutos por cliente.
 - a) Determine el número promedio de clientes en el sistema.
 - b) Determine el porcentaje del tiempo que el operador está ocupado.
 - c) Determine el tiempo promedio de permanencia en la fila.

21. En un proceso de control de calidad se pasan cajas de manera constante, con el fin de inspeccionar al azar cierto número de productos de una caja seleccionada también arbitrariamente. La probabilidad de seleccionar una caja para inspección es de 30%; de las cajas que se revisan, en el 50% de los casos se inspecciona sólo un producto, en el 30%, 2 productos, y en el 20% restante, 3 productos. Se sabe que la probabilidad de que una caja contenga uno o más productos defectuosos es de 2%, y que la probabilidad (en porcentaje) de que este producto sea encontrado durante la inspección es de $10 \times$ número de productos inspeccionados.
 - a) Simule 100 cajas que pasan por el proceso de control de calidad.
 - b) Determine el número de cajas que contendrán productos defectuosos.
 - c) Determine cuántas cajas con productos defectuosos no fueron detectadas. Si el costo de una caja con productos defectuosos que sale al mercado es de \$20/caja, determine el valor total en el que se incurriría.

22. Una empresa tiene asignado un camión para el transporte de sacos de harina. El camión realiza 10 viajes diarios y en cada viaje transporta toda la cantidad de sacos que su capacidad le permita. La capacidad del camión es de 1 tonelada y el peso de los sacos sigue una distribución de probabilidad de weibull con parámetro de forma 2, parámetro de escala 40 kg y parámetro de localización 210 kg. Después de simular 2 meses, determine el número promedio de sacos transportados por día y su desviación estándar.

23. Un granjero tiene una gallina que pone huevos a una razón Poisson con media de 2 huevos/día. El 20% de los huevos se rompen, del 30% de ellos nacen pollos y el resto permanecen como huevos. De los pollos el 20% muere y el 80% sobreviven. Simule este sistema durante 300 días y determine el ingreso promedio del granjero si cada huevo lo vende en \$2 y cada pollo en \$30.

24. El dueño de la única funeraria de un pequeño pueblo ha comprado un lote de 500 ataúdes con una longitud de 1.70 metros. La población de este pueblo tiene una estatura normal con una media de 1.65 y desviación estándar de 0.05 metros. El costo de cada ataúd es de \$3,000 y calcula vender cada uno en \$8,000. Al morir, el difunto es llevado a la funeraria para los preparativos correspondientes y en algunas ocasiones será necesario hacer arreglos adicionales para que quepa en el ataúd; como doblarle las rodillas, el cuello, o ambos, hacer un orificio en la base o en la tapa del ataúd, alargar el ataúd o simplemente llevarlo con las piernas colgando. Independientemente de la solución que más agrade a los deudos, el dueño ofrece un descuento de 5% sobre el precio regular. Simule este proceso hasta que se agoten las existencias de ataúdes y determine:
- La cantidad de difuntos que obtendrán un descuento.
 - La utilidad promedio que obtendrá la funeraria por este concepto.
25. Una tienda vende tres tipos de refrigeradores. El 25% de los clientes compra refrigeradores económicos a un precio de \$8,000/refrigerador, el 45% compra refrigeradores estándar a un precio de \$16,000/refrigerador y el resto compra refrigeradores de última generación a un precio de \$30,000/refrigerador. Simule la venta de 1000 refrigeradores y determine: el promedio, la desviación estándar, el error estándar, la curtosis y el coeficiente de asimetría del ingreso por día de la tienda.
26. Una empresa fabricante de agua desea abrir una nueva sucursal. El clima existente es muy variable, vea la siguiente tabla:

Clima	Caluroso	Templado	Frío	Helado
Probabilidad	0.50	0.25	0.15	0.10

La venta de agua se comporta diferente para cada clima de acuerdo con las siguientes distribuciones de probabilidad:

Caluroso				
Ventas (lt/día)	50	100	200	300
Probabilidad	0.10	0.30	0.40	0.20

Templado				
Ventas (lt/día)	40	50	100	200
Probabilidad	0.10	0.20	0.40	0.30

Frío				
Ventas (lt/día)	10	20	50	100
Probabilidad	0.05	0.70	0.20	0.05

Helado				
Ventas (lt/día)	0	5	10	20
Probabilidad	0.10	0.20	0.60	0.10

Si la venta por litro es de \$50 y los costos de producción son de \$1,000/día, simule y determine los siguientes estadísticos para la variable utilidad diaria:

- Promedio, mediana, moda, valor mínimo y valor máximo.
 - Desviación estándar, error estándar y rango.
 - Curtosis y coeficiente de asimetría.
 - Cuartil 25% y 75% e intervalos de confianza del 90 y 95%.
 - Histograma y distribución de probabilidad.
 - La probabilidad de que la utilidad sea mayor a 0.
27. Una empresa ha decidido lanzar a la venta por única ocasión, 1000 lavadoras de un modelo especial. La cantidad de lavadoras a fabricar serán de $1000 + x$ donde x es la cantidad de lavadoras que se mantendrán en inventario para reponer aquellas que se descompongan antes de la fecha límite de garantía. La vida de una lavadora sigue una función de densidad 3-Erlang con media de 3 años. La garantía indica que se repondrá cualquier lavadora que se descomponga antes de 1 año. Esta garantía no será válida para las lavadoras de repuesto. Simule el comportamiento de cada una de las 1000 lavadoras y encuentre el valor de x .
28. La venta diaria de un determinado producto se distribuye de acuerdo a la tabla de probabilidades mostrada. El costo de mantener inventario es de \$5/pieza, mientras que el costo del faltante es de \$10/pieza. Se desea evaluar los niveles de producción de 90, 94, 98 y 100. Determine cuál es el nivel de producción que minimiza los costos de operación, utilice 3 réplicas de 100 días cada una.

Demanda	88	90	92	94	96	98	100	102
probabilidad	0.05	0.2	0.1	0.15	0.2	0.15	0.1	0.05

29. Una estación de reproceso recibe piezas con problemas de calidad. En esta estación se revisan 4 puntos de calidad. La probabilidad de encontrar un defecto (d) es de 0.4 para cada uno de estos puntos de inspección. Las piezas que tengan $d \geq 3$ defectos son desechadas como desperdicio. Las que si se pueden rescatar, son reprocesadas en un tiempo exponencial con media de $1 + 2d$ minutos y regresadas a la línea de fabricación. Use 5 corridas de 100 piezas que si se puedan regresar a la línea, determine:
- El costo del desperdicio si cada pieza cuesta \$13.
 - El tiempo para completar 100 piezas reprocesadas.
30. A un cajero automático llegan clientes con un tiempo entre llegadas con distribución 3-Erlang con media de 6 minutos, cada cliente retira entre \$1,000 y \$5,000 en múltiplos de \$500 con distribución uniforme discreta. El cajero solo tiene disponible \$450,000.

- a) ¿Cuánto tiempo tardará el cajero para que deje de permitir al cliente retirar todo el dinero que deseaba?
- b) ¿Cuántos clientes sí pudieron retirar todo el dinero que deseaban?
- c) Si este cajero es reabastecido de dinero cada 24 horas, ¿Qué recomendación le haría al Banco?
31. Cada 20 minutos llegan piezas a una estación de reproceso. El número de defectos que una pieza puede tener como máximo es 3. Se sabe que estos defectos siguen una distribución Binomial con media 2.4. El tiempo para realizar las reparaciones correspondientes se distribuye exponencial con $\lambda = 0.2$ piezas por minuto por cada defecto que tenga la pieza. Determine, ¿cuánto tiempo tomará procesar 200 piezas? Use 5 réplicas.
32. Don Cleto vende afuera de un estadio de beisbol dos tipos de productos: agua de limón y emparedados de queso. Don Cleto prepara en casa 15 litros de agua y 5 emparedados. El costo de producción por litro es de \$2.00 y el precio de venta es de \$10.00. El costo de producción por emparedado es de \$3.50 y el precio de venta es \$50. La demanda diaria de agua sigue una distribución de probabilidad cuya función de densidad es:

$$f(x) = \frac{2}{225}(x-5) \quad \text{para } 5 \leq x \leq 20$$

y la demanda de los emparedados sigue una distribución de probabilidad geométrica con media de 5 tortas/día.

Cualquier producto sobrante al final se debe tirar y no tiene valor de recuperación ni es posible venderlo al día siguiente. Simule el comportamiento de este sistema durante un año y calcule:

- a) La utilidad promedio por día que obtiene Don Cleto.
- b) La probabilidad de no vender todos los emparedados.
- c) La probabilidad de no vender toda el agua.
33. Un vendedor produce 2 hectolitros de cerveza al inicio de cada juego del equipo de futbol de la ciudad de Morelia. El costo de fabricación es de \$100/hectolitro, y el precio de venta es de \$1100/hectolitro. El producto que no logre venderse tiene un valor de rescate de \$30/hectolitro. La demanda sabatina de este producto depende del resultado del juego del equipo local (ganar, empatar o perder) y para cada caso existe una función de densidad de acuerdo a los datos de la siguiente tabla.

Equipo local	Demanda: (hectolitros/juego)
Gana	Triangular ($a = 1.5, b = 4, c = 2$)
Pierde	Weibull con parámetros de localización = 0.5, escala = 1.5, forma = 3
Empata	$f(t) = t - 1 \quad 1 \leq t \leq 1 + \sqrt{2}$

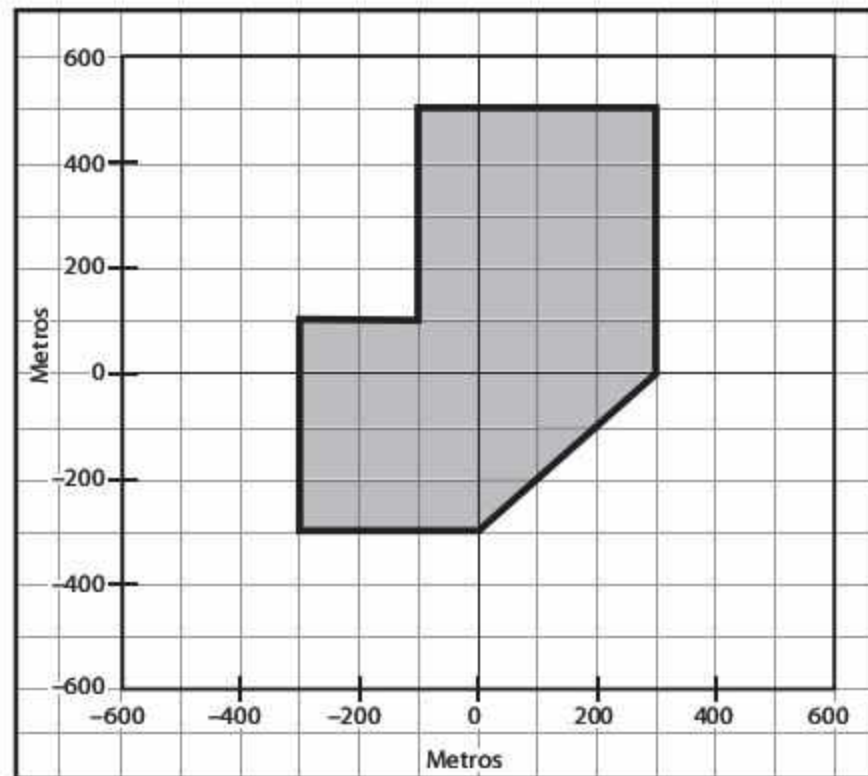
La probabilidad de que el equipo local obtenga cada uno de los resultados depende del resultado del juego anterior, de acuerdo a la siguiente matriz de probabilidades de transición (obtenida a partir de los resultados de la temporada actual) que se muestra en esta tabla.

		Resultado probable en el juego $i + 1$		
		Gana	Empata	Pierde
Resultado en el juego i	Gana	0.2	0.1	0.7
	Empata	0.4	0.3	0.3
	Pierde	0.5	0.4	0.1

Simule el comportamiento de este sistema durante 100 partidos. Considere que el último resultado de este equipo fue un triunfo y determine:

- a) La utilidad promedio por juego.
- b) La probabilidad de tener faltante.

34. Fuerzas rebeldes han decidido bombardear el palacio de un dictador. Para lograrlo, lanzarán las bombas en ataques convencionales consistentes sobre el objetivo. El área del palacio puede representarse con el polígono de la siguiente figura:



Cuando un bombardero llega al objetivo, lanza una carga de 1 bomba y aunque el bombardero siempre lanza hacia el centro del objetivo (0, 0), factores como experiencia, peso de la bomba, velocidad del bombardero, clima, visibilidad, velocidad del aire y fricción, etcétera, desvían las bombas hacia el Norte o Sur y hacia el Este o el Oeste.

Se ha demostrado en estudios previos, en otros ataques, que la posición donde cae la bomba en el eje Norte-Sur sigue una función de densidad que depende principalmente de la experiencia del piloto de acuerdo a:

Experiencia del piloto	Función de densidad
Veterano	Weibull ($\gamma = -500, \beta = 700, \alpha = 3$) metros
Novato	$f(x) = \frac{1}{500000}(x+500) \quad -500 \leq x \leq 500$ metros

Y que la posición donde cae la bomba en el eje Este-Oeste también es dependiente de la experiencia del piloto pero con las siguientes funciones de densidad:

Experiencia del piloto	Función de densidad
Veterano	Triangular ($a = -600, b = 800, c = 0$) metros
Novato	Uniforme ($a = -600, b = 600$) metros

Dentro de la fuerza de ataque rebelde, el 65% de los pilotos son novatos y el 35% veteranos.

Las fuerzas leales al dictador poseen baterías antiaéreas que permiten eliminar a los bombarderos antes que lleguen al palacio. La probabilidad de que un bombardero sea eliminado antes de llegar al palacio es de 20%.

Desarrolle un modelo de simulación MonteCarlo donde se simule el ataque de 1000 bombarderos y determine:

- La distancia en línea recta entre el centro de palacio y la posición donde pegó la bomba.
 - La probabilidad de que un bombardero logre pegarle al objetivo contabilizando aquellos eliminados antes de llegar al palacio.
 - ¿Quiénes son más precisos, los novatos o los veteranos?
 - ¿Consideraría apuntar hacia otra coordenada que no fuera el centro del palacio con el objetivo de maximizar la probabilidad de éxito? En caso afirmativo ¿Cuál sería esa coordenada? ¿Sería diferente para cada tipo de piloto?
35. En un sistema de producción, una máquina tiene un índice de producción de 1000 piezas/día (día = 24 horas). En ciertos días la máquina se descompone, la probabilidad de que la máquina se descomponga cualquier día es 0.80. El departamento de mantenimiento ha desarrollado un sistema de clasificación de fallas de acuerdo a su impacto y son: fallas leves, moderadas, graves y fatales. En la tabla siguiente se presenta un resumen de los resultados históricos de estas fallas.

Tipo de falla	Registros históricos del número de días en que ha ocurrido cada tipo de falla				
	Leve	Moderada	Grave	Fatal	Total de días
Días	10	15	25	35	85

De manera adicional se han hecho pruebas sobre el tiempo de reparación de la máquina, el cual depende del tipo de falla de acuerdo a los datos presentados en esta tabla.

Tipo de falla	Tiempo de reparación [horas]
Leve	4-Erlang(media = 4)
Moderada	Normal ($\mu = 3.5, \sigma = 1.1$)
Grave	$f(t) = t - 6 \quad 6 \leq t \leq 6 + \sqrt{2}$
Fatal	Exponencial con parámetro $\lambda = 0.2$

Durante el tiempo de reparación, la máquina no fabrica y su índice de producción disminuye proporcionalmente según la duración de la reparación. Si el precio de venta del producto es de \$100/pieza, y el costo de reparación es de \$11/hora, determine:

- a) El índice de producción diaria y su promedio después de 300 días.
- b) La utilidad diaria y su promedio después de 300 días.

36. En un proceso de fabricación de televisores, se desea saber el número de televisores a producir, considere como opciones 60, 70, 80, 90 y 100 televisores por día. En la siguiente tabla se muestran los datos históricos de la demanda por día.

99	94	80	73	86	95	90	70	92	83
89	93	62	94	86	62	77	72	63	88
96	99	98	77	96	74	86	81	94	81
70	98	78	86	74	94	95	77	79	89
65	94	88	72	71	70	76	95	92	96
93	72	97	75	97	72	76	69	61	69
99	83	83	61	64	95	80	70	79	65
80	85	72	68	83	78	84	64	72	67
94	100	78	93	66	99	97	74	93	95
70	93	64	86	72	80	83	65	66	96
95	84	88	68	65	70	88	90	85	92
81	68	92	64	84	69	73	71	98	66
68	74	93	94	86	69	82	71	79	100
84	65	99	100	61	78	73	80	72	100
100	98	82	60	93	75	84	75	76	71
94	74	84	72	71	71	85	69	88	97
93	63	91	89	89	99	93	62	99	77
69	82	66	91	73	88	61	63	67	76
100	93	82	72	75	92	87	74	60	93
77	60	84	64	83	75	68	64	97	78

El costo por faltante es de \$40/televisor por día, el costo de tener un televisor en el inventario es de \$20/televisor por día. Existe una probabilidad de 3% de que un televisor esté defectuoso debido al mal manejo de producto y al proceso mismo, por lo que antes de salir de la fábrica, un inspector de calidad lo rechaza y este televisor no puede ser reprocesado, lo que provoca una pérdida directa de \$90 por televisor. Por otro lado, el cliente realiza un proceso de inspección aleatoria para determinar si recibió televisores que se dañaron en el traslado de la planta a su tienda, el número de televisores que inspeccionan varía a diario. A continuación se muestran los recopilados del número de productos inspeccionados todos los días por el cliente.

4	2	1	1	1	2	2	0	2	1
2	2	0	2	1	0	1	0	0	2
3	3	3	1	3	1	1	1	2	1
0	3	1	1	1	2	2	1	1	2
0	2	2	1	0	0	1	2	2	3
2	1	3	1	3	1	1	0	0	0
4	1	1	0	0	2	1	0	1	0
1	1	1	0	1	1	1	0	1	0
2	4	1	2	0	3	3	1	2	2
0	2	0	1	1	1	1	0	0	2

De este proceso de inspección el número de televisores defectuosos que encuentra el cliente se distribuye de acuerdo a una distribución Binomial con una probabilidad de encontrar un televisor defectuoso de 5%. El fabricante paga el costo de un televisor defectuoso, el cual es de \$120 por televisor.

- Obtenga el costo promedio por día de cada política y un intervalo de confianza para este promedio diario.
- Con base en sus resultados determine cual de las políticas seleccionadas es la que debe implantar la empresa.



Capítulo 5

Simulación con ProModel

- 5.1** Introducción al uso de ProModel
- 5.2** Elementos básicos
- 5.3** Estructura de programación en ProModel
- 5.4** Construcción de un modelo
- 5.5** Arribos cíclicos
- 5.6** Caso integrador
- 5.7** Problemas

5.1 Introducción al uso de ProModel

ProModel es uno de los paquetes de software comerciales para simulación más usados en el mercado. Cuenta con herramientas de análisis y diseño que, unidas a la animación de los modelos bajo estudio, permiten al analista conocer mejor el problema y alcanzar resultados más confiables respecto de las decisiones a tomar.

Básicamente, este producto se enfoca a procesos de fabricación de uno o varios productos, líneas de ensamble y de transformación, entre otros. La misma compañía de desarrollo ofrece otros paquetes, como MedModel y ServiceModel, diseñados para simulación de sistemas médicos y de servicios, respectivamente. Sin embargo, aunque no es su especialidad, podemos realizar buenas simulaciones de operaciones de servicio con ProModel, tal como se verá a lo largo de este capítulo.

Para conocer las noticias más recientes sobre nuevos productos y casos de aplicación, visite la página Web <http://www.promodel.com>, la cual también pone a su disposición demos de sus artículos e información referente a ellos.

5.2 Elementos básicos

En ProModel podemos distinguir una serie de módulos que permiten al analista hacer un estudio más completo sobre el modelo que quiere simular. Cada uno de estos módulos cuenta con herramientas de trabajo que hacen de ProModel uno de los mejores paquetes de simulación que existen en el mercado. A continuación daremos una breve descripción de cada uno de ellos.

ProModel. Es el área de trabajo donde se definirán el modelo y todos sus componentes. En este módulo se programa todo lo que tiene que ver con las relaciones entre las variables del modelo, tanto contadores como relaciones lógicas, flujos, actividades y ciclos de producción, por ejemplo.

Editor gráfico. El editor gráfico de ProModel cuenta con una serie de bibliotecas que permiten dar una mejor presentación visual a los modelos realizados. Además, cuenta con la capacidad de importar y crear las imágenes necesarias para representar con mayor propiedad el problema a simular. Incluso pueden importarse dibujos hechos con algún software para dicho propósito.

Resultados. ProModel cuenta con una interfaz de resultados que facilita la administración, el manejo y el análisis de la información. En este módulo se pueden ver los resultados de todas las variables del modelo. Algunas de ellas se reportan de manera automática, y otras se obtienen bajo solicitud expresa del analista. Además, el módulo permite la interacción con programas de hoja de cálculo, como Excel.

Stat::Fit. El software incluye una herramienta estadística llamada Stat::Fit (algunas de sus funciones se comentaron ya en el capítulo 3), que permite hacer pruebas de bondad de ajuste sobre datos muestra, produciendo información muy importante para determinar las distribuciones asociadas a las variables aleatorias del modelo. Además, constituye una

gran ayuda si se desconoce cómo alimentar distribuciones complejas de la biblioteca de ProModel en el modelo de simulación.

Editor de turnos. El editor de turnos permite asignar turnos de trabajo a los elementos del modelo que lo requieran, por ejemplo, descansos programados, como el tiempo de comida.

Simrunner. Ésta es una herramienta muy útil en el análisis posterior del modelo. Con ella se pueden diseñar experimentos destinados a conocer el impacto de factores críticos que se generan a partir de la variación en los valores de las variables aleatorias seleccionadas para ello. Asimismo, permite discernir cuál es la mejor combinación de factores para obtener el máximo beneficio al mejorar un proceso.

Referencias y Ayuda. Estos módulos de ProModel facilitan el uso y la programación del software.

5.3 Estructura de programación en ProModel

En ProModel, la programación para la simulación constituye sólo una parte del proceso de construcción del modelo ya que, como se ha mencionado, el software también cuenta con diversas herramientas —de animación, por ejemplo— que el analista debe aprender a manejar para obtener los mejores resultados.

A fin de ayudarle a lograr una comprensión integral acerca del uso de ProModel, en este capítulo utilizaremos varios ejemplos que nos llevarán de lo más simple a lo más complejo. A pesar de lo anterior, esta obra no pretende cubrir de manera exhaustiva todos y cada uno de los elementos que componen el producto. Si desea obtener más detalles respecto de su funcionamiento, le recomendamos consultar los manuales de referencia que acompañan al paquete.

5.4 Construcción de un modelo

En esta sección comenzaremos nuestro análisis de algunas de las instrucciones de programación del paquete. Para empezar, comentaremos algunos modelos básicos de líneas de espera.

5.4.1 Modelo M/M/1 de líneas de espera

Un modelo sencillo de líneas de espera podría describirse como aquel en donde el tiempo entre llegadas y el tiempo de servicio son exponenciales. Consideraremos que el orden de atención (clientes en espera de algún servicio, piezas involucradas en un proceso de ensamble, etcétera) sigue la estructura “primero que llega, primero en recibir atención”. Por otro lado, daremos por sentado que tanto la capacidad de clientes (o piezas) que puede haber en el sistema analizado en un tiempo determinado como la población que puede requerir del servicio, son infinitas.

Ejemplo 5.1

Una prensa cuenta con un sistema automatizado de carga y descarga de piezas. Cada 5 minutos llegan piezas de diferentes características al sistema, con distribución exponencial. La prensa tarda 4 minutos, también con distribución exponencial, en terminar su trabajo con cada pieza, se considera carga, proceso y descarga. Suponga que puede tener cualquier cantidad de piezas que esperan ser procesadas, y simule el proceso por 100 días.

Un primer análisis del problema nos permite ver que nuestro sistema incluye diferentes elementos a considerar. Debemos suponer que las piezas llegan a una fila de espera, después son procesadas en la prensa y abandonan, por último, el área de trabajo con destino hacia algún otro almacén y/o proceso. Dado que lo que ocurra con ellas al salir de la prensa no nos interesa de momento, el sistema bajo análisis concluye cuando se terminan las piezas en la prensa. Una vez identificados estos detalles, procederemos a realizar la programación para simular el proceso en ProModel.

El primer paso, por supuesto, consiste en ejecutar el software para comenzar a trabajar en la definición del sistema que deseamos modelar. Una vez que se despliegue la ventana del programa, empezaremos por construir las localizaciones, es decir, una representación de todos aquellos lugares físicos donde las piezas serán trabajadas o esperarán su turno para ser procesadas. En este caso el sistema cuenta sólo con una fila o almacén temporal, y con la prensa en donde se realizará el trabajo. Para definir dichas localizaciones, abra el menú **Build** y haga clic en el comando **Locations**, como se muestra en la figura 5.1.

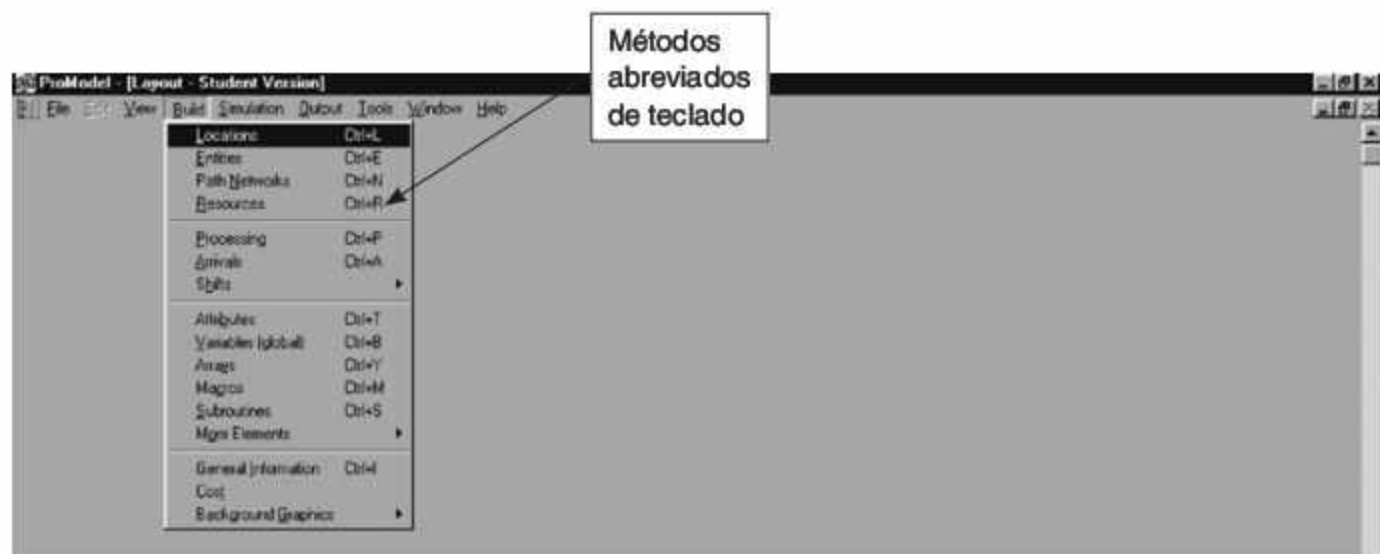


Figura 5.1 El comando Locations del menú Build nos permite comenzar a crear las localizaciones para nuestro modelo.


Además de Locations, el menú **Build** agrupa todos los comandos referentes a la construcción de elementos dentro del diseño de nuestro sistema: **Entities** (entidades),

Path Network (rutas de movimiento de los recursos o entidades), **Resources** (recursos), **Arrivals** (llegadas de entidades al sistema), y **Processing** (la programación de la simulación en sí misma), entre otros.

(Nota: Al igual que muchos otros programas, ProModel ofrece la posibilidad de acceder a sus comandos tanto a través de los menús como mediante métodos abreviados de teclado. Para conocer dichos métodos, abra cualquiera de los menús y observe la referencia a las teclas correspondientes a la derecha de cada comando [vea la figura 5.1]. Por ejemplo, para ejecutar el comando Locations oprima simultáneamente las teclas Ctrl y L.)

Una vez que ejecute el comando Locations aparecerán tres ventanas en la pantalla: **Locations**, **Graphics** y **Layout** (vea la figura 5.2). En la primera definiremos las características de las localizaciones y en la segunda las de los gráficos; la tercera ventana constituye el área en donde determinaremos la configuración general del modelo.

Gracias a la interfaz gráfica del programa, para definir cada una de las localizaciones podemos proceder de dos maneras. La primera consiste en escribir directamente en los campos de la ventana **Locations** la información correspondiente a cada localización: nombre, capacidad de atención, número de unidades, estado, reglas y demás datos relacionados. La otra es más intuitiva y aprovecha los botones del área **Graphics**. El procedimiento es como sigue:

- Haga clic con el botón izquierdo del ratón en uno de los iconos del área **Graphics** y libere el botón, ubique el cursor en el lugar del layout donde quiera colocar el ícono y vuelva a oprimir el botón izquierdo del ratón. De esta manera habrá creado una nueva localización. El ícono correspondiente tendrá un nombre preasignado en el campo **Name** de la ventana **Locations**. Para cambiar el nombre, simplemente selecciónelo y escriba. Para quitar la selección del ícono actual, sólo elija un nuevo ícono y repita la operación.
- Para señalar los lugares a donde queremos que lleguen las entidades, haga clic en el ícono predefinido de localización (un círculo con una equis ) y, sin soltar el botón del ratón, arrástrelo hasta la posición deseada en la ventana **Layout**.
- Para agregar texto a las localizaciones, haga clic en el botón de texto de la ventana **Graphics (Aa)**. Este texto puede editarse con sólo hacer doble clic sobre él.

Aunque para este primer ejemplo no es necesario, se puede alimentar el modelo con más información respecto de las localizaciones; por ejemplo, su capacidad de atención a entidades, el número de localizaciones iguales, si se tomarán en cuenta los tiempos de descompostura, etcétera.

(Nota: Es importante señalar que los gráficos únicamente constituyen un elemento visual de apoyo, y no la simulación en sí misma.)

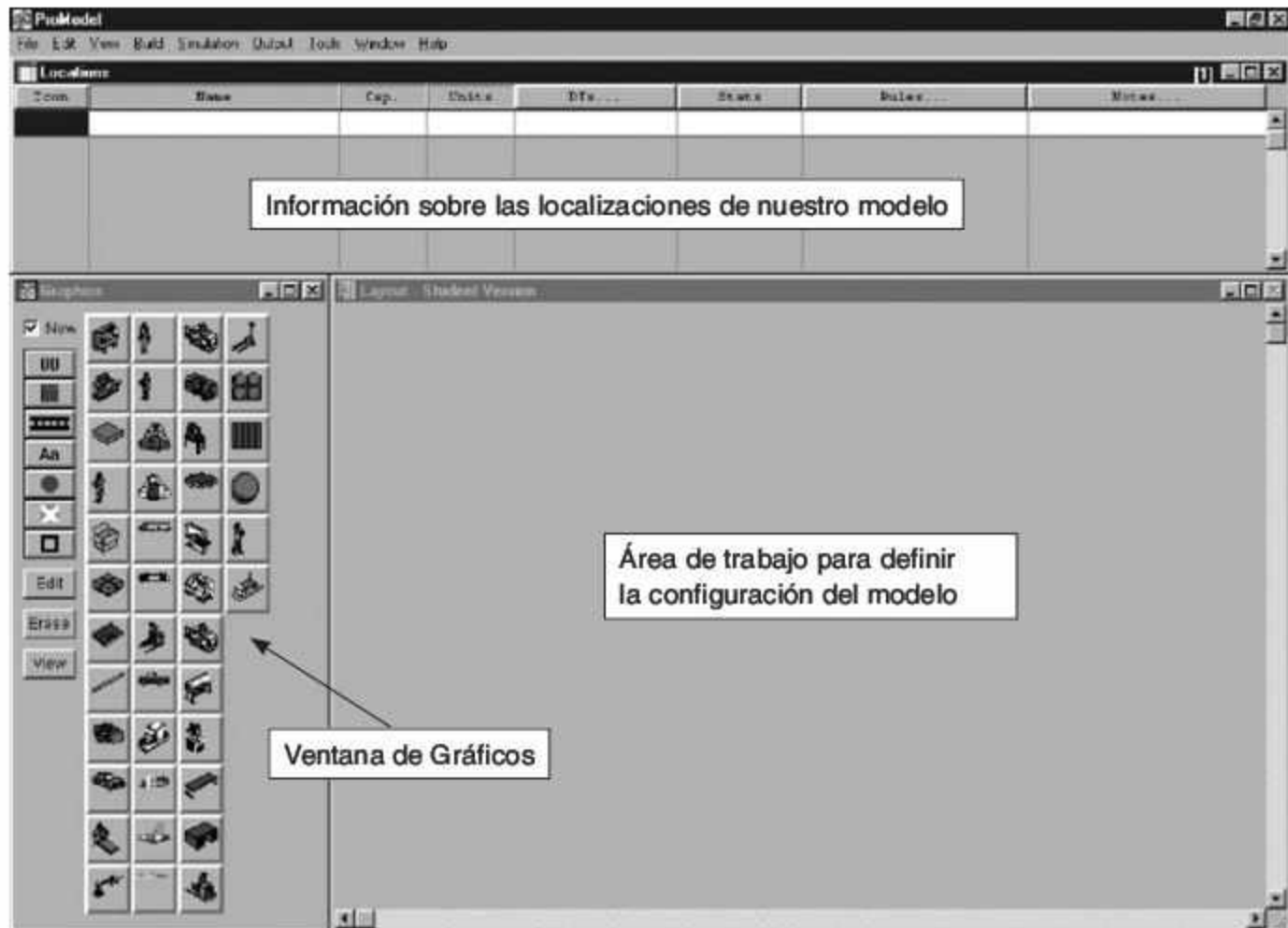


Figura 5.2 Definición de localizaciones en ProModel.


En el caso particular de nuestro ejemplo, debemos considerar que toda pieza que llegue puede esperar a ser atendida. Para ello definiremos una localización a la que llamaremos "fila", y le asignaremos una capacidad infinita en el campo **Cap.** al escribir *infinite*, o simplemente *inf*, para cada localización.

Si por alguna razón deseáramos cambiar el icono de una localización, todo lo que hay que hacer es:

- Seleccionar la fila en que reside dentro de la ventana **Locations**.
- Oprimir la tecla **Supr** (o **Delete**).
- Desmarcar la casilla de verificación **New** de la ventana **Graphics** y seleccionar el nuevo icono.

Otra posibilidad consiste en hacer clic con el botón derecho del ratón en el icono que define la localización en la ventana **Layout**. Enseguida aparecerá un menú contextual con comandos para editar o eliminar la localización y borrar, incluso, toda la información referente a ella. Por ejemplo, si selecciona el comando **Edit Graphic** podrá modificar el tamaño y color del icono seleccionado, pero no la localización en sí misma. Para usar un gráfico diferente que identifique la localización, tendrá que borrar el actual y remplazarlo por el nuevo.

(**Nota:** Si olvida desmarcar la casilla de verificación **New** de la ventana **Graphics**, al realizar cualquiera de las acciones anteriores estará creando más localizaciones de las necesarias [en el caso de nuestro ejemplo, dos]. Para eliminar aquellas que no le sean útiles, seleccione la fila apropiada en la ventana **Locations**, abra el menú **Edit** y haga clic en el comando **Delete**.)

Para continuar, definiremos la localización que representará la prensa. Igual que antes, seleccione un icono cualquiera en la ventana **Graphics**. Gracias a la interfaz gráfica de ProModel puede anexar una posición sobre el icono, de manera que “se vea” que la pieza llega a la prensa; para ello, emplee el botón . Una vez concluidas estas definiciones preliminares, la ventana Layout podría lucir como se ilustra en la figura 5.3.

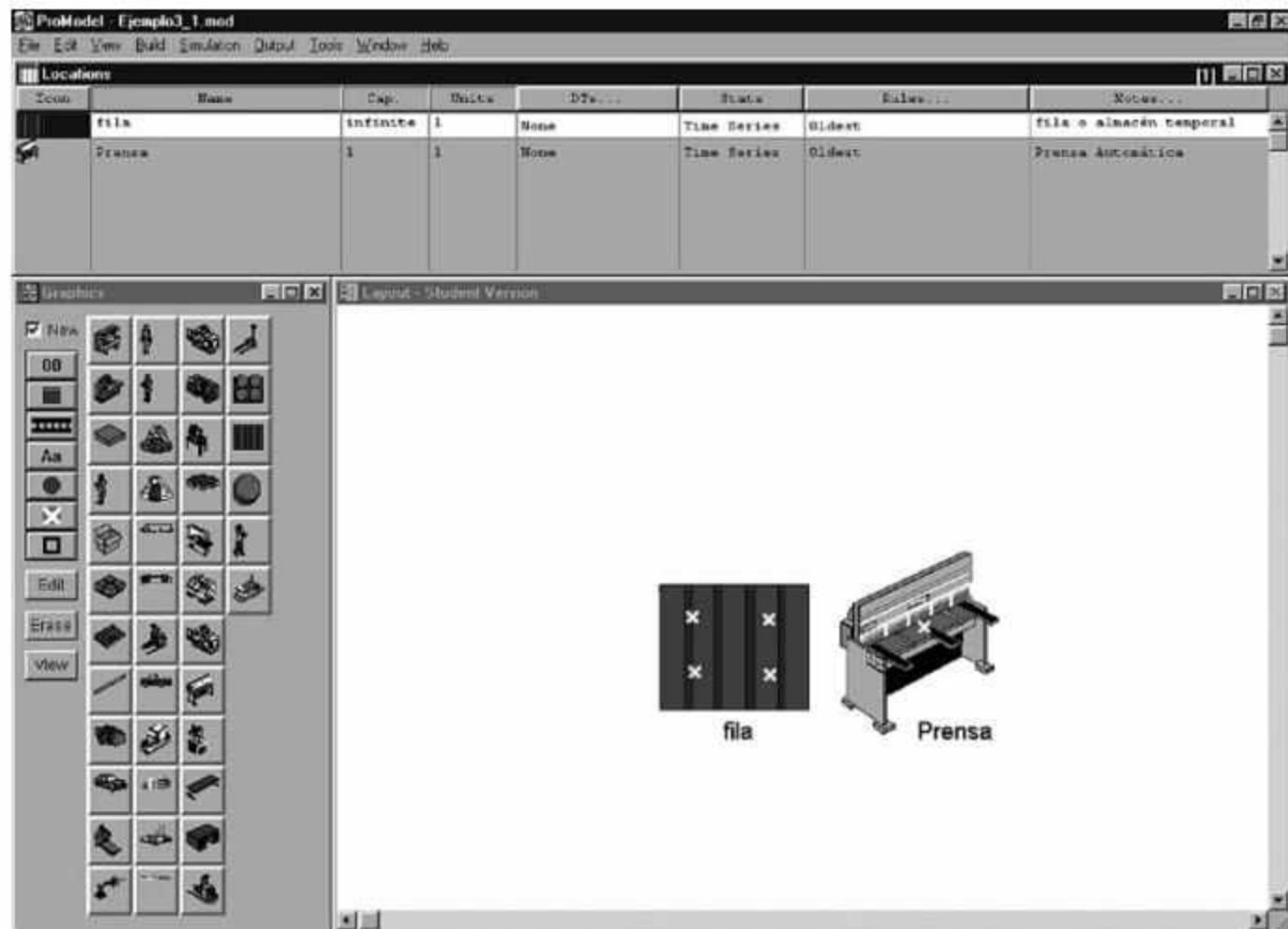


Figura 5.3 Definición de localizaciones en el Layout.

Una vez definida la configuración del proceso, pasaremos a definir la *entidad* que representará la pieza en proceso. Para ello:

- Abra el menú **Build** y haga clic en el comando **Entities**. Una vez más, en la pantalla aparecerán tres ventanas: **Entities**, **Entity Graphics** y **Layout**, cuyo propósito es muy similar al de sus equivalentes en el caso de la definición de localizaciones.

Tanto la definición de entidades como su edición se llevan a cabo mediante procedimientos parecidos a los que se realizaron con las localizaciones. Es posible modificar el gráfico seleccionado para cambiar sus dimensiones y su color, y definir, como se describe a continuación, varios gráficos para una misma entidad:

- Desmarque la casilla de verificación **New** de la ventana **Entity Graphics**. Enseguida aparecerán nuevos lugares para definir más iconos que identifican la misma entidad; una vez seleccionado el icono, su pantalla será similar a la que se ilustra en la figura 5.4.

(Nota: Al igual que en el caso de las localizaciones, si mantiene marcada la casilla de verificación **New** definirá nuevas entidades con cada selección de icono que haga.)

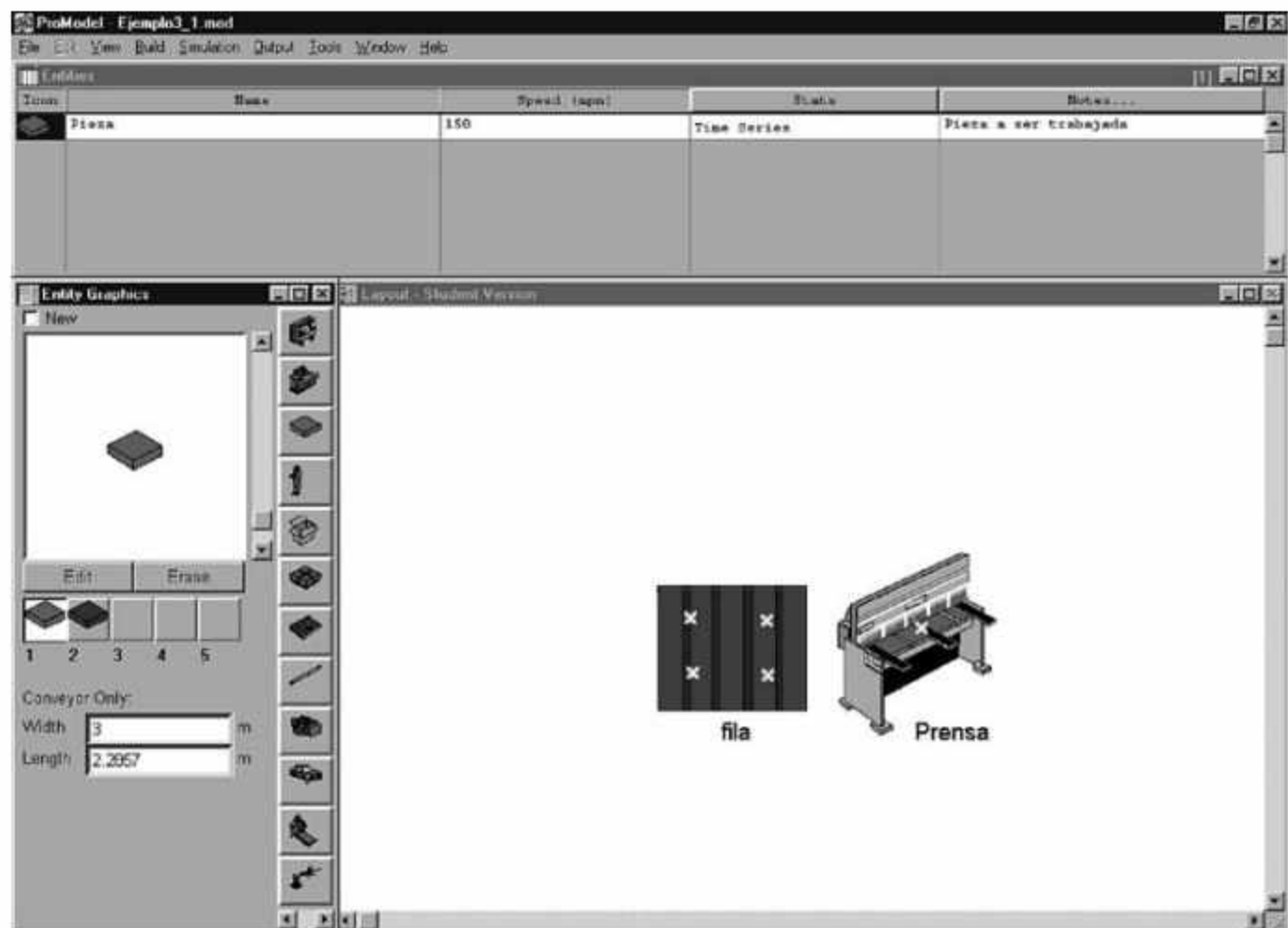


Figura 5.4 Definición de entidades.

Una vez definidas las entidades determinaremos su frecuencia de llegadas a nuestro modelo. Para ello:

- Abra el menú **Build** y haga clic en el comando **Arrivals**. A continuación se desplegará la ventana **Arrivals** (vea la figura 5.5). En ella definiremos la frecuencia de llegadas para nuestra pieza.

Entity	Location	Qty Each	First Time	Occurrences	Frequency	Log on	Visible
Pieza	Fila	1	0	inf	e(5)min		No

Figura 5.5 Definición de llegadas de la entidad al sistema.

- Para seleccionar la entidad oprima el botón **Entity**. Luego especifique a qué localización llegará la entidad; en este caso será a una localización llamada "fila": haga clic en el botón **Location** para que se desplieguen todas las localizaciones que definimos previamente.
- Ahora determine, en la columna **Qty Each**, cuántas piezas llegarán cada vez que se cumpla el tiempo entre llegadas; en este caso determinamos una (1) a la vez.
- Prosiga su trabajo, especifique esta vez el tiempo de ocurrencia del primer evento de llegada en la columna **First Time**.
- En la columna **Occurrences** debe indicarse el número de repeticiones del evento de llegada. En este caso especifique **infinite** (o simplemente **inf**), lo cual implica que se admitirá un número infinito de eventos de llegada.
- En la columna de **Frequency** especifique la distribución del tiempo entre llegadas; manejaremos un valor exponencial con media de 5 minutos: **e(5) min**.

(Nota: Si desea conocer las opciones predeterminadas de las distribuciones de probabilidad que ofrece ProModel, despliegue la ayuda del programa haciendo clic en el menú **Help**, consulte el tema **Functions** y elija la opción **Probability Distributions**).

Por último completaremos nuestro modelo definiendo la lógica de la simulación; para ello abra el menú **Build** y elija **Processing**. En esta ocasión se desplegarán dos ventanas en las que programaremos de manera secuencial el proceso que sigue la pieza en el sistema: **Process** y **Routing for**. En la primera definiremos las operaciones que se harán sobre la entidad, y en la segunda indicaremos la ruta secuencial en el proceso. Al analizar una vez más el ejemplo, verá que podemos dividir el proceso en los siguientes pasos:

1. La pieza llega a la fila para esperar su turno de procesamiento. Cuando se cumpla la condición sobre el estado de la prensa, la pieza abandonará la fila y seguirá su ruta hacia la localización "prensa".
2. La pieza llega a la prensa, donde se le procesa durante un tiempo promedio de 4 minutos, con distribución exponencial. Una vez terminado el proceso en la prensa, la pieza abandona esta localización; su siguiente paso es salir del sistema.

Cada uno de estos pasos deberá programarse de manera independiente, es decir, en un registro separado. Empezaremos por definir la llegada de las piezas a la fila. Para ello:

- Seleccione la entidad correspondiente en la ventana **Processing**, ya sea haciendo clic en el botón **Entity** o escribiendo directamente el nombre de la entidad en el campo de dicha columna.
- Para programar la localización de llegada de la entidad (en este caso la localización llamada "fila"), haga clic en el botón **Location**; debajo se desplegarán todas las localizaciones definidas.

Puesto que en esta localización la pieza sólo espera a que la prensa esté disponible, no se programa nada en la columna **Operation**. A continuación definiremos la ruta de salida en la ventana **Routing for**:

- En este caso la entidad de salida es de nuevo la pieza, por lo que ése es el nombre que escribimos en la columna **Output**.
- El destino de la pieza es la prensa, así que seleccionamos dicha localización en la columna **Destination**.
- La siguiente columna, **Rule**, indica la regla de movimiento; el valor predeterminado aquí es **FIRST 1**, lo que significa que la entidad avanzará tan pronto se tenga capacidad disponible en la localización de destino.
- La última columna, **Move Logic**, determina el movimiento lógico de salida; en este caso dejaremos en blanco el campo. Una vez completada, la primera línea de programación deberá quedar como se ilustra en la figura 5.6.

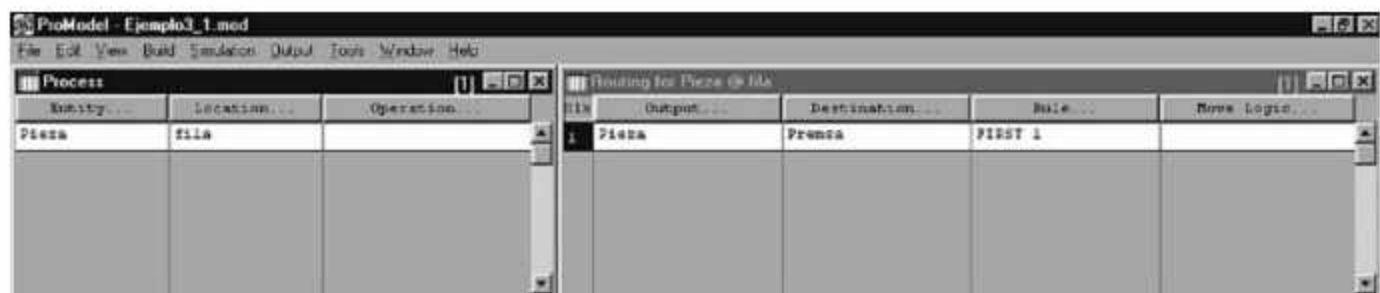


Figura 5.6 Definición de la primera línea de programación.

Para continuar es preciso definir el proceso que se llevará a cabo con la pieza en la prensa. Una vez más, comenzaremos por establecer que la entidad cuyo comportamiento nos interesa es la pieza, que la localización en la que se encuentra es la prensa, y que el proceso ocupa un tiempo específico de esta localización: 4 minutos promedio con distribución exponencial. Para conocer los comandos de programación necesarios para especificar lo anterior, haga clic en el botón **Operation** de la ventana **Process**. Enseguida se desplegará la ventana **Operation** (vea la figura 5.7), en donde se escribirá la lógica del proceso.

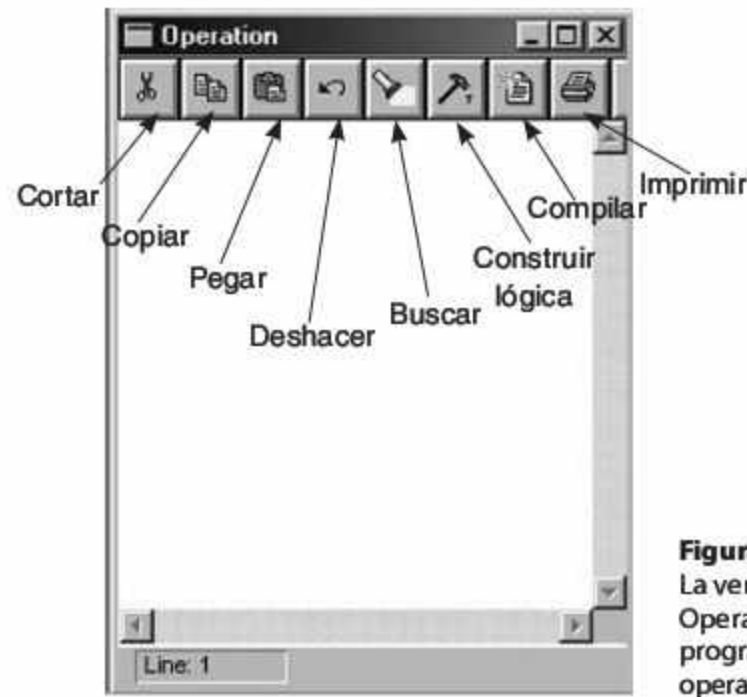


Figura 5.7
La ventana Operation permite programar las operaciones.

- Haga clic en el icono de martillo para comenzar la construcción lógica. Al hacerlo se abrirá otra ventana, que contiene todos los comandos de programación existentes.

ProModel hará una sugerencia de comandos que podrían resultar útiles. Al colocar el cursor del ratón sobre cada uno de ellos se mostrará una sugerencia en pantalla con una breve descripción de su utilización.

El comando que puede ser de utilidad en nuestro caso es **WAIT**, que implica una espera de la entidad en cierto momento (por ejemplo, para realizar una operación). Toda vez que queremos manejar un tiempo exponencial de 4 minutos, la instrucción completa será **WAIT E(4) min**. La sintaxis general del comando es la siguiente:

WAIT <unidades de tiempo>

Procedamos a definir la ruta de salida de este registro. En este caso la entidad de salida es la pieza, y su destino es salir del sistema. Finalmente la programación debe lucir como se ilustra en la figura 5.8.

Entity...	Location...	Operation...
Pieza	fila	
Pieza	Prensa	WAIT E(4)

File	Output...	Destination...	Rule...	More Logic...
1	Pieza	EXIT	FIRST 1	

Figura 5.8 Definición de la segunda línea de programación.

Observe que, al definir el segundo registro, la ventana de la ruta de salida empieza de cero. Esto significa que la ventana de programación nos permite ver los procesos de las piezas de manera secuencial, aunque la ventana correspondiente a la ruta de salida del proceso sólo mostrará la programación correspondiente a la línea seleccionada en la ventana del procesamiento.

Finalizada la programación, nos queda por definir el tiempo de simulación. Para ello:

- Abra el menú **Simulation** y haga clic en el comando **Options**. Enseguida se abrirá la ventana correspondiente, ahí, en el campo **Run hours** escribiremos **100 day**. En el campo **Number of Replications** podemos determinar el número de veces que deseamos correr el modelo, es decir el número de replicas. En este caso sólo requeriremos de una repetición.
- Seleccione en la sección **Output Viewer(s) to launch** el formato deseado para el reporte de resultados, para este ejemplo elegiremos la opción **Output Viewer 2.0 (3DR)**.

El modelo está terminado. Para ejecutarlo, lo único que tiene que hacer es desplegar el menú **Simulation** y hacer clic en el comando **Save & Run**. Una vez que esté corriéndose la simulación, podrá —si así lo desea— ajustar su velocidad con la barra que aparece en la parte superior de la ventana, o cancelar la animación mediante el comando **Animation Off** del menú **Options**.

Al terminar la simulación de los 100 días se desplegará un cuadro de mensaje que confirmará la finalización del tiempo programado. Si desea ver los resultados, haga clic en el botón **Yes** (éstos pueden compararse con los que se obtienen teóricamente mediante las ecuaciones matemáticas de líneas de espera para un modelo M/M/1). Enseguida se abrirá una ventana con varias fichas que muestran los resultados estadísticos de la simulación (vea la figura 5.9). Los datos pueden leerse y graficarse de inmediato con las herramientas que ofrece ProModel, o guardarse en archivos con formato de Excel para luego personalizarlos. En ambos casos podremos encontrar la siguiente información relevante (las cifras pueden variar dependiendo de los números aleatorios que haya utilizado durante la simulación).

General for ejemplo3_1	
Name	Value
Run Date/Time	30/03/04 10:21:50 p.m.
Model Path/File	C:\personal\libro\modelos Libro\Ejemplo3_1.mod
Model Title	

Figura 5.9 Reporte de datos generales del modelo (ficha General).

Ficha General

Los datos que despliega esta ficha indican qué archivo se usó para obtener los resultados, así como la fecha y hora en la que se realizó la simulación.

Ficha Locations

En esta sección (vea la figura 5.10) se presenta la información de cada una de las localizaciones, las horas simuladas, su capacidad (en este caso la capacidad infinita se representa con 999999), el número total de entidades que entraron durante la simulación, el tiempo promedio de estancia de las entidades en cada localización, el número promedio de piezas, el número máximo de entidades, el número actual de entidades al momento de finalizar la simulación, y el porcentaje de utilización de cada una de las localizaciones. También se pueden revisar las estadísticas independientes de cada localización con capacidad unitaria—como la prensa— y de aquellas que tienen capacidad mayor a uno—como la de la fila.

Locations for ejemplo3_1								
Name	Scheduled Time (MIN)	Capacity	Total Entries	Avg Time Per Entry (MIN)	Avg Contents	Maximum Contents	Current Contents	% Utilization
fila	144000.00	999999.00	28524.00	15.36	3.04	41.00	2.00	0.00
Prensa	144000.00	1.00	28522.00	4.00	0.79	1.00	1.00	79.28

Figura 5.10 Reporte estadístico de las localizaciones (ficha Locations).

La ficha Locations también incluye información respecto de los parámetros de un sistema de líneas de espera, como: la utilización de la prensa (P), que es un porcentaje de la operación; el número promedio de clientes en el sistema (L), que es el **Avg Contents** de la fila más el **Avg Contents** de la prensa; el número promedio de clientes en la fila (Lq), que es el **Avg Contents** de la fila; el tiempo promedio de permanencia en el sistema (W), que es la suma de los **Avg time per Entry** de la fila y de la prensa, y el tiempo promedio de permanencia en la fila (Wq), que es únicamente el tiempo de la fila.

Si comparamos estos resultados con los teóricos, veremos que son muy similares (vea la tabla 5.1). La diferencia puede deberse a que la simulación no ha llegado a estado estable, o a la variabilidad natural del modelo. En cualquier caso, es recomendable graficar la variable o variables de respuesta que se desea comparar.

Tabla 5.1 Comparación entre los resultados teóricos y los obtenidos mediante simulación.

Parámetro	Resultado teórico	Resultado de la simulación
L	4 piezas	3.83 piezas
Lq	3.2 piezas	3.04 piezas
W	20 minutos	19.35 minutos
Wq	16 minutos	15.35 minutos
P	80%	79.28%

Fichas Locations States Single/Tank y Locations States Multi

En la primera de estas fichas se presenta la información de las localizaciones que tienen capacidad de uno (conocidas como de capacidad unitaria), y la segunda la de aquellas que pueden contener más de una entidad a la vez durante la simulación (denominadas de multicapacidad; vea las figuras 5.11 y 5.12). En nuestro ejemplo tenemos una de cada tipo: la localización “fila” tiene capacidad infinita, mientras que la localización “prensa” tiene capacidad de uno.¹

Location States Single/Tank for ejemplo3_1							
Name	Scheduled Time (MIN)	% Operation	% Setup	% Idle	% Waiting	% Blocked	% Down
Prensa	144000.00	79.28	0.00	20.72	0.00	0.00	0.00

Figura 5.11 Reporte de localizaciones con capacidad unitaria (es decir, con capacidad para una sola entidad).

Location States Multi for ejemplo3_1					
Name	Scheduled Time (MIN)	% Empty	% Part Occupied	% Full	% Down
fila	144000.00	37.29	62.71	0.00	0.00

Figura 5.12 Reporte de localizaciones con multicapacidad (es decir, con capacidad para varias entidades).

En esta sección del reporte podemos encontrar información referente al porcentaje de tiempo vacío, ocupado de manera parcial y no libre respecto del tiempo disponible para cada localización con capacidad mayor a uno. En este caso, la localización “fila” se encuentra 37.29% del tiempo vacío, 62.71% del tiempo con al menos una pieza, y nunca llena ni no disponible, pues le asignamos capacidad infinita y no se programaron eventos que limitaran el acceso y/o salida de las entidades a esta localización. Por otro lado, la localización “prensa” es de capacidad unitaria, así que el reporte informa el porcentaje de tiempo que la prensa estuvo procesando alguna pieza (79.28% del tiempo), el porcentaje de tiempo dedicado a actividades de preparación (en este ejemplo no existían).

¹Es la suma de piezas en ambas localizaciones; lo mismo sucede en el caso del tiempo total de permanencia en el sistema.

ten), el porcentaje de tiempo que la prensa estuvo inactiva debido a que no había piezas que procesar, el porcentaje de tiempo que la localización espera a que un recurso u otra entidad lleguen para iniciar el proceso (por ejemplo, en las situaciones en que hay ensambles o cuando la prensa requiere de un dado especial para procesar la pieza), el porcentaje de tiempo en que la localización no está realizando trabajo alguno —ya que la capacidad de su localización destino está llena—, y finalmente el porcentaje de tiempo en el que la localización se encuentra no disponible.

Ficha Failed Arrivals

Esta ficha (vea la figura 5.13) lista las entidades de cada modelo, e indica si alguna de ellas no pudo entrar al sistema en la localización definida en **Arrivals**. Esto puede suceder, por ejemplo, cuando la localización de llegada tiene una capacidad finita. Si ésta se completa y una entidad desea ocupar un espacio en la localización, al no poder encontrarlo es destruida y eliminada del sistema. Esta información es útil, por ejemplo, cuando se analizan sistemas de líneas de espera con capacidad finita y se desea saber el porcentaje de clientes que no pudieron ser atendidos.

Failed Arrivals for ejemplo3_1		
Entity Name	Location Name	Total Failed
Pieza	fila	0.00

Figura 5.13 Estadística de entidades no ingresadas (ficha Failed Arrivals).

Ficha Entity Activity

Esta ficha del reporte refleja las estadísticas de cada entidad definida en el modelo. Como se observa en la figura 5.14, en este caso sólo tenemos la entidad llamada “pieza”. La información reportada es la entidad, el total de entidades que salieron del sistema (en este ejemplo 28521), las entidades que se encuentran en el sistema al finalizar la simulación, el tiempo promedio de permanencia en el sistema (19.35 minutos, que es el mismo que se informa en la ficha **Locations**), el tiempo promedio que la entidad pasó en un traslado o movimiento de una localización a otra (mismo que no se programó en nuestro modelo), el tiempo promedio que la entidad espera a otra entidad para un ensamble, o a un recurso para ser procesada o transportada (por ejemplo, por un montacargas), el tiempo promedio que se encuentra en procesamiento o viajando en un transportador y, finalmente, el tiempo que no puede avanzar debido a que la localización destino está totalmente ocupada (15.36 minutos, el tiempo promedio de espera en la fila).

Entity Activity for ejemplo3_1							
Name	Total Exits	Current Qty In System	Avg Time In System (MIN)	Avg Time In Move Logic (MIN)	Avg Time Wait For Res (MIN)	Avg Time In Operation (MIN)	Avg Time Blocked (MIN)
Pieza	29521.00	3.00	19.36	0.00	0.00	4.00	15.36

Figura 5.14 Estadísticas de la actividad de las entidades en el sistema (ficha Entity Activity).

Ficha Entity States

En esta ficha del reporte (vea la figura 5.15) podemos encontrar un resumen de los datos de la ficha Entity Activity, pero en términos porcentuales. Por ejemplo, como en este caso la entidad “pieza” pasa sólo 4 minutos en operación, el reporte indica que pasó 20.68% del tiempo total de permanencia en el sistema (19.35 minutos), mientras que estuvo bloqueada para continuar su camino a la localización destino el tiempo restante, 15.35 minutos (es decir, 79.32% del tiempo total).

Entity States for ejemplo3_1				
Name	% In Move Logic	% Wait For Res	% In Operation	% Blocked
Pieza	0.00	0.00	20.68	79.32

Figura 5.15 Estadística porcentual de la actividad de las entidades (ficha Entity States).

5.4.2 Mejoramiento visual del modelo

ProModel permite incrementar la capacidad visual del modelo mediante un conjunto de herramientas específicas para dicho propósito. En esta sección hablaremos sobre cómo utilizarlas, con base una vez más en el modelo que se construyó para el ejemplo 5.1.

Ejemplo 5.2

Nuestro trabajo en esta sección se basará en el ejemplo 5.1, aunque le haremos algunas modificaciones con el objetivo de mejorar su presentación al momento de ejecutar la simulación. Además, trataremos de obtener información relevante para el tomador de decisiones y/o para el programador del modelo.

Para comenzar, determinaremos la cantidad de piezas que hay en el almacén en cualquier momento dado. Esto se puede hacer de dos formas:

- Abra el menú **Build** y haga clic en el comando **Locations**.
- En la ventana **Graphics**, haga clic en el icono predeterminado para la función de contabilización de entidades en una localización (**00**). (Es importante resaltar que debe desmarcar la casilla de verificación **New** para poder integrar este contador a la localización que deseamos editar.)
- Vaya a la columna **Cap.** del registro de la localización que desea modificar (en este caso "fila"), y cambie su capacidad a **50**.
- Seleccione los iconos correspondientes en la ventana **Graphics**, como se muestra en la figura 5.16.

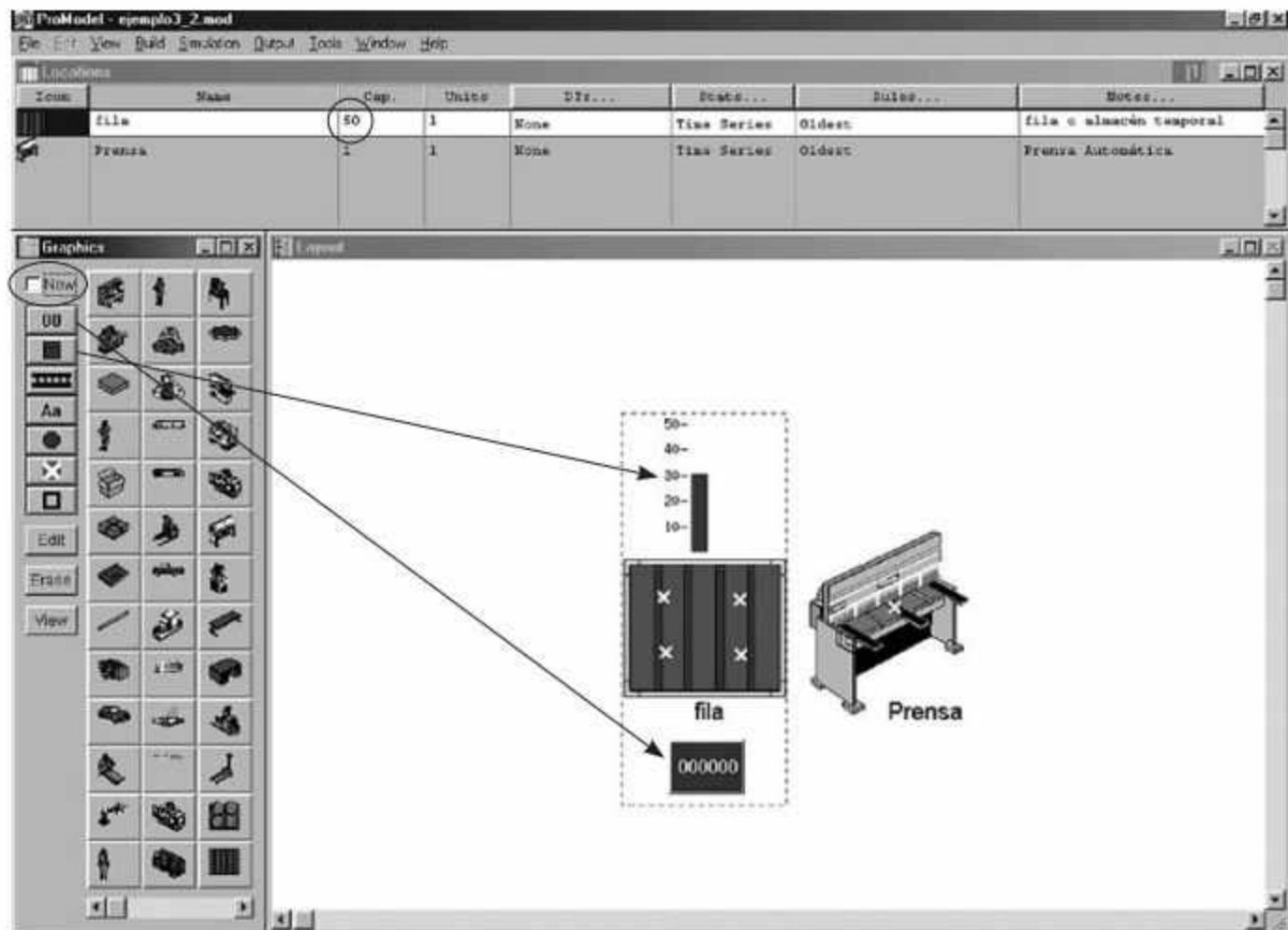


Figura 5.16 Determinación de la cantidad de piezas en una localización.

- Agregue una barra que ilustre la capacidad utilizada del total (es por eso que cambiamos la capacidad de la localización a 50; si la hubiéramos mantenido en infinito no aparecería registro alguno en la barra). Para ello emplearemos el icono predeterminado, la barra de color azul que se encuentra debajo del icono **00** en la ventana **Graphics**. Si al momento de colocar la barra de capacidad no ve la escala, de un doble click con el ratón sobre la barra y se abrirá la ventana de diálogo **Gauge/Tank** donde podrá cambiar las características visuales de la barra, active la opción **Show Escale** y cierre la ventana. Al hacerlo su pantalla deberá lucir como se ilustra en la figura 5.16.

La otra manera de llevar a cabo este procedimiento consiste en utilizar una variable e igualarla al comando predeterminado **CONTENTS(fila)** para contabilizar los contenidos de las localizaciones.

Para agregar el número de piezas procesadas utilizaremos una variable. Para ello:

- Abra el menú **Build**, haga clic en el submenú **More Elements** y elija **Variables (global)**. Enseguida se desplegará en pantalla la ventana de definición de variables, misma que se ilustra en la figura 5.17.



Figura 5.17 La ventana Variables (global) nos servirá para definir las variables del modelo.

Nuestro propósito es definir los parámetros de la variable *pzas_tot*. Para ello:

- Colóquese en el primer campo (**ID**) y modifique el nombre de la variable.
- Cambie al campo de la segunda columna (**Type**) para definir el tipo de variable, que puede ser entera (integer) o real; en este caso la variable es entera.
- En el campo de la siguiente columna, **Inicial Value**, determine como **0** el valor inicial de la variable.
- Como queremos que el icono de esta variable aparezca en la simulación, haga clic en la columna **Icon** y después, nuevamente en el lugar en donde quiere que aparezca el contador (vea la figura 5.18).

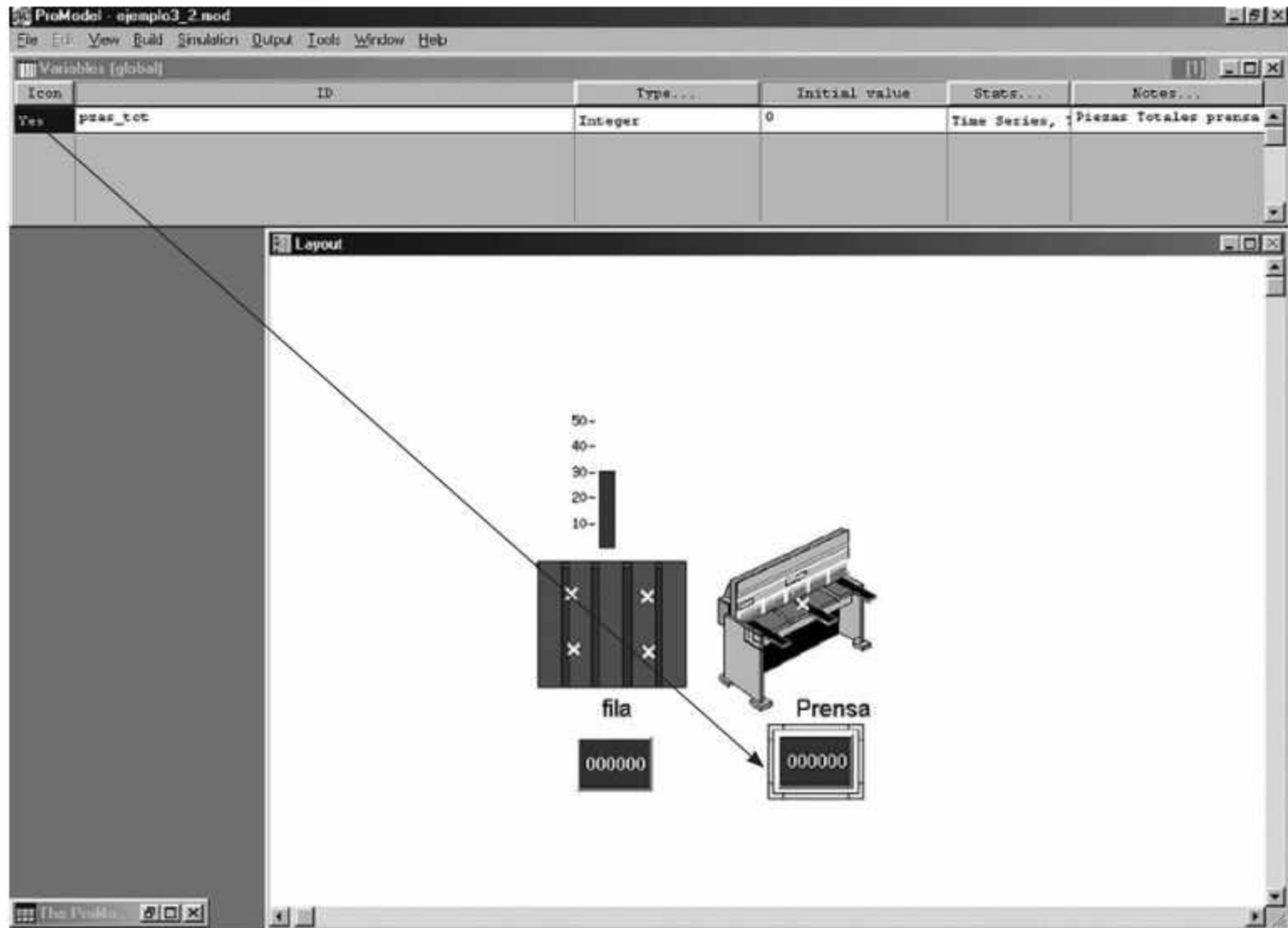


Figura 5.18 Determinación de los parámetros de la variable que contabiliza piezas totales.

El siguiente paso consiste en especificar que la variable cambie cada vez que entre una pieza a la prensa. Esto se logra al programar esta acción como una operación que se ejecutará al momento de que la pieza termine de ser procesada en la prensa. Para lograrlo:

- Elija el comando **Processing** del menú **Build**. En este caso añadiremos la instrucción **pzas_tot = ENTRIES(Prensa)** en el segundo registro de la programación, que corresponde al proceso que se realiza en la localización "prensa". Esta línea de programación hará que cada vez que una pieza termine su proceso de 4 minutos con distribución exponencial en la prensa se contabilice como una pieza terminada. La programación deberá quedar como se muestra en la figura 5.19.

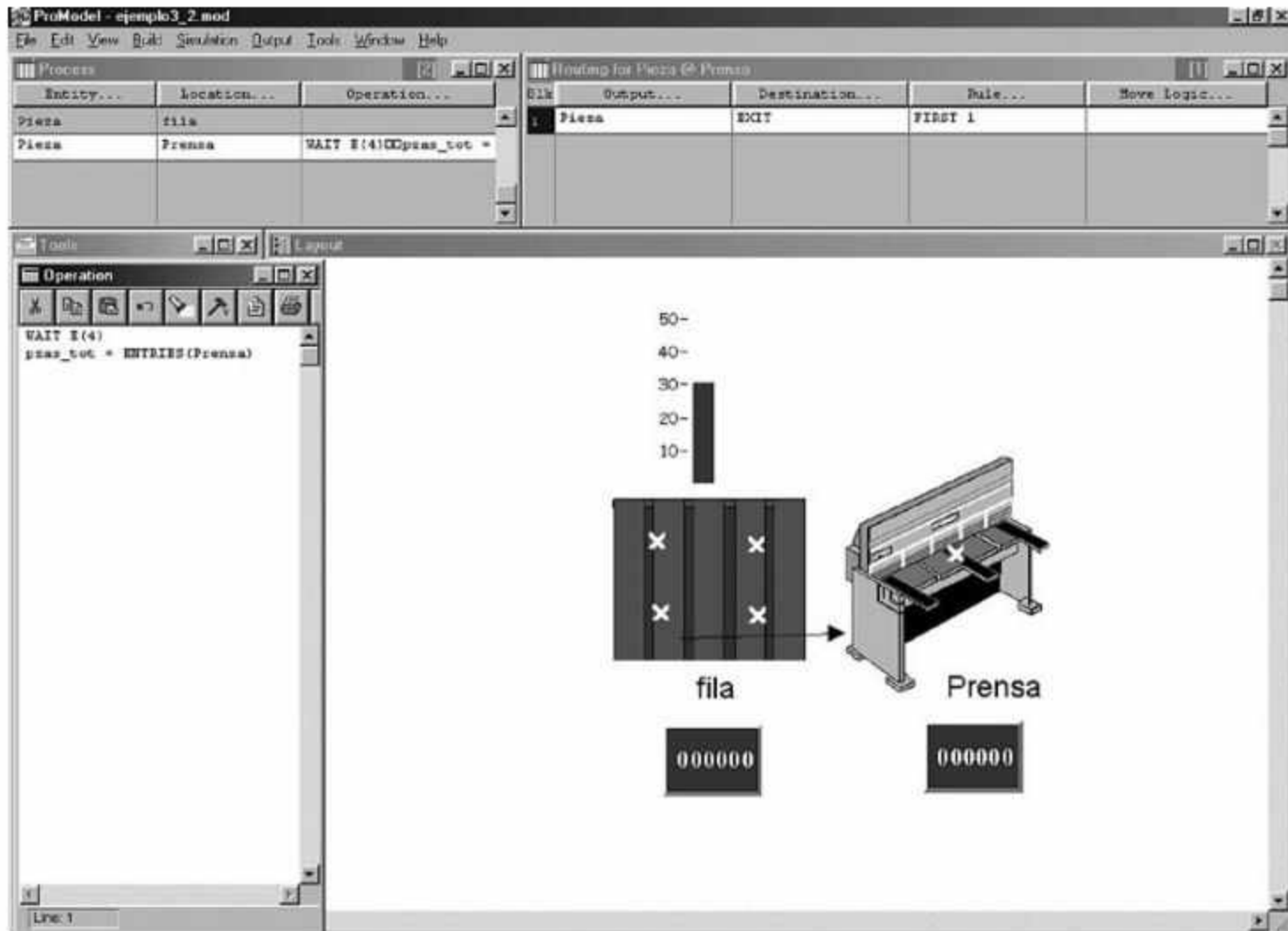


Figura 5.19 Uso de la instrucción ENTRIES().

Si corriéramos la simulación en este momento, podríamos ver que tanto el contador como la barra reflejan la cantidad de piezas que se encuentran en un momento determinado en el almacén definido en la simulación. Sin duda el modelo ya simula el problema que estamos ejemplificando, a pesar de que lo único que hemos hecho es agregar un par de gráficos que hagan más entendible lo que pasa.

La segunda modificación que haremos consistirá en cambiar el tiempo de simulación, de manera que su ejecución no sea muy larga. Suponga que queremos cambiar la duración del modelo a solo 30 días. Para ello:

- Despliegue el menú **Simulation** y haga clic en el comando **Options**, como se muestra en la figura 5.20.



Figura 5.20 Acceso a las opciones de la simulación.

- A continuación se abrirá el cuadro de diálogo **Simulation Options** (vea la figura 5.21), en donde es posible modificar varias opciones de la simulación. Por lo pronto, cambie el valor del cuadro de texto **Run hours** a **30 days** y haga clic en **OK**.

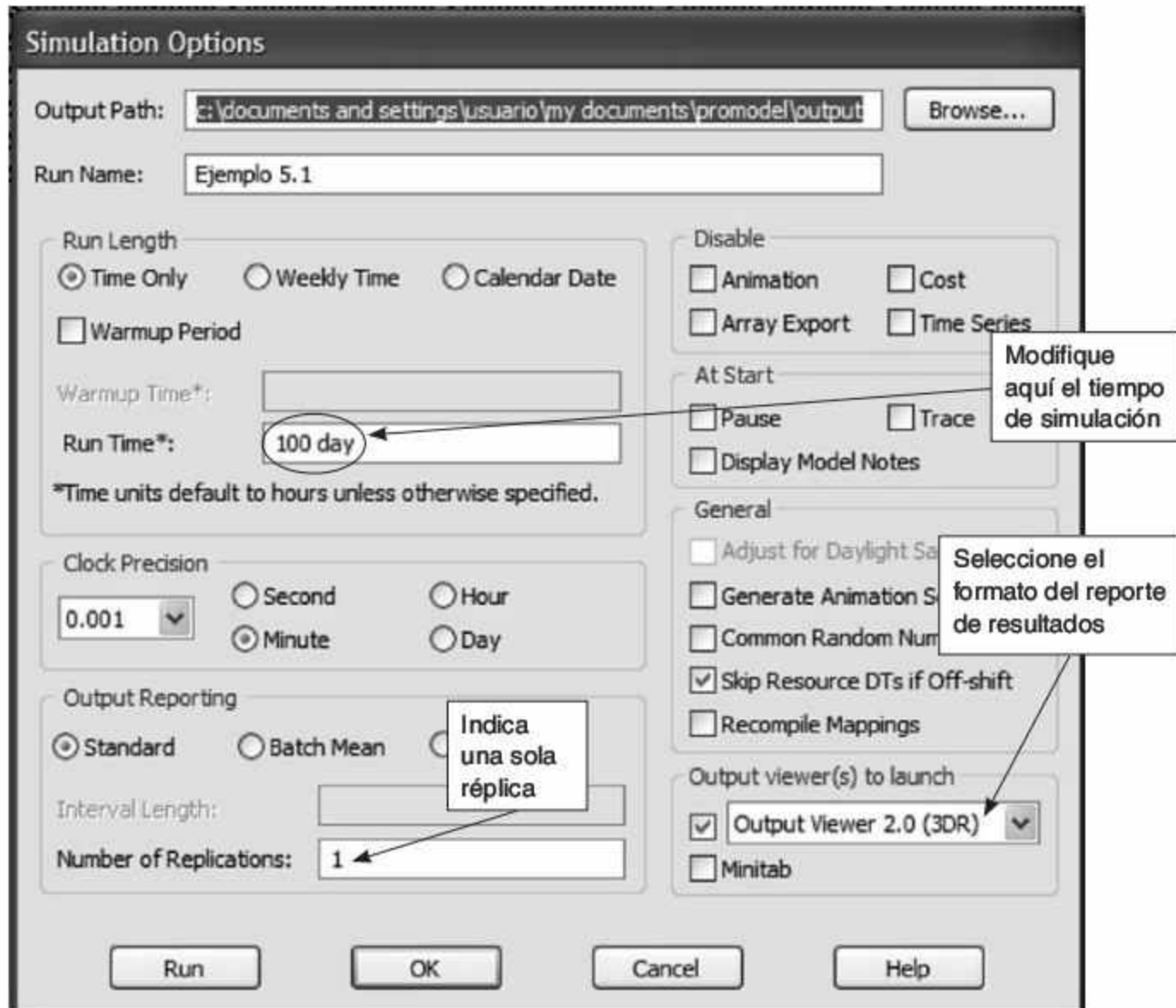


Figura 5.21 El cuadro de diálogo Simulation Options permite determinar diversos parámetros de la simulación.

Una interrogante importante para el tomador de decisiones es si las variables del modelo están en estado estable o todavía se encuentran en un estado transitorio. Como se mencionó en el capítulo anterior, si queremos evitar la variabilidad que ofrecen los resultados del estado transitorio, es necesario que nuestras soluciones se basen en las estadísticas del estado estable.

Una opción de mucha utilidad para obtener estadísticas estables consiste en definir un tiempo transitorio o de preparación (*warmup*) dentro del modelo. Como probablemente recordará, la gráfica de estabilización que mencionamos en capítulos anteriores nos muestra que los valores de las variables de respuesta en el estado transitorio suelen describir una alta variabilidad. Para evitar que el efecto de esta variación se diluya y poda-

mos obtener respuestas estables, es necesario contar con mayor tiempo de ejecución. Definir un periodo de *warmup* implica ejecutar el modelo durante cierto tiempo, después del cual las estadísticas regresarán a cero. Gracias a ello se eliminarán los registros correspondientes a las variables de respuesta en el estado transitorio, y se conservará únicamente el valor final de las variables de respuesta, lo cual implica menos tiempo de simulación y, por consiguiente, menor inversión de tiempo de computadora y menor costo. Esto es muy útil en casos en los que el sistema se encuentra vacío en el arranque. Si se da un tiempo transitorio mientras el sistema se llena, ese tiempo sería el que colocaríamos de *warmup*. Le sugerimos acomodar un tiempo de un día para comparar resultados entre los modelos con y sin tiempo transitorio. La simulación terminará cuando se cumplan 31 días: un tiempo transitorio que no será tomado en cuenta al generar las estadísticas, y 30 días que sí aportarán datos para obtener los promedios finales de las variables de respuesta.

Ahora nos falta colocar algún elemento que muestre la utilización de la prensa en todo momento. Esto nos servirá para determinar si la variable de respuesta que deseamos conocer —la utilización del equipo— se encuentra en estado estable o aún en estado transitorio. Con dicho propósito incluiremos lo que se conoce como un gráfico dinámico. Para construirlo:

- Corra la simulación a una velocidad lenta para darle tiempo a construir el gráfico antes de que termine la simulación, mientras ésta se encuentra en ejecución, abra el menú **Information** y haga clic en el comando **Dynamic Plot/New**. Al realizar esta selección aparecerá la ventana **Dynamic Plots** con la pestaña **Stats to Plot** activada con las diferentes estadísticas que ProModel recopila de manera automática, se recomienda mover el tamaño y la posición de la ventana hacia un lugar donde no estorbe la vista del modelo.
- Como en este caso deseamos vincular el gráfico dinámico con una localización, haga clic en el botón **Locations**. Luego seleccione la localización “prensa”, y determine la estadística del porcentaje de utilización (**Utilization %**). En seguida cambie a la pestaña **Chart** y verá el comportamiento de la variable a través del tiempo. La gráfica resultante puede ser modificada tal como si se tratara de un gráfico de Excel.

Si desea guardar el gráfico dinámico para utilizarlo en futuras ocasiones, agréguelo a la configuración de la siguiente manera:

Mientras el modelo esté aún en ejecución y con el gráfico creado anteriormente en la posición y dimensiones deseadas, despliegue de nuevo el menú **Information**, abra el submenú **Dynamic Plot** y haga clic en el comando **Configurations**. En ese momento aparecerá una ventana en la que podrá asignar un nombre al gráfico que acaba de crear, y guardarlo para emplearlo en alguna oportunidad posterior (por ejemplo, nos será útil en la solución del ejemplo 5.2). En el campo **Save/Rename As:** pondremos *figura prensa*, haga clic en el botón **Save**.

Una vez guardado el gráfico dinámico podremos activarlo al inicio de la simulación. Para ello modificaremos lo que se conoce como *lógica inicial* del modelo:

- Abra el menú **Build** y elija el comando **General Information**. Enseguida se desplegará en su pantalla el cuadro de diálogo correspondiente (vea la figura 5.22).



Figura 5.22
El cuadro de diálogo General Information.

Este cuadro de diálogo nos permite acceder a una opción para colocar notas que identifiquen el modelo. (Para crearlas, haga clic en el botón **Model Notes**, y para activarlas, despliegue el cuadro de diálogo **Simulation Options [Simulation/Options]** y marque la casilla de verificación **Display Notes**).

El cuadro de diálogo muestra además la ruta de la biblioteca de gráficos que actualmente se está usando en el modelo, y que permite definir las unidades de tiempo y distancia. Por último, en él podemos especificar eventos o características iniciales y finales del modelo. Por ejemplo, es posible desplegar un mensaje de advertencia que anuncie el inicio o el término de la simulación. Sin embargo, por el momento sólo activaremos el gráfico dinámico al comienzo de la simulación. Siga estos pasos:

- Haga clic en el botón **Inizialization Logic** para desplegar la ventana de programación correspondiente (vea la figura 5.23).



Figura 5.23
Ventana de programación para la lógica inicial del modelo.

Esta ventana es similar a la de **Operation** (vea la figura 5.7). En ella colocaremos la instrucción **DYNPLOT "nombre del gráfico"**. Para nuestro ejemplo escribiremos: **DYNPLOT "figura prensa"**. Una vez introducida la instrucción, cierre la ventana. Si ejecuta en este

momento la simulación, el gráfico dinámico aparecerá desde el inicio y mostrará su periodo transitorio, lo cual le permitirá observar si la variable graficada está estable o no.

Durante la ejecución es posible que el gráfico oculte la simulación de nuestro sistema. Para evitarlo podríamos modificar el tamaño del gráfico, aunque con ello sacrificaríamos su calidad. Otra solución consiste en definir una vista donde el sistema se muestre alineado hacia el lado contrario a donde aparece el gráfico dinámico. Para lograrlo:

- Primero dimensionaremos el sistema actual. Detenga la simulación y abra el menú **View** y haga clic en el comando **Views**.
- Enseguida escriba un nombre específico para la vista en **View Name**, y haga clic en la opción **Add** del cuadro de diálogo. Ahora el botón **Set View** del menú **View** está disponible para seleccionar la vista que acaba de definir. Haga clic en él.
- Para desplegar la vista, abra el menú **View**, elija el submenú **Views** y haga clic en el nombre de la vista que definió en el paso anterior. Otra opción es ejecutar la vista mediante el método abreviado de teclado que aparece al lado de su nombre, el cual se compone de la tecla Ctrl y un número que corresponde al número de la vista, en este caso **Ctrl+1**.

Puesto que la vista deberá ser activada al inicio de la simulación, regrese a la ventana **Iniciation Logic** y escriba en el siguiente renglón después de la instrucción del gráfico dinámico: **VIEW "Nombre de la vista"**. De esta manera, la vista se activará al comienzo de la simulación, al igual que el gráfico dinámico. El resultado de estas acciones podría verse como se ilustra en la figura 5.24. La sintaxis general de la instrucción VIEW es:

VIEW "nombre de la vista"

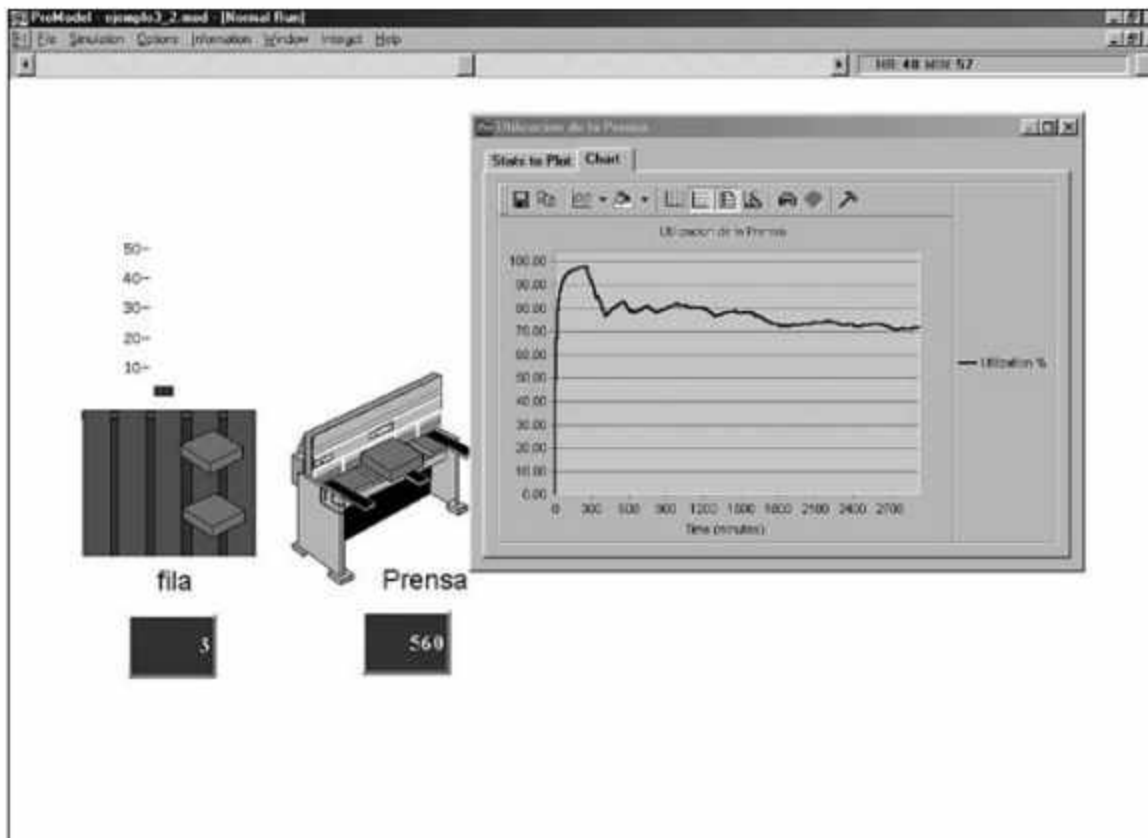


Figura 5.24
Ejemplo 5.2
terminado.

Como pudimos ver en este ejemplo, incrementar el potencial gráfico de nuestro modelo es relativamente sencillo. Pero las herramientas de ProModel no sólo están destinadas a mejorar la interpretación visual del modelo; también nos permiten integrar muchos otros elementos con propósitos distintos, por ejemplo: la tasa de descomposturas de un equipo, el tiempo que tarda en repararse, la probable generación de piezas defectuosas en el proceso, los niveles de retrabajo o de utilización de los recursos, etcétera. Es evidente, que, cuanto más apegado a la realidad queramos que resulte nuestro modelo, más cantidad de ajustes tendremos que hacer; ésta es la razón por la que un buen modelo de simulación toma tiempo en ser construido. No obstante, es preciso tomar en cuenta, en todo momento, que de nada sirve un modelo perfecto gráficamente si no se tienen buenos datos estadísticos de entrada.

5.4.3 Modelado de un sistema que incluye más de un proceso

Ejemplo 5.3

Dos tipos de piezas entran a un sistema. La primera es un engrane que llega a una estación de rectificado donde se procesa por 3 ± 1 minutos; la distribución de probabilidad asociada a las llegadas de este engrane a la fila de la rectificadora es una distribución normal con tiempo promedio de 13 minutos y desviación estándar de 2 minutos. La segunda pieza es una placa de metal que llega a una prensa con una distribución de probabilidad exponencial con media de 12 minutos. La prensa procesa un engrane cada 3 minutos con distribución exponencial. Al terminar sus procesos iniciales, cada una de estas piezas pasa a un proceso automático de lavado que permite limpiar 2 piezas a la vez de manera independiente; este proceso, con distribución constante, tarda 10 minutos. Finalmente, las piezas se empaacan en una estación que cuenta con 2 operadores, cada uno de los cuales empaaca un engrane en 5 ± 1 minuto y una placa en 7 ± 2 minutos. Se sabe que los tiempos de transporte entre las estaciones son de 3 minutos con distribución exponencial. No hay almacenes entre cada proceso: sólo se tiene espacio para 30 piezas antes de la prensa y 30 antes de la rectificadora. Suponga que cada día de trabajo es de 8 horas. Simule este sistema por 40 días, indique el momento en que se inicia y se termina la simulación.

Esquematación inicial del modelo

Antes de comenzar a definir el modelo en ProModel, es conveniente analizar el problema. El primer paso consiste en esquematizar el sistema, como se muestra en la figura 5.25.

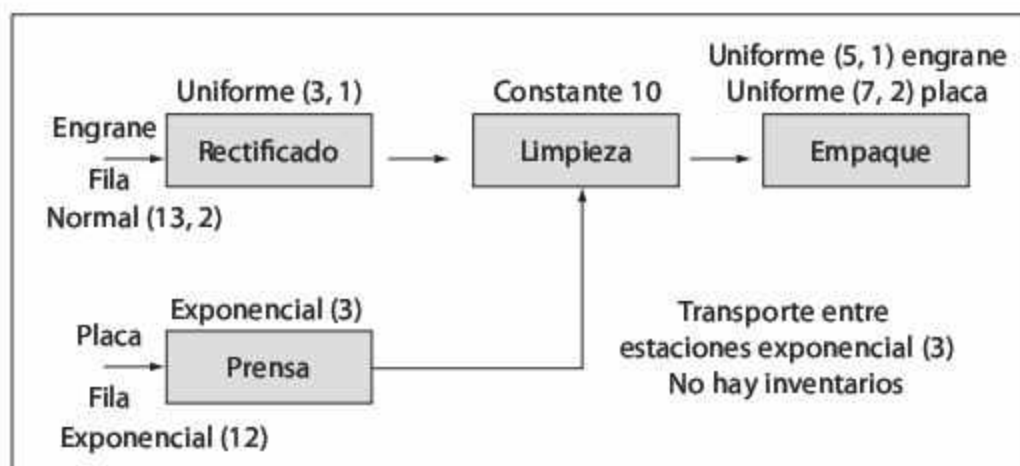


Figura 5.25
Esquema del sistema a modelar en el ejemplo 5.3.


Definición de localizaciones

Recordemos que el modelado en ProModel comienza por la definición de las localizaciones físicas de nuestros procesos, en este caso:

1. La fila de llegada para la rectificadora, con capacidad para 30 piezas.
2. La fila de llegada para la prensa, con capacidad para 30 piezas.
3. El proceso de rectificado, con capacidad para una pieza.
4. El proceso de prensado, con capacidad para una pieza.
5. El proceso de limpieza, con capacidad para limpiar dos piezas de manera independiente.
6. El proceso de empaque, en el que participan dos operadores independientes.

En este modelo aparece un nuevo tipo de localización, ya que debemos definir filas de entrada. En muchos sistemas se tienen bandas o transportadores que se encargan de desplazar las piezas de un proceso a otro; en otros casos, como el de las instituciones bancarias, hay solamente una fila para atender al cliente. ProModel permite simular estos detalles.

Por ejemplo, para definir una fila:

- Abra el menú **Build** y elija **Locations**.
- Seleccione el icono que parece una escalera horizontal () en la ventana **Graphics**, y haga clic en la posición de la ventana **Layout** donde quiere que aparezca la fila de rectificado. Luego, al mover el cursor del ratón, una flecha indicará que está definiendo la fila; colóquese en el lugar donde quiere que termine la fila haga doble clic. Es importante mencionar que si sólo hace un clic en la posición final, seguirá construyendo la misma fila; esta característica es muy útil para definir en una sola localización bandas o transportadores que pasen por toda la planta o por varios procesos.

Podría ocurrir que al definir nuestra fila el icono apareciera como una banda de rodillos más que como una fila; sin embargo, es importante que el modelo sepa que se ha definido una fila y no una banda, pues al momento de la simulación trata cada elemento de manera diferente. Le recomendamos que consulte la ayuda de ProModel para conocer todas las características que se pueden asignar a una fila y a una banda. Por lo pronto, una buena forma de asegurarse de que la localización es una fila (**queue**) y no una banda (**conveyor**), es con un doble clic en ella desde la ventana **Layout**; enseguida se desplegará el cuadro de diálogo **Conveyor / Queue** (vea la figura 5.26). En este cuadro podrá controlar varias características de la localización. Haga lo siguiente:

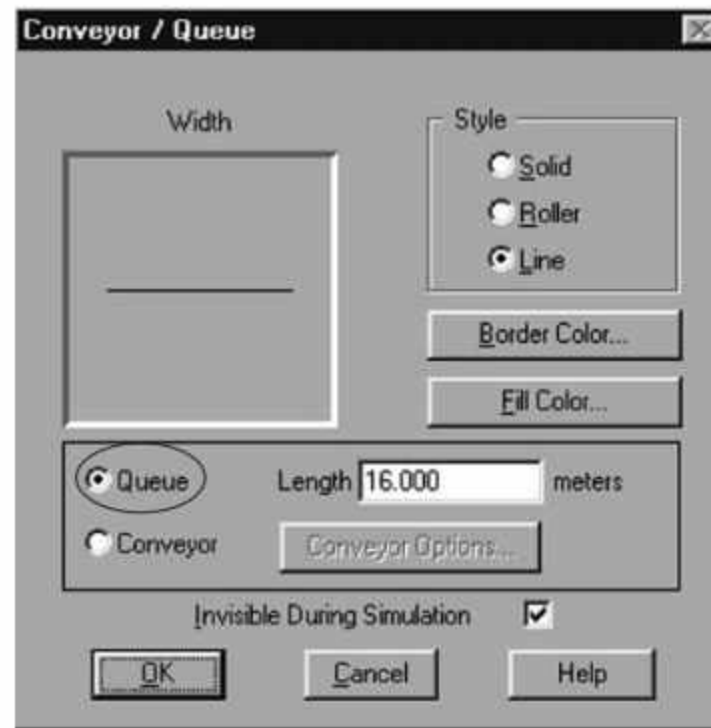


Figura 5.26
Definición de las características de una fila.

- Asegúrese de que esté marcada la casilla de verificación de la opción **Queue**.
- Recuerde que el icono es sólo una representación visual, así que puede decidir, en la sección **Style**, si el icono aparecerá sólido (**Solid**), como banda de rodillos (**Roller**), o sólo como una línea (**Line**).
- Si desea que el icono no aparezca al momento de ejecutar la simulación, marque la casilla de verificación de la opción **Invisible During Simulation**.
- Además de estas características, el cuadro de diálogo permite definir otras, como el color de borde y de relleno del icono (mediante los botones **Border Color** y **Fill Color**) y su longitud (**Length**), en metros.
- Al terminar de definir las características de la fila de rectificado, haga clic en el botón **OK**.

Repita el procedimiento para determinar la localización de la prensa. (Recuerde que ambas filas tienen una capacidad de 30 piezas solamente.) Luego defina la prensa y la rectificadora de la misma manera que definimos otros procesos en los ejemplos anteriores.

A continuación definiremos la lavadora, seleccionando para ello el icono que deseamos que la represente. Para una mejor visualización, coloque 2 iconos de posicionamiento sobre el icono que representará a la lavadora (recuerde que ésta tiene capacidad de limpiar dos piezas a la vez y de manera independiente).

Por último, defina los operadores de ensamble. De acuerdo con la descripción, en el proceso participan dos operadores que realizan la misma operación, pero de manera independiente. Observe que, en el caso de la lavadora, un mismo equipo tiene capacidad para realizar 2 procesos de lavado, mientras que ahora tenemos dos operarios que realizan la misma operación de empaque.

Para definir esto en el modelo podemos proceder de dos maneras. La primera consiste en especificar a cada operador de empaque como una nueva localización; sin embargo,

a nivel de programación tendremos que determinar rutas de entrada y salida para cada uno de ellos. La segunda opción es más práctica: se trata de establecer que el proceso de ensamble tiene 2 unidades de capacidad, una por cada operador. Para lograrlo:

- Defina una operación de empaque y, al terminar, coloque un 2 en la columna **Units** de la ventana **Locations**. Después de aceptar este cambio aparecerá una segunda localización, idéntica a la que definimos originalmente.

Si desea cambiar de posición dicha localización, hágalo; esto no afectará el modelo. Es importante mencionar que si la definición del proceso implica más de un icono, es posible moverlos todos a la vez si se toman de la línea punteada que aparece en su contorno.

Esta manera de definir localizaciones iguales facilita la programación y permite seguir las tratando de manera independiente, lo cual resulta muy útil cuando queremos simular, por ejemplo, un banco con 10 cajeros que realizan las mismas operaciones.

Al terminar estas definiciones, el modelo se verá como se muestra en la figura 5.27.

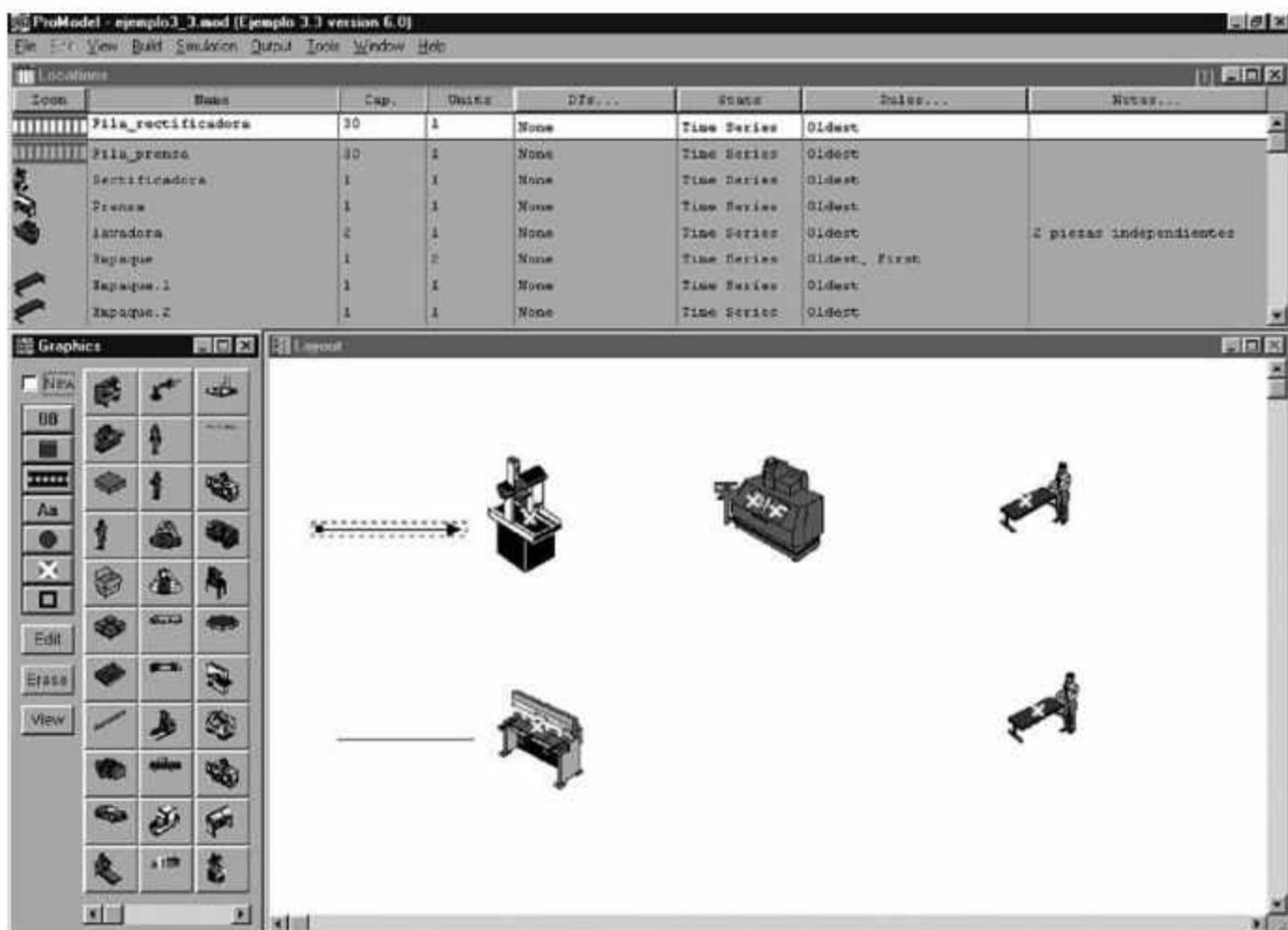


Figura 5.27 Definición de localizaciones para el ejemplo 5.3.

Observe que, aunque las filas aparecen como líneas en el modelo, en la ventana **Locations** siempre lucirán como bandas de rodillos.

Definición de entidades

El siguiente paso en la construcción de nuestro modelo será la definición de las entidades. Para ello es necesario desplegar la ventana apropiada mediante el comando **Entities** del menú **Build**. En este problema será necesario definir dos entidades: una que represente el engrane y otra que represente la placa.

Comenzaremos por definir el engrane seleccionando el gráfico para dicho propósito. Como se mencionó en el ejemplo 5.1, para mejorar el aspecto visual podemos agregar una segunda gráfica. Después repetimos el mismo proceso para la placa. Observe que en la parte inferior de la ventana **Graphics** aparecen unas dimensiones bajo el concepto **Conveyor Only**: estas dimensiones son las que tomaría la pieza si entrara a alguna localización definida como banda y no como fila. Uno de los problemas que podrían presentarse al momento de simular el modelo, es que estas dimensiones sean demasiado grandes, lo que ocasionaría que las piezas no pudieran ser contenidas en la banda. Al momento de definir el modelo es importante considerar que el simulador tomará en cuenta las dimensiones físicas de las piezas definidas en la opción **Conveyor Only** en caso de entrar a una banda; sin embargo, las ignorará cuando entren a una fila.

Definición de llegadas

El siguiente paso en la construcción del modelo es la definición de los arribos o llegadas de las piezas al sistema; para ello, abra el menú **Build** y haga clic en el comando **Arrivals**.

Al especificar los parámetros, recuerde que las llegadas de los engranes tienen distribución normal con media de 13 minutos y desviación estándar de 2 minutos, mientras que las de las placas tienen distribución exponencial con media de 12 minutos. Una vez definidas ambas llegadas, la ventana **Arrivals** deberá lucir como se muestra en la figura 5.28.

Entity...	Location...	Qty Each...	First Time...	Occurrences	Frequency	Legis...	Visible
Engrane	Fila_rectificadora	1	0	inf	N(13,2)		No
placa	Fila_prensa	1	0	inf	E(12)		No

Figura 5.28 Definición de llegadas de las entidades (ejemplo 5.3).

Definición del proceso: uso de la opción View Text

A continuación definiremos la lógica de procesamiento de la simulación. Para ello ejecute el comando **Processing** del menú **Build**.

Para programar las operaciones y rutas de ambas entidades (engranes y placas), procederemos como se indicó en el ejemplo 5.1, pero primero es conveniente tener un esquema del proceso secuencial de cada una de ellas. Es en este tipo de situaciones donde herramientas como los diagramas de operación resultan útiles para realizar una programación más eficiente.

Uso de la instrucción DISPLAY

A partir de los pasos que hemos seguido hasta este momento, el modelo deberá poder ejecutarse sin problemas. Sin embargo, aún no hemos incluido el mensaje de inicio y de fin de la simulación que se nos pidió. Para hacerlo:

- Abra el menú **Build** y haga clic en el comando **General Information** para desplegar el cuadro de diálogo correspondiente (vea la figura 5.22).
- Haga clic en el botón **Initialization Logic** y, en la ventana que aparece, escriba la instrucción **DISPLAY "Inicio de la Simulación"**. Esta instrucción desplegará una ventana de mensaje que detendrá la simulación hasta que hagamos clic en uno de los botones incluidos en ella: si hacemos clic en **Cancel**, la simulación no se ejecutará; si hacemos clic en **OK** la simulación comenzará.

Una vez que haya programado el mensaje de inicio, deberá hacer lo propio con el mensaje de finalización de la simulación.

- Vuelva a desplegar el cuadro de diálogo **General Information**, y ahora haga clic en el botón **Termination Logic**.
- Cuando se abra la ventana **Termination Logic**, coloque nuevamente el comando **DISPLAY**, pero esta vez con un mensaje de fin de la simulación. (Recuerde colocar el texto entre comillas dobles.)

La instrucción **DISPLAY** es muy útil para programar mensajes de alerta dentro de la simulación, o para realizar interacción con el usuario del modelo. Sin embargo, tiene el inconveniente de que detiene la simulación, por lo que es importante utilizarla únicamente cuando el mensaje sea relevante. Por otro lado, si el programador desea colocar comentarios dentro de la programación, puede hacerlo en los espacios reservados para las notas del modelo. Además, si se considera necesario, es posible usar comentarios en la programación de las operaciones y rutas de las entidades, mediante cualquiera de las siguientes opciones al comienzo del renglón:

```
// texto de un solo renglón
# texto de un solo renglón
/* texto en varios renglones */. En este caso es necesario definir el inicio del comentario y la finalización del mismo; es por ello que se utilizan dos símbolos.
```

La sintaxis general de la instrucción **DISPLAY** es:

DISPLAY "mensaje" {,<expresión>},

Nuestro modelo se encuentra casi terminado. Sólo nos falta incluir el tiempo que deseamos simular el sistema.

Definición del tiempo de simulación

En el planteamiento del problema se estableció que cada día tiene 8 horas hábiles de trabajo. También se estipuló que el modelo del sistema abarcaría 40 días, por lo que el tiempo total de simulación será de 320 horas. Dé los pasos pertinentes para determinar estos parámetros como sigue:

- Abra el menú **Simulation** y haga clic en el comando **Options**. En el cuadro de diálogo que se despliega, especifique **320 hr.** en el campo **Run Hours**. Tenga cuidado y no lo indique como **40 day**, porque si lo hace el modelo simulará el sistema por 40 días de 24 horas cada uno.

Estamos listos para guardar y ejecutar el modelo. Abra el menú **Simulation** y haga clic en el comando **Save & Run**. Verifique que el modelo se ejecute sin problemas.

Entidades que no pudieron entrar al sistema

Una de las problemáticas que pueden presentarse al momento de modelar un sistema, radica en que la capacidad de las localizaciones de llegada resulte insuficiente para recibir todas las piezas que arriban al sistema. Cuando esto ocurre, ProModel genera, al final de la simulación, un mensaje de advertencia como el que se ilustra en la figura 5.30 (en español, el mensaje dice "¿Quiere ver los resultados?") (*Nota: Se presentaron fallas en la llegada de entidades debido a capacidad insuficiente*).

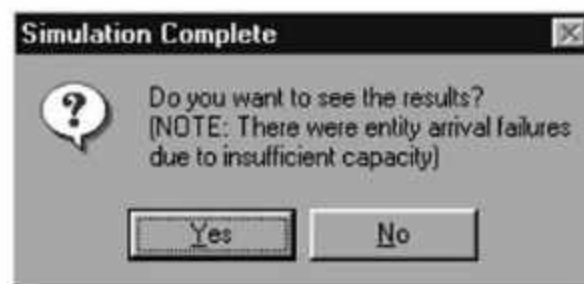


Figura 5.30
Aviso de llegadas fallidas.

En este ejemplo se espera que se presente esta situación. Es posible, sin embargo, que no sea así. Todo depende de la computadora que se esté usando, y también del número de veces que se ejecute la simulación, puesto que al ejecutarse en repetidas ocasiones, los números aleatorios que se utilizan para el modelo cambian, y a su vez modifican los resultados finales.

Observe que el mensaje de la figura anterior no estipula cuántas entidades no pudieron ser simuladas. Para conocer el dato preciso, consulte la información de la ficha **Failed Arrivals** en el reporte de resultados, donde se mostrará el número de piezas que no pudieron entrar al sistema.

En cualquier caso, a fin de evitar la ocurrencia de este tipo de problema se sugiere al lector que cambie la capacidad de las filas de entrada. Dicho incremento no disminuye de manera lineal el número de piezas que no pudieron ser simuladas. Incluso si se incrementa 300% la capacidad actual de las filas, el modelo seguirá presentando entidades que no pudieron ser simuladas. El analista debe evaluar si es mejor tener filas más grandes —que ocupan más espacio— o modificar algunos de los procesos, de manera que las piezas puedan fluir mejor por el sistema.


5.4.4 Inclusión de gráficos de fondo en el modelo

En algunos casos podría ser necesario modelar procesos sobre un plano real de una planta o de un área de trabajo, en especial con miras a mejorar la presentación visual de la simulación. En ProModel esto es posible por medio de un cambio del gráfico de fondo del modelo. Veamos cómo funciona esto en el ejemplo siguiente.

Ejemplo 5.4

Tome como base el ejemplo 5.1, modifique el fondo de la simulación y agregue un código de colores a la prensa, para saber cuándo está trabajando y cuándo se encuentra ociosa. Simule este sistema por 40 días.

Para comenzar, haremos las modificaciones pertinentes en las localizaciones: agregue un código de colores a la prensa, para identificar sus periodos activos e inactivos:

- Abra el menú **Build** y haga clic en el comando **Locations**.
- Seleccione el icono de la prensa en la ventana **Layout**.
- Desmarque la casilla de verificación de la opción **New** en la ventana **Graphics**.
- Seleccione el icono del punto azul  en la ventana **Graphics**, y arrástrelo hasta colocarlo a un lado de la prensa.

El icono cambiará de color automáticamente durante la simulación, lo que le indica el estado de la prensa: será azul si está ociosa, verde si está realizando alguna operación, y rojo cuando no esté disponible (en este caso, cuando se presente el evento del mantenimiento preventivo).

El siguiente paso consiste en cambiar el fondo de la simulación. Podemos hacerlo de dos maneras; la primera consiste en modificar así el fondo del área de trabajo:

- Abra el menú **View**, elija el submenú **Layout Settings**, y haga clic en el comando **Background Color** (vea la figura 5.31).
- Seleccione el color que desee en la ventana que se despliega, y haga clic en el botón **OK**; el cambio se reflejará de inmediato en el área de trabajo.



Figura 5.31 Siga esta ruta para cambiar el color de fondo del área de trabajo de la simulación.

El otro método para modificar el fondo del área de simulación nos permite, además, colocar una imagen con formato BMP, WMF, GIF o PCX. Esto facilita la importación de archivos de AutoCad, por ejemplo, permite trabajar sobre el plano de una planta real. En

este caso es recomendable colocar el fondo antes de comenzar a definir las localizaciones, ya que de otra manera éstas podrían quedar desfasadas de su lugar si se hiciera una modificación de tamaño al plano, lo cual exigiría invertir más tiempo en su reubicación.

Para poder insertar el archivo gráfico como fondo:

- Abra el menú **Build**, elija el submenú **Background Graphics**, y haga clic en el comando **Behind Grid** (vea la figura 5.32).



Figura 5.32 Primer paso para la inserción de una imagen de fondo en el área de trabajo.

Enseguida se desplegará en la pantalla una interfaz gráfica que nos permite insertar imágenes de fondo, o incluso diseñar nuestros propios fondos para el área de trabajo (vea la figura 5.33). Sin embargo, en este ejemplo ilustraremos sólo cómo insertar una imagen con formato BMP.

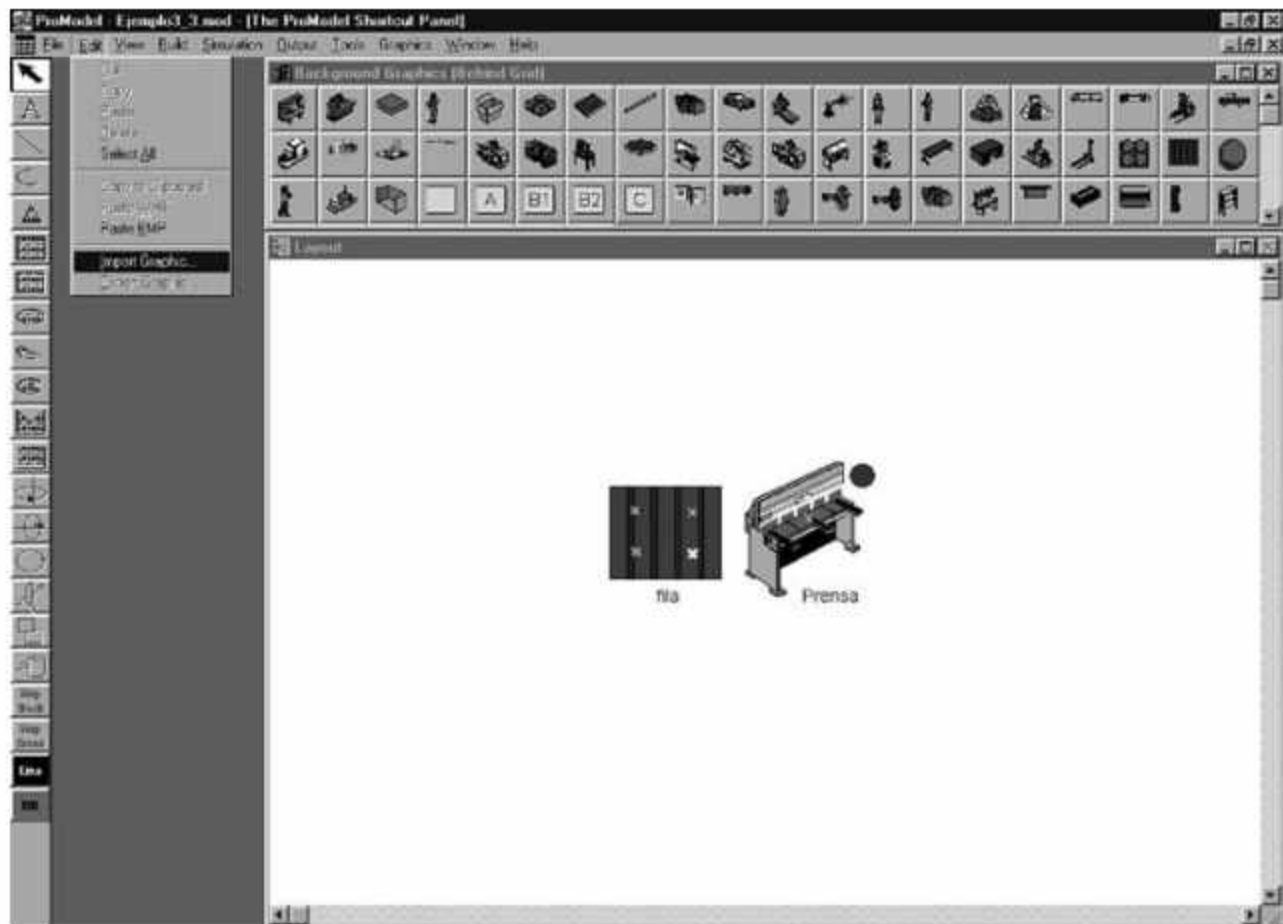


Figura 5.33 Importación de una imagen de fondo a la simulación.

Una vez que entre al área de importación y generación de imágenes de fondo:

- Abra el menú **Edit** y elija **Import Graphic**. A continuación se desplegará un cuadro de diálogo **Open**, simile al habitual en los programas de plataforma Windows.
- Localice el archivo que le interese en la unidad y carpeta de su computadora en donde esté almacenado (en ProModel los formatos predeterminados para importación son BMP y WMF, aunque admite otros formatos). Selecciónelo y haga clic en el botón OK. El archivo gráfico se abrirá de inmediato como fondo de la simulación (las imágenes importadas generalmente aparecen en la esquina superior izquierda del área de trabajo).

En ocasiones el gráfico importado resulta muy pequeño, por lo que tendrá que agrandarlo manualmente. Para ello, selecciónelo, coloque el cursor del ratón en una de sus esquinas, haga clic y, sin soltar el botón del ratón, arrastre hasta lograr el tamaño deseado. Es importante señalar que, en el caso de los archivos BMP en particular, un incremento de tamaño implica también mayor distorsión de la imagen.

Una vez completada la importación podríamos ver un fondo como el que se muestra en la figura 5.34. En este caso se colocó un gráfico predeterminado de Windows como fondo del sistema.

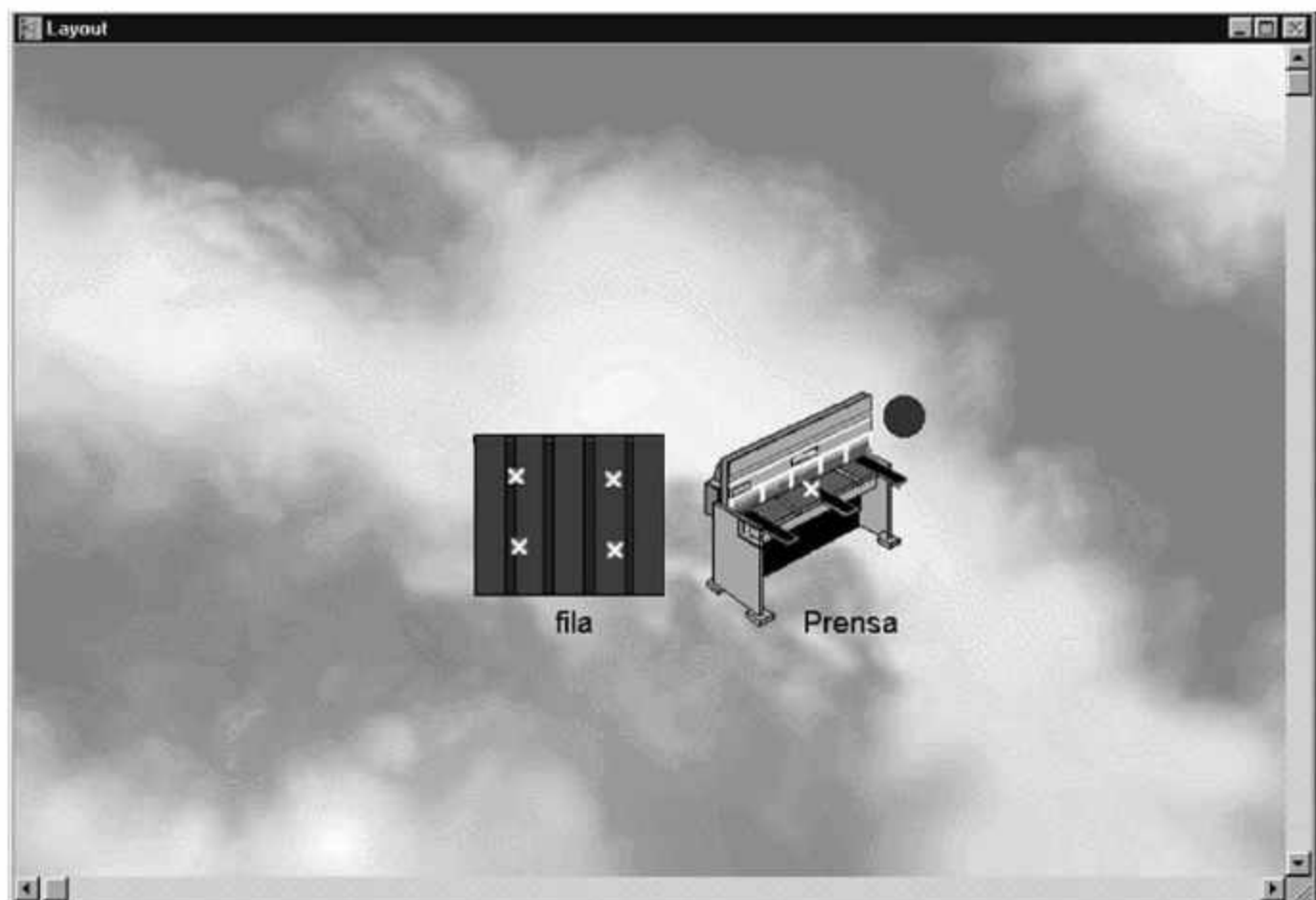


Figura 5.34 El fondo de esta simulación utiliza un archivo gráfico en formato BMP para mejorar su presentación.

Con esto concluimos las modificaciones visuales requeridas en el ejemplo 5.4.

5.5 Arribos cíclicos

En los ejemplos anteriores se ha utilizado el módulo **Arrivals** para simular la llegada de entidades a cada sistema. En estos ejemplos se ha supuesto que la frecuencia de llegadas no cambia durante todo el tiempo de simulación, de esta manera en el ejemplo 5.1 las entidades llegarán siempre de una en una (**QTY each... : 1**) con un tiempo entre arribos exponencial de 4 minutos (**Frequency : E(4)**), como se muestra en la figura 5.5. De manera similar en el ejemplo 5.2 tanto la llegada de los engranes como la de las placas ocurrirá de una en una con un tiempo entre arribos $N(13,2)$ y $E(12)$ respectivamente en el transcurso de la simulación (ver figura 5.28).

En muchos sistemas la llegada de las entidades sigue un comportamiento cíclico. Algunos ejemplos son las llegadas de pacientes a un consultorio, la llegada de autobuses a una parada, la llegada de alumnos a un salón de clases, los aviones a un aeropuerto o los clientes a un restaurante. Estos arribos cíclicos se definen en el módulo **Arrival Cycles (Build / More Elements / Arrivals Cycles)**. Las llegadas pueden ser definidas como un porcentaje o como una cantidad, y puede hacerse en formato acumulativo o no acumulativo. Una vez definido el ciclo de arribos, puede ser asignado en el campo **Qty each...** dentro de **Arrivals (Build / Arrivals)**.

Ejemplo 5.5

A una clínica llegan todos los días a consulta un promedio de 70 pacientes con distribución Poisson. Los registros históricos muestran el siguiente patrón de llegadas:

De:	A:	Porcentaje
6:00	7:00	30
7:00	9:30	10
9:30	12:00	10
12:00	13:00	10
13:00	15:00	5
15:00	19:00	35

El tiempo de consulta sigue una función de densidad uniforme entre 25 y 35 minutos. Se dispone de 3 doctores para las consultas. Corra el modelo de simulación durante treinta días para encontrar el tiempo promedio de espera de un paciente antes de ser atendido.

Defina en el ventana **Locations (Build/Locations)** dos localizaciones: *Fila* y *Doctores*, y en la ventana **Entities (Build/Entities)** la entidad *Pacientes* con sus correspondientes características. Enseguida, en la ventana **Process (Build/Processing)** defina la lógica de proceso de acuerdo a lo mostrado en la figura 5.35.

```

*****
*                               Locations                               *
*****
Name      Cap      Units  Stats      Rules      Cost
-----
Doctores  1         3      Time Series Oldest, , First
Doctores.1 1         1      Time Series Oldest, ,
Doctores.2 1         1      Time Series Oldest, ,
Doctores.3 1         1      Time Series Oldest, ,
Fila      INFINITE 1      Time Series Oldest, FIFO,

*****
*                               Entities                               *
*****
Name      Speed <fpm>  Stats      Cost
-----
Pacientes 150          Time Series

*****
*                               Processing                             *
*****
                                Process                                Routing
Entity  Location Operation      Blk  Output  Destination Rule  Move Logic
-----
Pacientes Fila
Pacientes Doctores wait U<30,5> min  1    Pacientes Doctores FIRST 1
                                           1    Pacientes EXIT   FIRST 1

```

Figura 5.35 Localizaciones, entidades y lógica de la ruta de los pacientes.

En la figura 5.36 se muestra una imagen del modelo de simulación de la clínica.

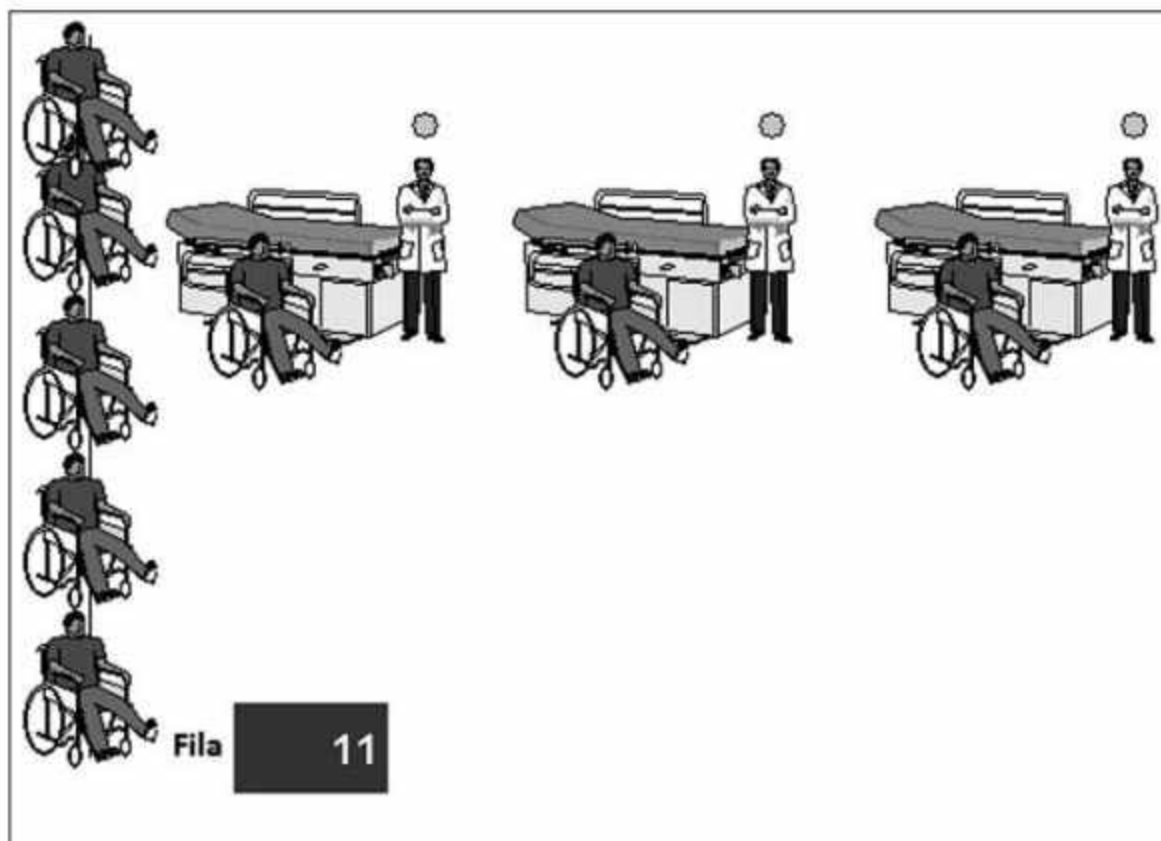


Figura 5.36 Esquematación del ejemplo 5.5.

Para definir el ciclo de arribos activamos el módulo **Arrival Cycles (Build / More Elements / Arrivals Cycles)**, se abre la tabla de edición que se muestra en la figura 5.37. Al ciclo de arribos se le llamará **Llegadas**. Debido a que los datos están expresados en porcentaje, seleccionamos **Percent** en el campo **Qty/%**. Estos porcentajes son no acumulativos de tal forma que especificamos **No** en el campo **Cumulative**.

ID	Qty / %	Cumulative	Table...
Llegadas	Percent	No	Defined

Figura 5.37 Esquemmatización del ciclo de arribos del ejemplo 5.5.

Enseguida activamos el botón **Table...** para activar la ventana de edición del ciclo mostrada en la figura 5.38.

Time (Hours)	Qty / %
1	30
3.5	10
6	10
7	10
9	5
13	35
24	0

Figura 5.38 Esquemmatización del ciclo de arribos.

En este ejemplo, aun cuando los porcentajes de pacientes que llegan a la clínica son no acumulativos, el tiempo siempre será acumulativo. De tal forma que la tabla se lee como sigue: 30% del total de pacientes arriba en la primera hora, 10%, entre la hora 1 y la hora 3.5 de la simulación, y así sucesivamente. Las llegadas se distribuyen de manera uniforme dentro del intervalo en el que el paciente llega.

Una vez definido el ciclo de arribos, ahora puede ser asignado al campo **Qty each...** del módulo de **Arrivals (Build/Arrivals)**. En la ventana de diálogo que se muestra en la figura 5.39 seleccionamos el ciclo de arribos creado anteriormente y la cantidad de pacientes que llegan por ciclo.

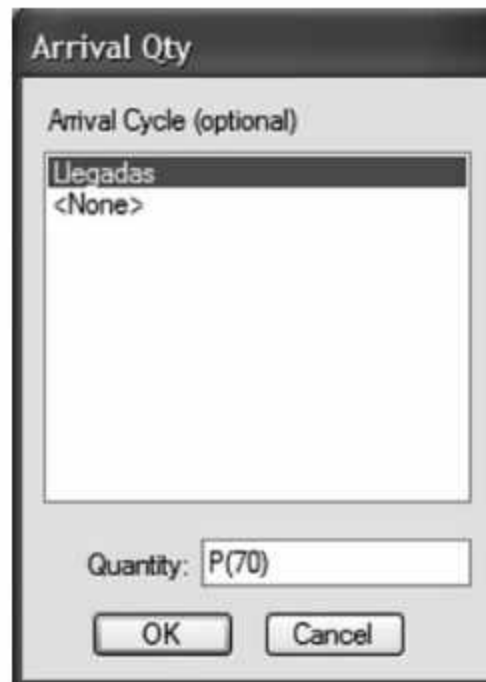


Figura 5.39 Programación del ciclo de arribos dentro del módulo Arrivals.

El módulo de arribos queda como se muestra en la figura 5.40.

Entity...	Location...	Qty Each...	First Time...	Occurrences	Frequency	Logic...	Disabl...
Pacientes	Fila	P(70); Llegadas	0	30	24 hr		No
Ciclo de arribos			Tiempo de simulación		Tiempo entre arribos		

Figura 5.40 Arribos a la clínica.

Al ejecutar este modelo se despliega el resultado de 47.80 minutos de tiempo promedio de espera de los pacientes en fila (figura 5.41).

General Report (Normal Run - Rep. 1)									
Ejemplo 613.mod (Normal Run - Rep. 1)									
Name	Scheduled Time (HR)	Capacity	Total Entries	Avg Time Per Entry (MIN)	Avg Contents	Maximum Contents	Current Contents	% Utilization	
Doctores.1	709.66	1.00	756.00	30.06	0.53	1.00	0.00	53.38	
Doctores.2	709.66	1.00	734.00	29.84	0.51	1.00	0.00	51.44	
Doctores.3	709.66	1.00	695.00	30.00	0.49	1.00	0.00	48.97	
Doctores	2128.99	3.00	2185.00	29.97	0.51	3.00	0.00	51.26	
Fila	709.66	999999.00	2185.00	47.80	2.45	21.00	0.00	0.00	

Figura 5.41 Resultados de la ejecución del modelo por tipo de cliente, con el formato de reporte Viewer 2.0 (3DR).

Si la opción de formato de salida de los resultados dada de alta en **Simulation/Options** es **Viewer 4.0**, el despliegue de los resultados de la ficha **Locations** se verá de una forma similar al mostrado en la figura 5.42.

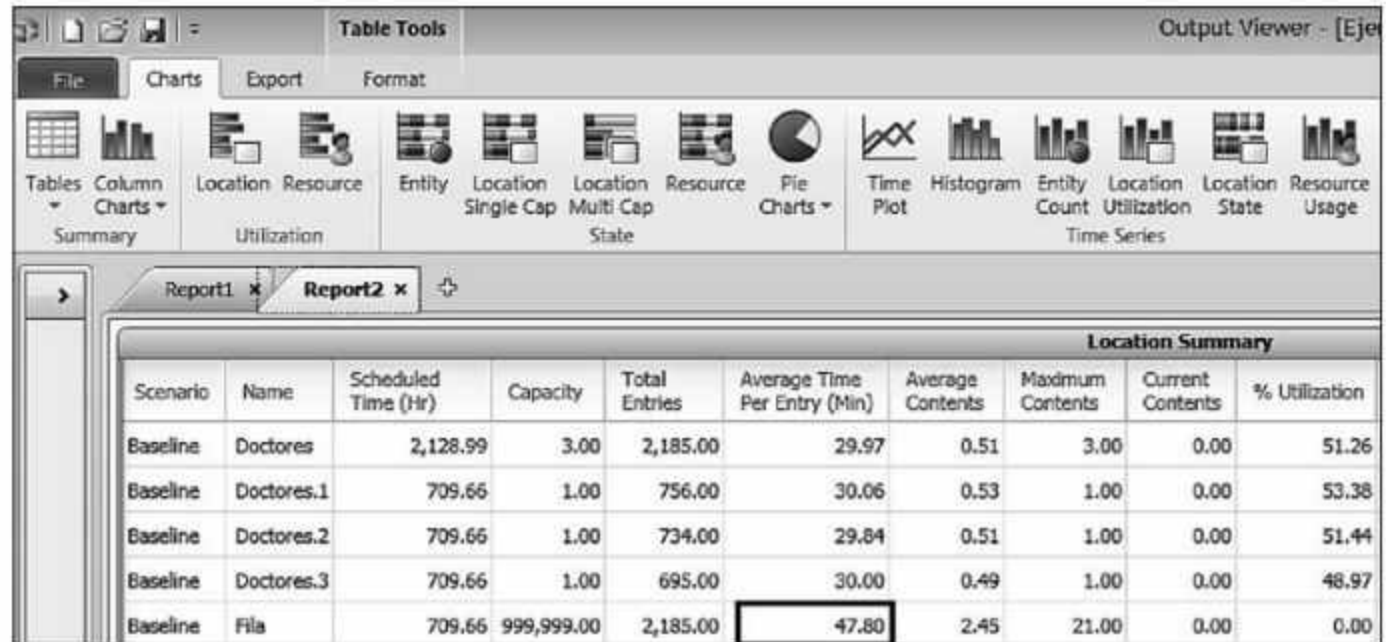


Figura 5.42 Resultados de la ejecución del modelo por tipo de cliente, con el formato de reporte Viewer 4.0.

5.6 Caso integrador

Se tiene una línea de empaque a la que llegan piezas cada 2 minutos con distribución exponencial. Esta línea cuenta con cinco procesos, que se describen a continuación:

1. **Recepción de materiales.** Cuenta con un espacio ilimitado de almacenamiento. En este lugar se reciben las piezas que llegan al sistema, y luego éstas pasan a un proceso de lavado. El traslado de las piezas de una estación a otra es de 3 minutos con distribución exponencial.
2. **Lavado de la pieza.** La lavadora tiene capacidad para limpiar 5 piezas a la vez. El tiempo de proceso de cada pieza se distribuye normalmente con media de 10 minutos y desviación estándar de 2 minutos. De aquí pasan a un proceso de pintura, antes del cual llegan a un almacén con capacidad para un máximo de 10 piezas. El tiempo de traslado entre estas estaciones es de 2 minutos con distribución exponencial.
3. **Pintura.** En el área de pintura se tiene capacidad para pintar 3 piezas a la vez. El tiempo de pintado tiene una distribución triangular de (4, 8, 10) minutos. Posteriormente las piezas pasan a un horno, el cual cuenta con un almacén que tiene capacidad para 10 piezas. El tiempo de transporte entre estos procesos está uniformemente distribuido con un límite inferior de 2 minutos y uno superior de 5 minutos.
4. **Horno.** En el horno se seca la pintura. El horno sólo puede procesar una pieza a la vez. La duración del proceso es de 3 ± 1 minutos. De aquí son transportadas a dos mesas de inspección visual. No existe un almacén entre el horno y las mesas de inspección. El tiempo de transporte entre estas estaciones es de 2 ± 1 minutos.

5. **Inspección.** En cada mesa hay un operario que realiza la inspección de 3 elementos en cada pieza. La revisión de cada elemento tarda 2 minutos con distribución exponencial. Al finalizar este proceso, las piezas salen del sistema.

Realice lo siguiente:

- a) Simule el sistema por 90 días de 24 horas cada uno.
- b) Ejecute 3 réplicas de la simulación.
- c) Analice el archivo de resultados del modelo.
- d) Obtenga un intervalo de confianza para el número de piezas producidas.
- e) Determine, en una tabla, las utilizaciones de todas las localizaciones del modelo.

Análisis del modelo

Cada una de las siguientes preguntas es independiente, y tienen como base el modelo original. Respóndalas con base en el análisis de sus resultados.

1. ¿Dónde se encuentra el cuello de botella de este sistema?
2. ¿Qué sugerencias haría para mejorar el sistema?
3. El hecho de que una entidad se encuentre en estado de bloqueo significa que la pieza ha terminado sus operaciones en la localización actual pero no puede avanzar a la siguiente, puesto que no hay espacio para colocarla. De acuerdo con esto, ¿considera que es grave el problema de bloqueo de las piezas? ¿En qué localizaciones? ¿Qué se puede hacer para mejorar la situación? Haga los cambios que considere necesarios al modelo, y ejecútelo nuevamente para determinar la mejora porcentual respecto del número de piezas terminadas.
4. Si pudiera lograr una mejoría de 10% en el tiempo de proceso de alguna de las estaciones, ¿en cuál de ellas sería y por qué?
5. ¿Es necesario que alguno de los almacenes sea más grande? ¿Cuál y por qué razones?
6. ¿Considera necesario colocar un almacén entre el horno y las mesas de inspección? ¿De qué capacidad?
7. Cada pieza deja una utilidad de \$5 y ninguna de las inversiones debe recuperarse en más de 3 meses. ¿Cuál sería su recomendación si se está analizando la posibilidad de comprar otro horno con la misma capacidad y que cuesta \$100,000?
8. ¿Cuál sería su recomendación si lo que se desea comprar es otra lavadora de la misma capacidad y con un costo de \$100,000?
9. ¿Valdría la pena contratar otro operario para la inspección? El costo de esta operación es de \$50,000.
10. Con base en su conocimiento del sistema, haga combinaciones de los incisos anteriores y trate de obtener la mayor cantidad de piezas con el mínimo costo de inversión.

5.7 Problemas

1. Los espectadores llegan a un estadio de beisbol cada 2 ± 1 segundos y hacen cola para entrar. El tiempo que se requiere para pasar por la puerta giratoria del estadio es 5 ± 3 segundos. Modele este sistema y simule el paso de 300 personas por la puerta. Determine la longitud promedio de la fila y la utilización de la puerta.
2. Un muelle cuenta con una grúa para descargar barcos. El tiempo de descarga es de 2 ± 1 días y los barcos llegan de uno en uno a una tasa promedio de 6 barcos cada 14 ± 2 días. Si un barco llega y la grúa está ocupada, espera en una línea para ser descargado posteriormente. Simule un año y determine el tiempo promedio que transcurre desde que un barco llega al sistema hasta que termina su descarga y la longitud máxima de la fila.
3. A una máquina de autolavado llegan coches con un tiempo entre arribos de 5 ± 5 minutos/auto. El tiempo de lavado es de 4.5 ± 1 minuto/auto. Frente a la máquina hay un techo que proporciona sombra a los 4 primeros automóviles de la fila. Haga 100 réplicas de este sistema, de 8 horas cada una, y determine el número promedio de autos haciendo fila en la sombra y haciendo fila en el sol.
4. Una tienda emplea a un dependiente para atender a sus clientes. El tiempo entre arribos es de 5 ± 4 minutos/cliente. El dependiente tiene que cobrar y empaclar los artículos comprados. El primer proceso consume 1 ± 0.5 minutos y el segundo, 3.5 ± 2 minutos. Simule hasta que asegure que el sistema llegue a estado estable, use un gráfico dinámico de la utilización del dependiente y calcule: a) la utilización del dependiente y b) el tiempo promedio de espera en la fila.
5. Una gasolinera tiene una bomba y una fila con capacidad para n autos donde puedan esperar antes de ser atendidos. El tiempo entre arribos de los vehículos es de 3 ± 1 minutos/auto. El tiempo para dar servicio es de 5 ± 2 minutos. Elabore un modelo en ProModel con sólo dos localizaciones, una que simule la bomba y otra, la fila de tamaño finito. Ejecute el modelo durante 100 horas con $n = 1, 2, 3, 4, 5$ e indique en cada caso el número total de autos que no pudieron entrar en la gasolinera por falta de espacio. ¿Cuál es el número mínimo de espacios que se deberían tener para asegurar que no hubiera rechazos de autos?
6. Un centro de maquinado recibe tres diferentes tipos de piezas. Antes del centro existe un almacén de producto en proceso, con capacidad prácticamente infinita. El tiempo de operación y la tasa de entrada de las piezas son las siguientes:

Tipo de pieza	Tasa de entrada (piezas/hr)	Tiempo de maquinado (min/pieza)
1	2	3
2	4	5
3	2	10

- Simule este sistema en ProModel durante 100 horas y determine:
- La utilización del centro de maquinado.
 - Número total de piezas producidas.
 - Tiempo promedio de espera de las piezas en el almacén.
 - Número promedio de piezas en el almacén.
7. A un operario de limpieza se le entregan simultáneamente 60 piezas cada hora. El tiempo de limpieza es de 50 segundos/pieza. Simule el proceso anterior durante 500 horas para determinar:
- La utilización del operario.
 - Tiempo promedio de permanencia de las piezas en todo el proceso.
 - Tiempo promedio de espera de las piezas antes de ser limpiadas.
 - La cantidad total de piezas limpiadas.
8. Al operario de limpieza del problema anterior le entregan ahora una pieza cada minuto. El tiempo de limpieza es de 50 segundos/pieza. Simule el proceso anterior durante 500 horas para determinar:
- La utilización del operario.
 - Tiempo promedio de permanencia de las piezas en todo el proceso.
 - Tiempo promedio de espera de las piezas antes de ser limpiadas.
 - La cantidad de piezas limpiadas.
 - ¿Existen diferencias entre los resultados de este problema y el anterior? ¿A qué atribuye que algunos resultados sean similares y otros no?
9. Un sistema de pintura consta de dos procesos en serie: pintura y horneado, antes de cada proceso hay un almacén de capacidad infinita. El tiempo de pintura es de 10 minutos/pieza, y el tiempo de horneado es de 6 minutos/pieza. Para el proceso hay dos pintores y un horno. La tasa de entrada es de 7 piezas/hora (pieza tipo 1) y de 3 piezas/hora (pieza tipo 2). El tiempo para moverse de un proceso a otro es de 30 segundos. Simule el sistema con 5 días para determinar:
- La utilización de cada operación.
 - El tiempo promedio de permanencia de las piezas en todo el proceso.
 - El tiempo promedio de espera de las piezas antes del pintado y antes del horneado.
10. A un centro de copiado arriban tres tipos de trabajos, cada uno llega individualmente. Si un trabajo no puede iniciarse de inmediato, espera en una fila común hasta que esté disponible alguna de las tres copiadoras. El tiempo de copiado y la tasa de entrada de los trabajos son como en el siguiente cuadro:

Tipo de trabajo	Tasa de entrada (trabajos/hr)	Tiempo de copiado (min/trabajo)
1	4	12
2	8	15
3	16	1

Después del proceso de copiado los trabajos son inspeccionados por un empleado en un tiempo de 3, 6, 10 minutos para los trabajos 1, 2 y 3, respectivamente. Simule el sistema en ProModel durante 50 horas, y determine:

- a) La utilización del empleado y de las copiatoras en la situación propuesta.
- b) Número de empleados y copiatoras mínimos necesarios para asegurar el flujo constante de los trabajos.

11. La aerolínea VFC ha instalado un módulo con tres agentes para la atención de los pasajeros, existen tres tipos de pasajeros. Los tiempos entre llegadas y los tiempos de servicio para cada uno son como en esta tabla:

Tipo de pasajero	Tiempo entre arribos (minutos/pasajero)	Tiempo de servicio (min/pasajero)
A	5.5 ± 2	3 ± 1
B	10.5 ± 5	8 ± 5
C	15.5 ± 10	12 ± 7

Desarrolle un modelo de simulación con animación. Simule durante 1000 horas. Si cada agente atiende sólo a un tipo de pasajero, determine la longitud máxima de cada fila, la longitud promedio de cada una de las filas y la utilización de cada agente.

12. Cierta tipo de pieza entra a una línea de producción; el proveedor entrega las piezas en grupos de 5 cada 10 minutos. La línea consta de 5 operaciones, con una máquina dedicada a cada operación. Los tiempos de proceso son:

Operación	1	2	3	4	5
Tiempo (min/pza)	2	1	0.5	0.25	0.125

El tiempo para moverse entre estaciones es de 0.0625 minutos. La animación debe incluir un contador de las piezas producidas. Simule en ProModel el proceso de 1000 piezas para determinar:

- a) Tiempo total de simulación.
- b) Utilización de cada operación.
- c) Tiempo de espera antes de la primera operación.
- d) Porcentaje del tiempo que la pieza está bloqueada.

13. ELECTRSA fabrica componentes electrónicos. El proceso consta de cuatro estaciones: ensamble, soldadura, pintura e inspección. Antes de cada estación hay almacenes de capacidad para 50 componentes. Los órdenes de trabajo tienen un tiempo entre arribos de 20 ± 14 minutos. Los tiempos de procesamientos en cada estación son:

Operación	ensamble	soldadura	pintura	inspección
Tiempo (min/ pza)	13.5 ± 1.5	36 ± 20	55 ± 15	8 ± 6

Hay un operario en la estación de ensamble, uno en inspección, tres en soldadura y cuatro en pintura. Elabore un modelo de simulación de este sistema de manufactura, simule 100 días de 8 horas cada uno. Determine la utilización de cada estación de trabajo, el número de componentes en promedio que están esperando antes de cada operación y el número promedio de órdenes fabricadas cada 8 horas.

14. A un cajero automático llegan clientes cada 10 minutos con distribución exponencial. El tiempo que tarda cada cliente en hacer sus movimientos bancarios se distribuye exponencialmente con media de 4 minutos. Lleve a cabo lo que se indica a continuación:
 - a) Si se desea que el cajero no tenga más de 5 clientes haciendo fila en un momento determinado, ¿qué recomendación haría al banco? Considere una simulación con 3 réplicas una semana de 40 horas de trabajo cada una.
 - b) Realice un análisis de sensibilidad variando el tiempo promedio de servicio del cajero, y determine cuál es el tiempo máximo que un cliente debe tardar para que la fila de espera no exceda 5 clientes en ningún momento.
 - c) Programe un gráfico dinámico que muestre la utilización del cajero automático durante la simulación. ¿Qué observaciones puede hacer respecto de la gráfica de estabilización generada?

15. A un centro de copiado llegan clientes cada 5 minutos, con distribución exponencial. Ahí son atendidos por un operario con un promedio de servicio de 6 minutos con distribución exponencial. Sólo hay espacio para tres personas en la fila; si llega alguien más, se le envía a otro centro de copiado. Simule el sistema a partir de esta información y determine:
 - a) ¿Cuál es el número promedio de clientes que esperan en la fila?
 - b) ¿Cuál es la utilización del centro de copiado?
 - c) Si cada cliente que se va le cuesta \$5 al centro de copiado, ¿a cuánto asciende la pérdida esperada?

16. En una autopista hay una sola caseta de cobro para automóviles y camiones. El tiempo entre llegadas de autos es 25 ± 10 segundos y el de camiones 60 ± 20 segundos. La caseta de cobro, operada por una sola persona, tiene distintos tiempos de cobro según el tipo de vehículo. El tiempo de cobro para los camiones es de 50 ± 20 segundos y el de los autos es de 20 ± 15 segundos. Realice 10 réplicas de 8 horas y encuentre la utilización del operario y el tiempo de permanencia en el sistema por cada tipo de vehículo.

17. Un centro de servicio cuenta con 3 cajeros. Los clientes llegan individualmente y lo hacen en promedio a razón de 60 por hora con distribución de Poisson. El tiempo promedio que se requiere para atender a un cliente es de 2 minutos con distribución

exponencial. Los clientes hacen una sola fila y no hay límite para su longitud. Haga lo siguiente:

- a) Simule el sistema por 40 horas.
- b) Determine la utilización de los cajeros.
- c) Si el costo de tener a un cliente haciendo fila es de \$5/cliente promedio-hr, determine el costo de operación de este sistema.

18. Un banco está diseñando su área de cajas, y desea determinar cuántas ventanillas debe colocar. Después de hacer un análisis estadístico, la empresa sabe que sus clientes llegan con un tiempo medio de 3 minutos con distribución exponencial; asimismo, se sabe —a partir de información de otras sucursales del mismo banco— que cada cajero tarda un tiempo promedio de 8 minutos con distribución exponencial en atender a un cliente. La empresa planea utilizar una sola fila donde se ubicarían todos los clientes, para después pasar a la primera caja desocupada.

Si el costo de tener un cajero es de \$15 por hora, y además se sabe que por política de la empresa el costo de que un cliente esté esperando a ser atendido es de \$8/cliente-hr, determine:

- a) El número óptimo de cajeros de acuerdo con el costo. Tome en cuenta este número de cajeros, determine también la probabilidad de que el sistema esté vacío, la utilización de los cajeros, el tiempo promedio en el sistema, el tiempo promedio en la fila, y el número promedio de clientes en el sistema.
- b) Realice este mismo análisis considerando que cada caja tiene su propia fila y que las llegadas se distribuyen proporcionalmente al número de cajas; es decir, las llegadas a cada fila son iguales al número total de llegadas, dividido entre el número de cajas.
- c) ¿Cuál de los dos modelos sería mejor y a qué atribuye este resultado?
- d) ¿Cuál sería el costo de utilizar el modelo original de una sola fila y evitar, al mismo tiempo, que 70% de los clientes hagan fila?

19. Un sistema recibe piezas de acuerdo con una distribución uniforme de entre 4 y 10 minutos. Las piezas son colocadas en un almacén con capacidad infinita, donde esperan a ser inspeccionadas por un operario. El tiempo de inspección tiene una distribución exponencial con media de 5 minutos. Después de la inspección las piezas pasan a la fila de empaque, con capacidad para 5 piezas. El proceso de empaque está a cargo de un operario que tarda 8 minutos con distribución exponencial en empaquetar cada pieza. Luego, las piezas salen del sistema.

- a) Simule el sistema por 40 horas.
- b) Identifique dónde se encuentra el cuello de botella.
- c) Genere vistas para cada uno de los procesos por separado.
- d) Incrementar el espacio en el almacén cuesta \$5/semana-unidad de espacio adicional; aumentar 10% la velocidad de empaque cuesta \$15/semana; el costo de incluir otro operario para que se reduzca el tiempo de empaque a 5 minutos con distribución exponencial, es de \$20/semana. Cada unidad producida deja una utilidad de \$0.40. Con base en esta información, determine qué mejoras podrían hacerse al sistema para incrementar su utilidad semanal.

20. A un torno llegan barras cada 3 minutos con distribución exponencial. Ahí se procesan de acuerdo con una distribución normal con media de 5 minutos y desviación estándar de 1 minuto. Posteriormente pasan a un proceso de inspección. La capacidad del almacén previo al torno es de 10 piezas. Por otro lado, a una fresadora llegan placas cada 5 ± 1 minutos. El tiempo de proceso de las placas en la fresadora se distribuye triangularmente (2,4,7). Después, las placas pasan a inspección. La capacidad del inventario antes de la fresadora es de 10 piezas. En el proceso de inspección se cuenta con espacio disponible para almacenar 15 piezas, mientras que el tiempo de inspección es de 4 ± 2 minutos para las barras y de 6 ± 1 para las placas. Tras la inspección, las piezas salen del sistema. Considere un tiempo de transporte entre estaciones de 2 minutos, con distribución constante.
- Simule el sistema por 40 horas para determinar la capacidad mínima de cada inventario, de manera que todas las piezas que lleguen al sistema se procesen y no exista bloqueo por falta de capacidad.
 - ¿Cuál es la diferencia en el número de piezas producidas entre el modelo original y el modelo mejorado?
 - Coloque mensajes de inicio y fin de la simulación.
 - Cambie las gráficas de ambas piezas después de ser procesadas.
21. A una línea de emergencias médicas llegan llamadas normalmente distribuidas con un tiempo medio entre llamadas de 6 minutos, y desviación estándar de 1.5 minutos. El tiempo de atención de cada llamada es de 10 minutos con distribución exponencial. Se desea determinar el número de líneas necesarias para que por lo menos 80% de los clientes no tenga que esperar a ser atendido. Simule el sistema por 7 días de 24 horas cada uno para responder las siguientes preguntas.
- ¿Cuántas líneas telefónicas se necesitan para cumplir con la meta?
 - Si el costo de cada línea es de \$150/semana, ¿cuál es el costo de operación por semana?
 - ¿Cuál es el costo de incrementar el porcentaje de clientes atendidos sin espera a 90%?
 - ¿Cuál sería el número de llamadas en espera que se tendría en el caso a? ¿Cuál sería en el caso c?
 - Si el costo de cada cliente en línea de espera es de \$40/cliente, ¿qué opción sería mejor, a o c?
22. A una empresa llegan piezas con media de 8 minutos y distribución exponencial. Las piezas entran a un almacén con capacidad para 50 unidades, donde esperan a ser procesadas en un torno. Ahí son torneados por 3 minutos con distribución exponencial. El tiempo de transportación del almacén al torno tiene una distribución normal con media de 4 minutos y desviación estándar de 1 minuto. Después, las piezas son transportadas a una estación de inspección donde hay dos operarios, cada uno trabajando de manera independiente. La inspección tarda 6 ± 2 minutos por pieza. El tiempo de transporte entre el torno y los operarios es de 4 ± 1 minutos.
- Simule el sistema por 30 días de 8 horas de trabajo cada uno.
 - Incluya un contador y una gráfica de barra para las piezas en el almacén.

- c) Incluya un indicador de actividad para el turno.
 - d) Realice 3 réplicas y obtenga un intervalo de confianza para el número de piezas que salieron del sistema.
23. Una tienda departamental recibe hombres y mujeres con un tiempo entre arribos de 5 ± 2 minutos y 3 ± 3 minutos respectivamente. Los clientes seleccionan primero la ropa y el tiempo para este proceso es de 20 ± 5 minutos para los hombres y 50 ± 30 para las mujeres. Una vez seleccionada la ropa, los clientes van a la sección de probadores y el tiempo para trasladarse hacia ese lugar es de 40 ± 10 segundos. La tienda tiene separados los probadores para cada género y ha asignado 2 probadores para hombres y 6 para mujeres. Los clientes esperan en fila en caso de que los probadores estén ocupados, considere que hay una fila para hombres y otra para mujeres. El tiempo para probarse la ropa es de 12 ± 7 minutos para los hombres y de 20 ± 8 minutos para las mujeres. Simule este sistema y haga recomendaciones a la tienda, por ejemplo:
- a) ¿Son suficientes los probadores para cada género?
 - b) ¿Agregaría o quitaría probadores? ¿Cuántos?
 - c) ¿Recomendaría que el total de los probadores fueran compartidos por ambos géneros?
24. A las cajas de la cafetería de una universidad llegan 800 estudiantes por día. La tasa de entrada durante el día es variable y se comporta de la siguiente forma:

De:	A:	Porcentaje
6:00	7:00	5
7:00	10:00	20
10:00	12:00	10
12:00	14:00	50
14:00	18:00	5
18:00	19:00	10

- Existen 4 empleados para cobrar y una fila común donde los estudiantes pueden hacer fila antes de ser atendidos. El tiempo de atención es de 45 ± 20 segundos. Simule el proceso anterior durante 30 días. Incluya en la simulación un gráfico dinámico en el que se observe la cantidad de estudiantes en la fila a través del tiempo, y determine:
- a) La utilización de los empleados.
 - b) El tiempo promedio de permanencia de los estudiantes en la fila.
 - c) Con base en los resultados anteriores ¿agregaría más empleados?
 - d) Con base en la gráfica ¿qué recomendaciones haría?

25. Una zapatería tiene un empleado que atiende a dos tipos de clientes: niños y adultos. Diariamente recibe 30 niños y 60 adultos. La tasa de llegada de los niños durante el día es variable y depende de la hora del día de acuerdo con las siguientes tablas:

Tasa de llegada para los niños:

De:	A:	Porcentaje
8:00	12:00	0
12:00	14:00	20
14:00	16:00	20
16:00	18:00	50
18:00	20:00	10

Para los adultos:

De:	A:	Porcentaje
8:00	10:00	10
10:00	12:00	10
12:00	14:00	5
14:00	13:00	5
13:00	18:00	50
18:00	20:00	20

El tiempo de atención es de 4 ± 2 minutos para los niños y 10 ± 4 minutos. Simule el proceso anterior durante 30 días. Incluya en la simulación un gráfico dinámico en el que se observe la cantidad de clientes en la fila a través del tiempo, y determine:

- La utilización del empleado.
- El tiempo promedio de permanencia de los clientes en la fila.
- Con base en los resultados anteriores ¿agregaría más empleados?, ¿cuántos?



Capítulo 6

Instrucciones de ProModel

- 6.1** Uso de la biblioteca de funciones probabilísticas
- 6.2** Recursos
- 6.3** Paros en los equipos
- 6.4** Reglas de ruteo
- 6.5** Ensamblajes, acumulación y agrupamiento de piezas
- 6.6** Transporte entre estaciones
- 6.7** Instrucciones de control
- 6.8** Optimización con Simrunner
- 6.9** Caso integrador 1
- 6.10** Caso integrador 2
- 6.11** Problemas

6.1 Uso de la biblioteca de funciones probabilísticas

La biblioteca de distribuciones de probabilidad de ProModel permite simular la variabilidad de los procesos. En la tabla 6.1 se numeran las distribuciones de probabilidad incluidas en ProModel y la forma de programar cada una de ellas.

Tabla 6.1 Funciones de probabilidad disponibles en ProModel.

Distribución de probabilidad	Codificación en ProModel	Parámetros
Beta	$B(1, \alpha_1, \alpha_2, a, b)$	α_1 : forma α_2 : forma a: mínimo b: máximo
Binomial	$BI(N, p)$	N: intentos p: probabilidad
k-Erlang	$ER(\mu, k)$	k: forma μ : media
Exponencial	$E(\mu)$	μ : media
Gamma	$G(\alpha, \beta)$	α : forma β : escala
Gaussiana inversa	$IG(\alpha, \beta)$	α : forma β : escala
Geométrica	$GEO(p)$	p: probabilidad
Lognormal	$L(\mu, \sigma)$	μ : media σ : desviación
Normal	$N(\mu, \sigma)$	μ : media σ : desviación
De Poisson	$P(\mu)$	μ : media
De Pearson V	$P5(\alpha, \beta)$	α : forma β : escala
De Pearson VI	$P6(\alpha_1, \alpha_2, \beta)$	α_1 : forma α_2 : forma β : escala
Triangular	$T(a, c, b)$	a: mínimo c: moda b: máximo
Uniforme	$U(\mu, h)$	μ : media h: medio rango
Uniforme discreta	$U(a, b)$	a: mínimo b: máximo
De Weibull	$\gamma + W(\alpha, \beta)$	α : forma β : escala γ : localización

Ejemplo 6.1

A un proceso llegan tres diferentes tipos de pieza. El proceso consta de dos operaciones en serie: lavado e inspección. Antes de cada operación las piezas pasan por almacenes de producto en proceso, con capacidad prácticamente infinita. Se dispone de una lavadora y de dos inspectores en paralelo. Los datos de tiempo entre llegadas y tiempos de proceso para cada tipo de pieza son los siguientes:

Pieza	Tiempo entre llegadas (min/pieza)	Tiempo de proceso (min/pieza)	
		Lavado	Inspección
A	Exponencial (6)	Uniforme (3 ± 2)	Normal (8, 2)
B	Exponencial (9)	3-Erlang (4)	Triangular (3, 5, 7)
C	Exponencial (8)	De Weibull (0, 2, 3)	De Weibull (9, 1, 4)

Para empezar, definiremos las 4 localizaciones, como se muestra en la figura 6.1 (recuerde que esto se hace mediante el comando **Build / Locations**).

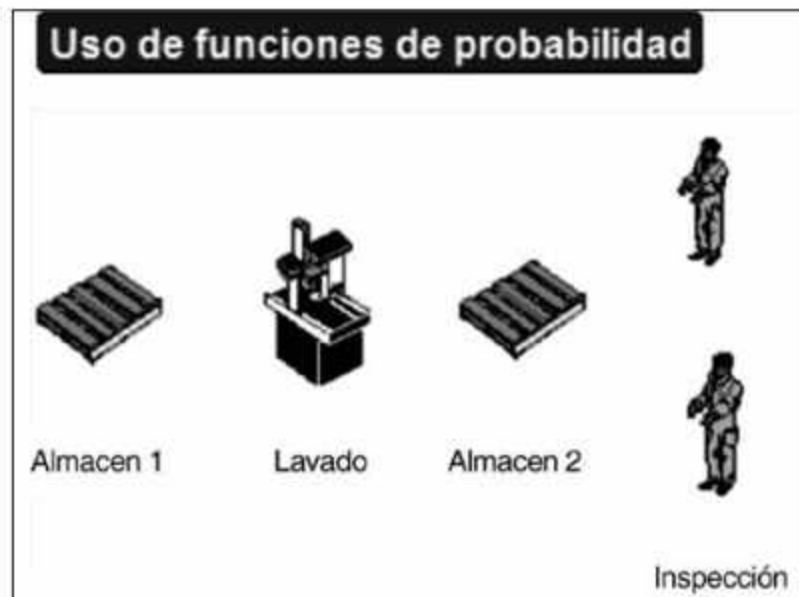


Figura 6.1 Layout del ejemplo 6.1.

Ahora determinaremos las entidades correspondientes a los tres tipos de piezas (**Build / Entities**), y el tiempo exponencial entre llegadas, para ello utilice la columna **Frequency** de la ventana **Arrivals (Build / Arrivals)**, como se muestra en la figura 6.2.

Arrivals					
Entity...	Location...	Qty Each...	First Time...	Occurrences	Frequency
Pieza_A	Almacen_1	1	0	INF	E(6)
Pieza_B	Almacen_1	1	0	INF	E(9)
Pieza_C	Almacen_1	1	0	INF	E(8)

Figura 6.2 Programación de la distribución de probabilidad del tiempo entre llegadas.

Enseguida programaremos las tres diferentes rutas y cada uno de los tiempos de proceso, usaremos la columna **Operation** de la ventana **Processing (Build / Processing)**. La figura 6.3 muestra parte del proceso y la forma de programar las distribuciones de probabilidad.

Entity...	Location...	Operation...
Pieza_A	Almacen_1	
Pieza_A	Lavadora	WAIT U(3,2)
Pieza_A	Almacen_2	
Pieza_A	Inspector	WAIT N(8,2)
Pieza_B	Almacen_1	
Pieza_B	Lavadora	WAIT ER(4,3)
Pieza_B	Almacen_2	
Pieza_B	Inspector	WAIT T(3,5,7)
Pieza_C	Almacen_1	
Pieza_C	Lavadora	WAIT W(2,3)
Pieza_C	Almacen_2	
Pieza_C	Inspector	WAIT 9+W(1,4)

Figura 6.3
Programación de las funciones de probabilidad del tiempo de proceso.

En caso de que no recuerde la forma de programar alguna de las funciones, haga clic en el botón **Operation** de la ventana **Processing** para usar el constructor de lógica, que permite acceder a la ventana de diálogo **Logic Builder** con sólo hacer clic en el icono del martillo. Luego haga clic en la opción **Distribution Functions** de la sección **Logic Elements** para desplegar las diferentes funciones de probabilidad (vea la figura 6.4).

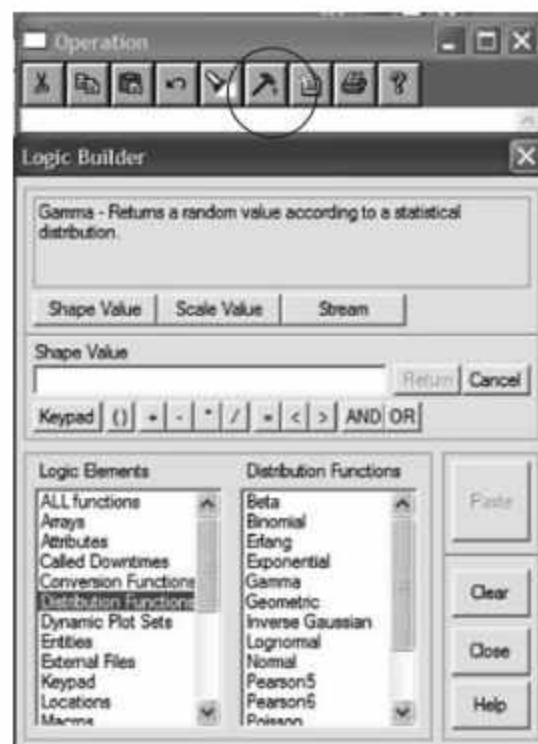


Figura 6.4
Uso del constructor de lógica para programar funciones de probabilidad.

Para consultar el texto completo del programa que hemos creado en ProModel para el ejemplo 6.1, abra el menú **File** y elija **View Text**. El resultado deberá ser similar al que se ilustra en la figura 6.5.

```

*****
*                               Formatted Listing of Model:
*                               C:\Archivos de programa\ProModel\Models\EJEMPLO4.1.MOD
*                               *****
Time Units:                      Minutes
Distance Units:                  Feet
*****

*                               Locations
*                               *****
Name          Cap Units  Stats          Rules          Cost
-----
Almacen_1     INF 1      Time Series   Oldest, ,
Lavadora      1 1          Time Series   Oldest, ,
Almacen_2     inf 1      Time Series   Oldest, ,
Inspector     1 2          Time Series   Oldest, , First
Inspector.1   1 1          Time Series   Oldest, ,
Inspector.2   1 1          Time Series   Oldest, ,
*****

*                               Entities
*                               *****
Name          Speed (fpm)  Stats          Cost
-----
Pieza_A       150           Time Series
Pieza_B       150           Time Series
Pieza_C       150           Time Series
*****

*                               Processing
*                               *****
Process                               Routing
Entity  Location  Operation          Blk  Output  Destination  Rule  Move Logic
-----
Pieza_A Almacen_1
Pieza_A Lavadora  WAIT U(3,2)        1    Pieza_A  Lavadora     FIRST 1
Pieza_A Almacen_2
Pieza_A Inspector WAIT N(8,2)        1    Pieza_A  Inspector   FIRST 1
Pieza_A Inspector WAIT N(8,2)        1    Pieza_A  EXIT        FIRST 1
Pieza_B Almacen_1
Pieza_B Lavadora  WAIT ER(4,3)       1    Pieza_B  Lavadora     FIRST 1
Pieza_B Almacen_2
Pieza_B Inspector WAIT T(3,5,7)      1    Pieza_B  Inspector   FIRST 1
Pieza_B Inspector WAIT T(3,5,7)      1    Pieza_B  EXIT        FIRST 1
Pieza_C Almacen_1
Pieza_C Lavadora  WAIT W(2,3)        1    Pieza_C  Lavadora     FIRST 1
Pieza_C Almacen_2
Pieza_C Inspector WAIT 9+W(1,4)     1    Pieza_C  Inspector   FIRST 1
Pieza_C Inspector WAIT 9+W(1,4)     1    Pieza_C  EXIT        FIRST 1
*****

*                               Arrivals
*                               *****
Entity  Location  Qty Each  First Time Occurrences  Frequency  Logic
-----
Pieza_A Almacen_1 1          0          INF          E(6)
Pieza_B Almacen_1 1          0          INF          E(9)
Pieza_C Almacen_1 1          0          INF          E(8)

```

Figura 6.5 Sintaxis del programa para el ejemplo 6.1.

En ocasiones no es posible tener la distribución o función de probabilidad teórica por insuficiencia de datos o bien porque la información no se ajustó a alguna de las distribuciones teóricas existentes. Por ejemplo, si en el proceso antes descrito la información del tiempo de lavado de la pieza A en lugar de ser una función Uniforme(3 ± 2) minutos fuera la siguiente: 10% de las piezas se lavan en 1 minuto, el 40% se lavan en 2 minutos, el 5%, en 3 minutos y el 45% restante, en 32 minutos, tendríamos que crear funciones personalizadas similares a las que se muestran en la figura 6.6 o en la figura 6.7. Estas funciones se crean a través de la opción **Build/ More Elements /User Distributions**.

The screenshot shows the 'User Distributions' dialog box with the following configuration:

ID	Type	Cumulative	Table...
tiempo_lavado	Discrete	No	Defined

Below the dialog box is a 'Table for tiempo_lavado' with the following data:

Percentage	Value
10	1
40	2
5	3
45	32

Figura 6.6 Función de usuario mediante el formato de función no acumulada.

The screenshot shows the 'User Distributions' dialog box with the following configuration:

ID	Type	Cumulative	Table...
tiempo_lavado	Discrete	Yes	Defined

Below the dialog box is a 'Table for tiempo_lavado' with the following data:

Percentage	Value
10	1
50	2
55	3
100	32

Figura 6.7 Función de usuario mediante el formato de función acumulada.

Para la creación de estas funciones se requiere de la siguiente información:

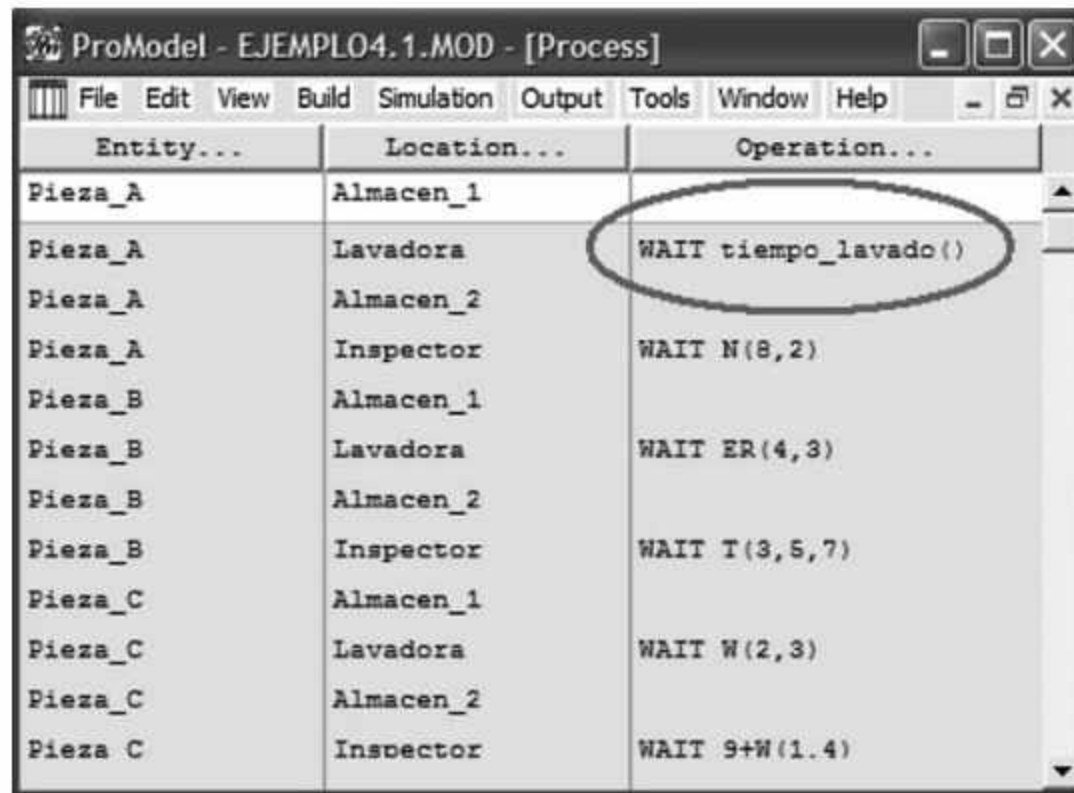
ID: Nombre de la función a crear

Type: Puede ser discreta o continua y depende de las características de la variable dependiente.

Cumulative: Se activa la opción *Yes* o *No* para indicar si los porcentajes de la función serán alimentados en forma acumulativa o no.

Table: Permite asociar cada porcentaje con el valor de la variable dependiente que le corresponde.

Una vez que la función de usuario es dada de alta, podemos hacer uso de ella durante la creación de la ruta de proceso. Como en este caso la función se utiliza para simular el tiempo de lavado de la pieza A, al hacer uso de ella en el **Process** lo haríamos como **WAIT tiempo_lavado()** tal como se muestra en la figura 6.8. Este tipo de funciones pueden ser utilizadas en cualquier campo de lógica del ProModel.



Entity...	Location...	Operation...
Pieza_A	Almacen_1	
Pieza_A	Lavadora	WAIT tiempo_lavado()
Pieza_A	Almacen_2	
Pieza_A	Inspector	WAIT N(8,2)
Pieza_B	Almacen_1	
Pieza_B	Lavadora	WAIT ER(4,3)
Pieza_B	Almacen_2	
Pieza_B	Inspector	WAIT T(3,5,7)
Pieza_C	Almacen_1	
Pieza_C	Lavadora	WAIT W(2,3)
Pieza_C	Almacen_2	
Pieza_C	Inspector	WAIT 9+W(1.4)

Figura 6.8
Utilización de la función de usuario en el ejemplo 6.1.

6.2 Recursos

Los recursos son mecanismos que requieren las entidades para completar una operación, y se caracterizan en primer lugar por tener una disponibilidad limitada. En ProModel encontramos dos tipos de recursos:

Recursos estáticos. Son aquellos sin una ruta de movimiento y que, por lo tanto, permanecen inmóviles. Se utilizan principalmente para modelar recursos necesarios para llevar a cabo una tarea dentro de una localización (por ejemplo, el operador de una máquina). También pueden emplearse en más de una localización, o bien para mover entidades de una localización a otra, siempre y cuando la ausencia de movimiento no sea un factor relevante en el modelo.

Recursos dinámicos. Son aquellos que se mueven a través de una red de rutas (**Path Networks**). Estos recursos permiten transportar entidades entre localizaciones, para modelar, por ejemplo, un montacargas que mueve contenedores de una máquina a un almacén, o un operario que tiene que operar dos o más máquinas; en estos casos, el tiempo de traslado entre las máquinas impacta los resultados del modelo.

Las instrucciones más comunes relacionadas con el uso de un recurso son GET, FREE, USE y MOVE WITH. Analicemos a continuación dos de las más utilizadas.

Instrucción GET

La sintaxis general de esta instrucción es:

GET {<Cantidad>} <recurso> {,<prioridad 1>,<prioridad 2>}} {AND o OR {<Cantidad>} <recurso> {,<prioridad 1> {,<prioridad2>}}}

La instrucción **GET** captura un recurso o combinación de recursos, de acuerdo con cierta prioridad especificada. Si ya se tiene un recurso capturado, la entidad tratará de capturar uno adicional.

Ejemplos:

GET *Herramienta* 1
 GET 1 *Operario*, 2, 25
 GET *Grua* 1, 20 AND (*Grua* 2, 10 OR *Polipasto*, 20, 70)
 GET 2 *Cajas*, AND (*Pegamento* OR *Cinta*)

Instrucción FREE

La sintaxis general es:

FREE <{cantidad} recurso 1>,...

FREE libera recursos previamente capturados con las instrucciones GET o JOINTLY GET.

Ejemplos:

FREE *pallet*
 FREE *Juan, Paco, Luisa*
 FREE 4 *Tornillos*, 3 *Tuercas*
 FREE *all*

Ejemplo 6.2

Considere el sistema de manufactura que se ilustra en la figura 6.9, el cual consta de dos procesos en serie: torneado y fresado de barras en las máquinas llamadas Lathe y Mill respectivamente. El tiempo de torneado es de 3 min/pieza, y el de fresado es de 2.7 min/pieza. Para operar ambas máquinas se ha contratado a un solo operario llamado Machinist. Las barras esperan antes de cada proceso en almacenes denominados Pallet1 y Pallet2. La tasa de entrada es de 10 piezas/h. Simule el sistema por 24 horas para determinar la utilización del equipo y del personal.

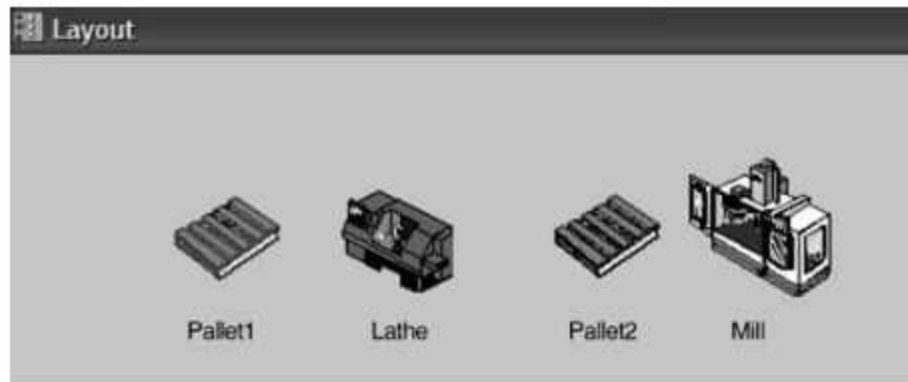


Figura 6.9
Layout del sistema de manufactura (ejemplo 6.2).

Iniciaremos la construcción del modelo definiendo los siguientes elementos:

- En la ventana **Locations (Build / Locations)** active la ventana de edición que se ilustra en la figura 6.10, y defina las localizaciones **Lathe**, **Mill**, **Pallet1** y **Pallet2**.





Locations				
Icon	Name	Cap.	Units	DTs...
	Lathe	1	1	None
	Mill	1	1	None
	Pallet1	inf	1	None
	Pallet2	inf	1	None

Figura 6.10
Definición de las localizaciones y su capacidad.

- En la ventana **Entities (Build / Entities)** defina la entidad **Barra** (vea la figura 6.11).


Entities		
Icon	Name	Speed (fpm)
	Barra	150

Figura 6.11
Ventana de edición para la definición de la entidad *Barra*.

- En la ventana **Arrivals (Build / Arrivals)** especifique un número infinito de ocurrencias para la entidad **Barra**, con un tiempo entre llegadas de 6 minutos al **Pallet1** (vea la figura 6.12). (Al definir un número de ocurrencias infinito, es necesario que el modelo de simulación se detenga mediante la determinación de un tiempo de corrida.)

Arrivals					
Entity...	Location...	Qty each...	First Time	Occurrences	Frequency
Barra	Pallet1	1	0	INF	6

Figura 6.12
Modelado de las llegadas de material.

- En la ventana **Resources (Build / Resources)** defina el recurso **Machinist** y la cantidad del recurso que se tiene (vea la figura 6.13).

Icon	Name	Units	DTs...	Stats...	Specs...
	Machinist	1	None	By Unit	No Network

Figura 6.13 Ventana de edición para la definición de recursos.

- En la ventana **Processing (Build / Processing)**, cree la ruta de producción de las barras a través del torno y la fresadora, incluya en dichas localizaciones la captura del recurso con la instrucción **GET Machinist**, y su liberación con la instrucción **FREE Machinist** después del tiempo de proceso. La figura 6.14 muestra una parte de la programación de la ruta.

Entity...	Location...	Operation...
Barra	Pallet1	
Barra	Lathe	300FREE Machinist
Barra	Pallet2	
Barra	Mill	GET Machinist

Id	Output...	Destination...	Rule...
1	Barra	Pallet2	FIRST 1

Operation window content:

```

GET Machinist
WAIT 3
FREE Machinist
    
```

Figura 6.14 Ruta de producción de las barras.

- Por último, abra el menú **Simulation** y haga clic en el comando **Options** para desplegar el cuadro de diálogo que se muestra en la figura 6.15. En el cuadro de texto de este cuadro establezca la longitud de la corrida de simulación en 24 horas.

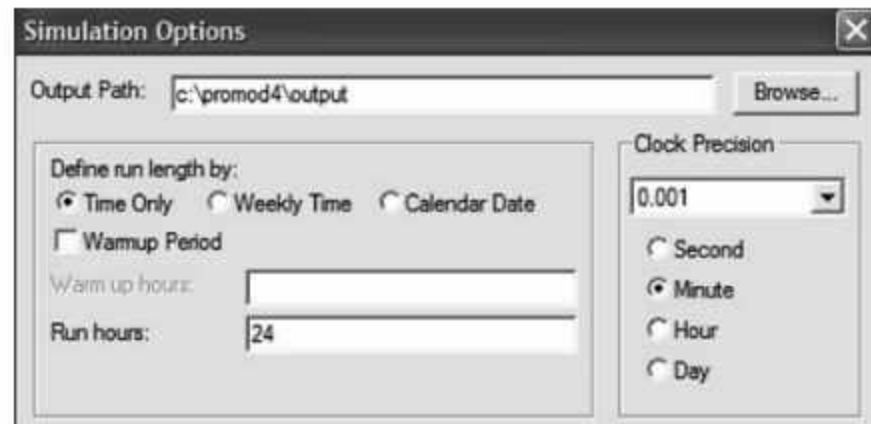


Figura 6.15
Definición de la duración de la corrida de simulación.

Para consultar el resultado de los pasos anteriores, abra el menú **File** y haga clic en el comando **View Text**. En la figura 6.16 hemos encerrado en un círculo el uso de las instrucciones GET y FREE.

Locations							
Name	Cap	Units	Stats	Rules	Cost		
Lathe	1	1	Time Series	Oldest,			
Mill	1	1	Time Series	Oldest,			
Pallet1	inf	1	Time Series	Oldest,			
Pallet2	inf	1	Time Series	Oldest,			
Entities							
Name	Speed (fpm)	Stats	Cost				
Barra	150	Time Series					
Resources							
Name	Units	Stats	Res Search	Ent Search	Path	Motion	Cost
Machinist 1		By Unit	None	Oldest		Empty: 150 fpm Full: 150 fpm	
Processing							
Process				Routing			
Entity	Location	Operation	Blk	Output	Destination	Rule	Move Logic
Barra	Pallet1		1	Barra	Lathe	FIRST	1
Barra	Lathe	GET Machinist WAIT 3 FREE Machinist	1	Barra	Pallet2	FIRST	1
Barra	Pallet2		1	Barra	Mill	FIRST	1
Barra	Mill	GET Machinist WAIT 2.7 FREE Machinist	1	Barra	EXIT	FIRST	1
Arrivals							
Entity	Location	Qty each	First Time	Occurrences	Frequency	Logic	
Barra	Pallet1	1	0	INF	6		

Figura 6.16 La programación completa de la simulación.

Ejecute el modelo. Observe que no se presenta ningún tipo de movimiento del recurso, ya que no se trata de un recurso dinámico. Esto se debe a que, al definir el recurso *Machinist*, el parámetro de la columna **Specs** de la ventana **Resources** se conservó como **No Network** (vea la figura 6.13). Más adelante emplearemos recursos en combinación con Path Networks para permitir la visualización de recursos en movimiento.

Finalmente, revise el reporte de resultados de la simulación. Para ello:

Abra el **Output** y haga clic en el comando **View Statistics**. Seleccione en la pestaña **Chart** la opción **Tables/Location Summary** y **Tables/Resources Summary**. Las ventanas que se desplegarán serán similares a las que se muestran en la figura 6.17. Como puede observar, el recurso ha sido utilizado un total de 95% del tiempo, 50% del cual corresponde al torno, y 45% a la fresadora.

Location Summary									
Scenario	Name	Scheduled Time (Hr)	Capacity	Total Entries	Average Time Per Entry (Min)	Average Contents	Maximum Contents	Current Contents	% Utilization
Ejemplo	Lathe	24.00	1.00	241.00	2.99	0.50	1.00	1.00	50.00
Ejemplo	Mill	24.00	1.00	240.00	2.70	0.45	1.00	0.00	45.00
Ejemplo	Pallet1	24.00	999,999.00	241.00	0.00	0.00	1.00	0.00	0.00
Ejemplo	Pallet2	24.00	999,999.00	240.00	0.00	0.00	1.00	0.00	0.00

Resource Summary							
Scenario	Name	Units	Scheduled Time (Hr)	Work Time (Min)	Number Times Used	Average Time Per Usage (Min)	% Utilization
Ejemplo	Machinist	1.00	24.00	1,366.00	461.00	2.84	95.00

Figura 6.17 Reporte de resultados del uso de localizaciones y recursos con formato Output Viewer 4.0.

6.3 Paros en los equipos

Un paro provoca que un recurso o localización quede inhabilitada para operar, o fuera de servicio. Desde el punto de vista de la simulación, un paro puede representar fallas, descansos, mantenimientos preventivos (en cuyo caso se necesitarán uno o más recursos), interrupciones programadas o cambios de turno.

Una localización puede quedar fuera de servicio en función del tiempo de simulación (Clock), por tiempo de uso (Usage), por número de entidades procesadas (Entity), o por un cambio en el tipo de entidad a procesar (Setup). Los paros se procesan de manera independiente, por lo que diferentes paros pueden ocurrir de manera simultánea en una misma localización (excepto aquellos que se deban a cambios de tipo de entidad).

Por su parte, un recurso puede quedar fuera de servicio solamente en función del tiempo de simulación (Clock), o por tiempo de uso (Usage).

Otro método para definir paros por descansos o turnos consiste en utilizar el editor de turnos (Shift Editor). Este procedimiento tiene la ventaja de permitir paros en todo un grupo de localizaciones.

Para definir un paro en una localización o en un recurso:

- Abra el menú **Build** y haga clic en el comando **Locations** (o **Resources**, según el caso) para desplegar la ventana de edición que se muestra en la figura 6.18 (por supuesto, si está definiendo un paro en un recurso, la ventana mostrará el nombre **Resources** en la barra de título).

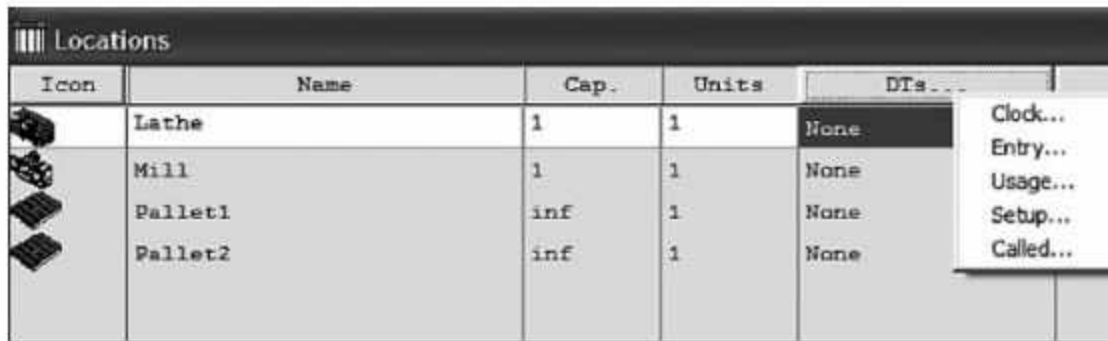


Figura 6.18 Programación de diferentes tipos de paro en las localizaciones.

- Haga clic en el botón leyenda **DTs** y elija, en el menú contextual que aparece, el tipo de paro que desea programar. Enseguida se abrirá una ventana de edición similar a la que se ilustra en la figura 6.19.

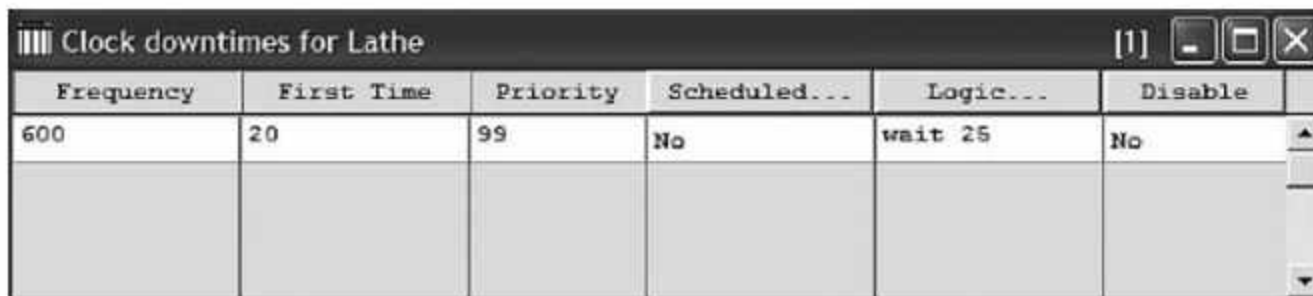


Figura 6.19 Determinación de parámetros de un paro en una localización.

Las ventanas de edición de paros contienen un conjunto de columnas donde podemos especificar los valores de los siguientes parámetros:

Frequency. Representa el tiempo que transcurre entre paros consecutivos.

First time. Representa la hora en que ocurre el primer paro; si se deja en blanco, el parámetro tomará el valor asignado en el campo **Frequency**.

Priority. Representa la prioridad relativa que tendrá el paro. De manera predeterminada, la prioridad es 99. Si los paros ocurren hasta que la entidad que está en posesión de la localización sale de ella, el rango de prioridades es de 0 a 99. Si un paro ocurre mientras la localización está ocupada, interrumpiendo el proceso de la entidad, el rango de prioridades es de 100 a 999.

Scheduled. Permite indicar si los paros son programados o no. Los paros programados se deducen del tiempo total de simulación para el cálculo estadístico de los resultados.

List. En caso de tener varias unidades de un mismo recurso, este parámetro permite seleccionar el número de la unidad al que se aplica el paro.

Node. En caso de recursos dinámicos, representa el nodo, dentro de su ruta, donde debe esperar el recurso mientras el paro esté activo.

Logic. Permite programar los eventos y actividades que ocurren durante el paro. En este campo es posible utilizar las instrucciones GET, FREE, WAIT, USE, GRAPHIC y JOINTLY GET.

Disable. Permite deshabilitar temporalmente el paro, sin tener que eliminarlo.

Ejemplo 6.3

Considere un sistema de manufactura similar al del ejemplo 6.2, con dos máquinas en serie para los procesos de torneado y fresado. El tiempo de torneado es de 3 min/pieza, y el de fresado es de 2.7 min/pieza. Para operar ambas máquinas se ha contratado a un solo operario. La tasa de entrada es 10 piezas/h.

El torno tiene una frecuencia de fallas exponencial de 400 minutos, y para su reparación se necesita un mecánico. El tiempo de reparación es de 10 ± 3 minutos, con distribución uniforme. El operario de producción descansa 5 minutos con distribución exponencial cada 120 minutos de trabajo. Simule el sistema por 24 horas para determinar el impacto de las fallas y los descansos en la utilización del equipo y del personal.

Partiendo de los elementos definidos en el modelo del ejemplo anterior:

- Abra el menú **Build** y elija **Resources** para acceder a la ventana de edición correspondiente (vea la figura 6.20).
- Defina como recurso al mecánico que hará las reparaciones del torno. En cuanto al recurso *Machinist*, haga clic en el botón **DTs** para editar sus descansos.

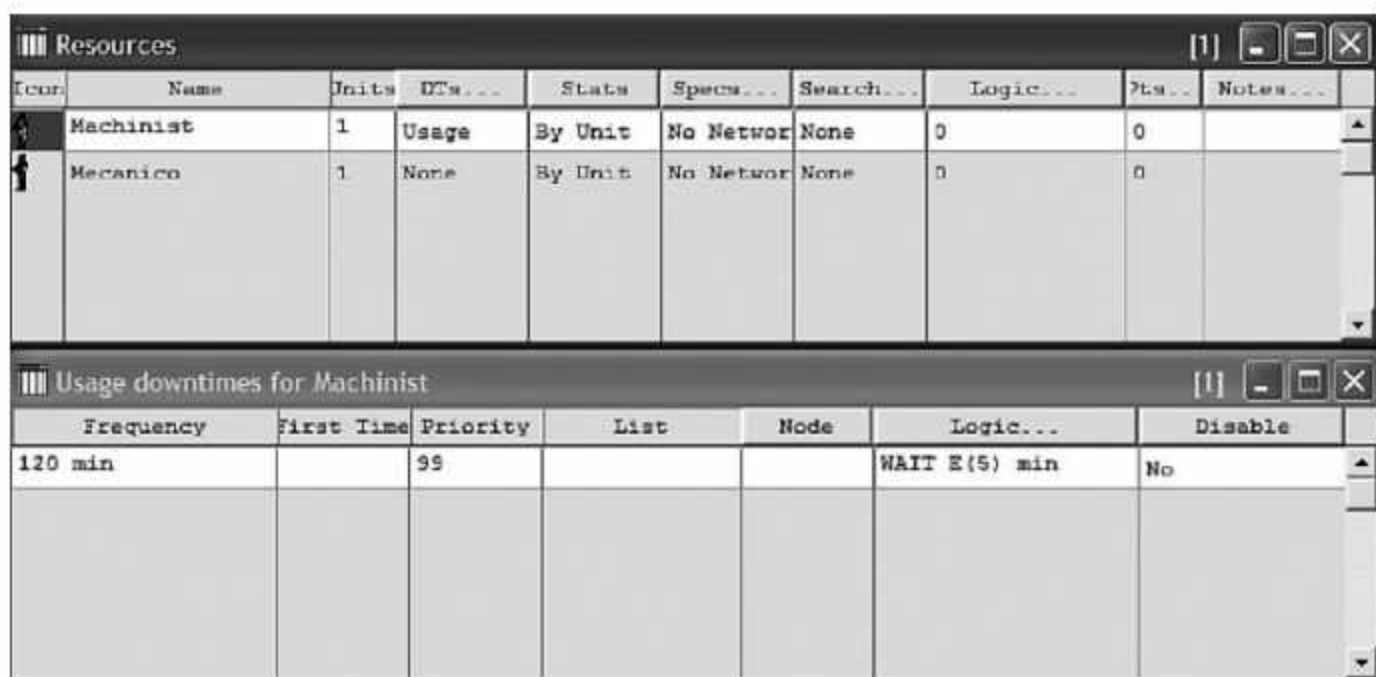


Figura 6.20 Edición de los descansos de Machinist y adición del recurso Mecanico.

- Abra la ventana **Locations (Build / Locations)**. Seleccione la localización *Lathe* (vea la figura 6.21), para editar las fallas del equipo haga clic en el botón **DTs** y luego en **Clock** (del menú contextual). En este caso la prioridad de 999 permite que la máquina falle aún teniendo una entidad en proceso.

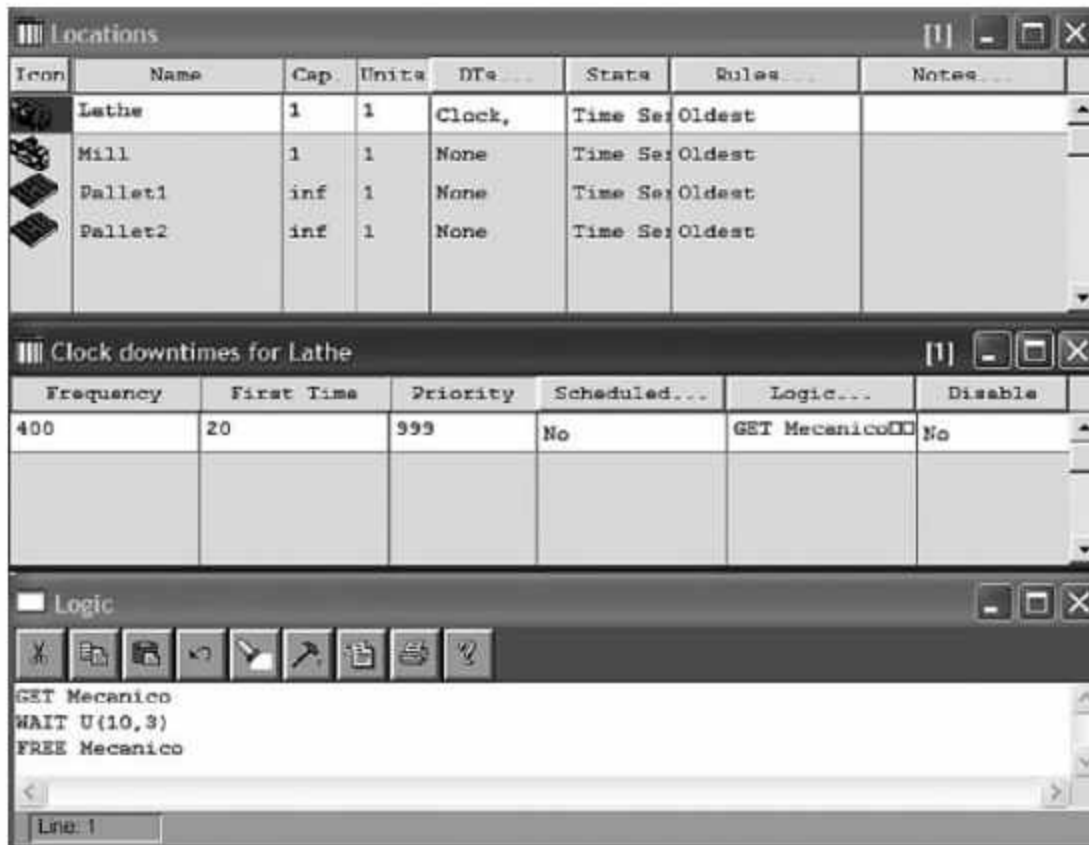


Figura 6.21 Edición de los descansos de *Machinist* y uso del recurso *Mecanico* para hacer la reparación.

- Abra el reporte de resultados (**Output / View Statistics**) para verificar las estadísticas del nuevo recurso *Mecanico*, y el impacto de los paros en las estadísticas de *Machinist*, *Lathe* y *Mill* (vea la figura 6.22).

Location Summary									
Scenario	Name	Scheduled Time (Hr)	Capacity	Total Entries	Average Time Per Entry (Min)	Average Contents	Maximum Contents	Current Contents	% Utilization
Ejemplo	Lathe	24.00	1.00	240.00	5.54	0.92	1.00	1.00	92.34
Ejemplo	Mill	24.00	1.00	239.00	2.84	0.47	1.00	1.00	47.16
Ejemplo	Pallet1	24.00	999,999.00	241.00	3.83	0.64	2.00	1.00	0.00
Ejemplo	Pallet2	24.00	999,999.00	239.00	0.00	0.00	1.00	0.00	0.00

Resource Summary							
Scenario	Name	Units	Scheduled Time (Hr)	Work Time (Min)	Number Times Used	Average Time Per Usage (Min)	% Utilization
Ejemplo	Machinist	1.00	24.00	1,361.63	481.00	2.83	94.56
Ejemplo	Mecanico	1.00	24.00	39.99	4.00	10.00	2.78

Figura 6.22 Impacto de los paros en la utilización de los recursos y las localizaciones. (Formato Output Viewer 4.0).

Utilización de turnos

ProModel ofrece al usuario una buena interfase para incluir paros programados en los equipos mediante la programación de turnos de trabajo y descansos de los elementos operativos de un sistema (localizaciones o recursos). Algunos ejemplos de estos paros podrían ser horarios de comida, descansos, cambios de turno, descansos dominicales etcétera. Los turnos de trabajo de cada elemento pueden definirse en forma separada e incluir ciclos de trabajo que se repitan semanalmente. Los turnos de trabajo se definen y se asignan en el módulo **Build/Shifts**.

Ejemplo 6.4

Los clientes compran computadoras en cierta tienda. Hay tres tipos de clientes: el 40% compra computadoras económicas, el 50% computadoras estándar y el 10%, computadoras de última generación. El tiempo entre arribos de los clientes está exponencialmente distribuido con media de 16 minutos. El tiempo de compra (en minutos) depende del tipo de computadora a comprar: para una computadora económica es de 8 a 12 minutos, para una estándar de 10 a 16 minutos y para una de última generación es de 14 a 22 minutos, en los tres casos con una función de densidad uniforme. La tienda sólo tiene un dependiente. El horario de atención de lunes a viernes es de 9:00 a 13:00 h y de 15:00 a 20:00 h, el sábado de 9:00 a 14:00 h y los domingos permanece cerrada. Corra el modelo de simulación durante un mes para encontrar el tiempo promedio de espera antes de ser atendido.

Definamos tres localizaciones (Entrada, Espera y Venta) y cuatro entidades (Clientes, Pobres, Ricos y Magnates). Los arribos y la lógica de proceso se definen de acuerdo a la lógica mostrada en las figuras 6.23 y 6.24. Una imagen del modelo de simulación se muestra en la figura 6.25.

Processing							
Process			Routing				
Entity	Location	Operation	Blk	Output	Destination	Rule	Move Logic
Clientes	Puerta		1	Pobres	Espera	0.400000	1
				Ricos	Espera	0.500000	
				Magnates	Espera	0.100000	
Pobres	Espera		1	Pobres	Venta	FIRST	1
Pobres	Venta	WAIT U(10,2) min	1	Pobres	EXIT	FIRST	1
Ricos	Espera		1	Ricos	Venta	FIRST	1
Ricos	Venta	WAIT U(13,3) min	1	Ricos	EXIT	FIRST	1
Magnates	Espera		1	Magnates	Venta	FIRST	1
Magnates	Venta	WAIT U(18,4) min	1	Magnates	EXIT	FIRST	1

Figura 6.23 Lógica de la ruta de los clientes.

Arrivals					
Entity...	Location...	Qty Each...	First Time...	Occurrences	Frequency
Clientes	Puerta	1	0	INF	E(16) min

Figura 6.24 Arribos a la tienda.

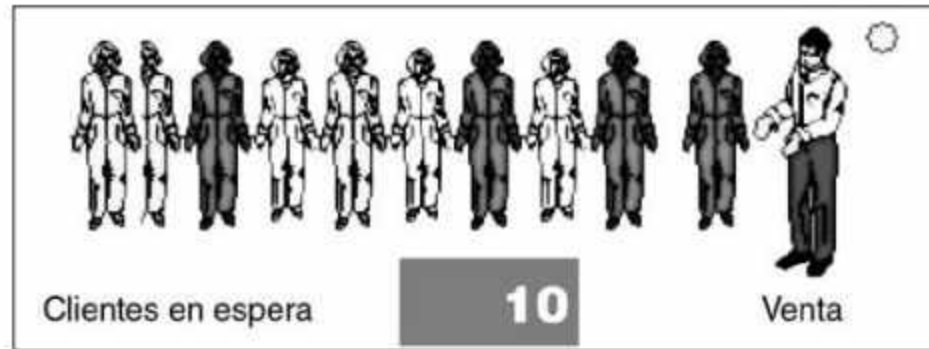


Figura 6.25
Esquematación
del ejemplo 6.4.

Para definir el horario de trabajo:

- Active el editor de turnos **Shifts Editor (Build / Shifts / Define)** y se abrirá un calendario semanal similar al de la figura 6.26

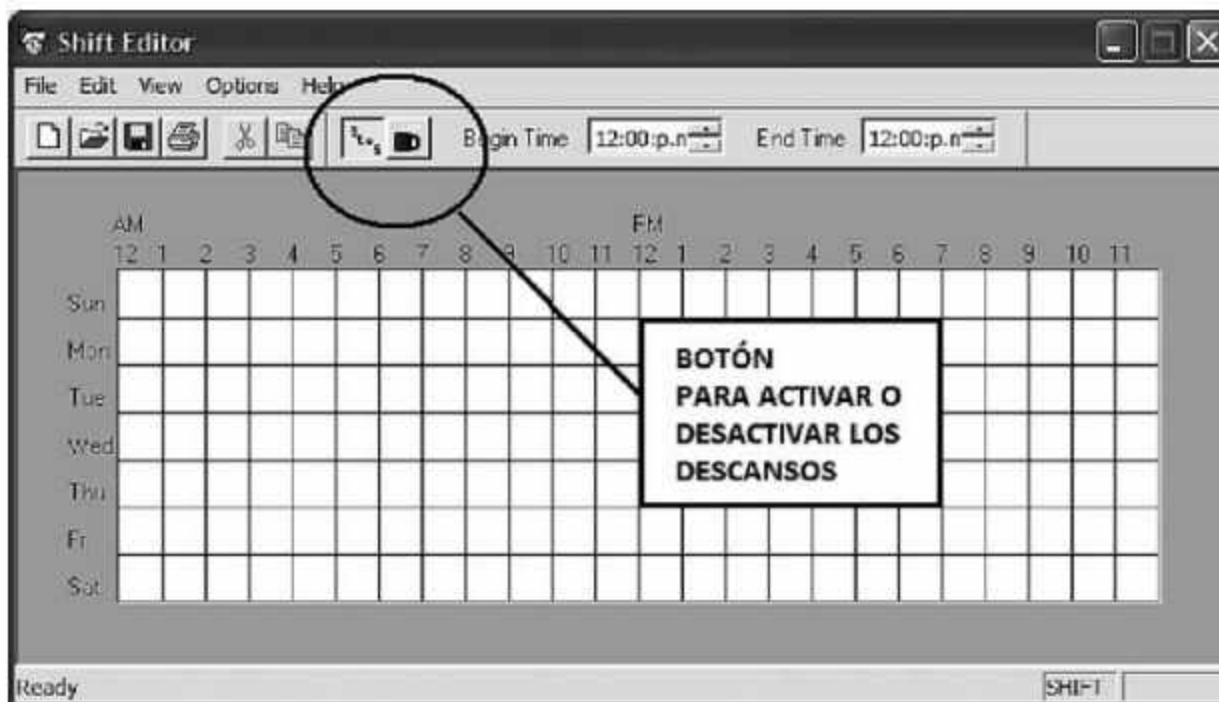


Figura 6.26 Editor de turnos.

- Verifique que el icono con la taza de café permanezca resaltado (desactivado). Seleccione cualquier día de la semana y deslice de izquierda a derecha el ratón manteniendo oprimido el botón izquierdo, notará que el área de ese día se cubre de color azul.
- Oprima el icono con la taza de café para activar el proceso de descansos. Deslice otra vez de izquierda a derecha el ratón con el botón izquierdo oprimido cubriendo el horario donde no se trabajará, al finalizar observará superpuesta un área de color rojo. En este proceso siempre quedará el área roja arriba del área azul, y en caso de no ser así el editor marcará error. Al terminar cualquier día obtendremos una imagen similar a la figura 6.27.

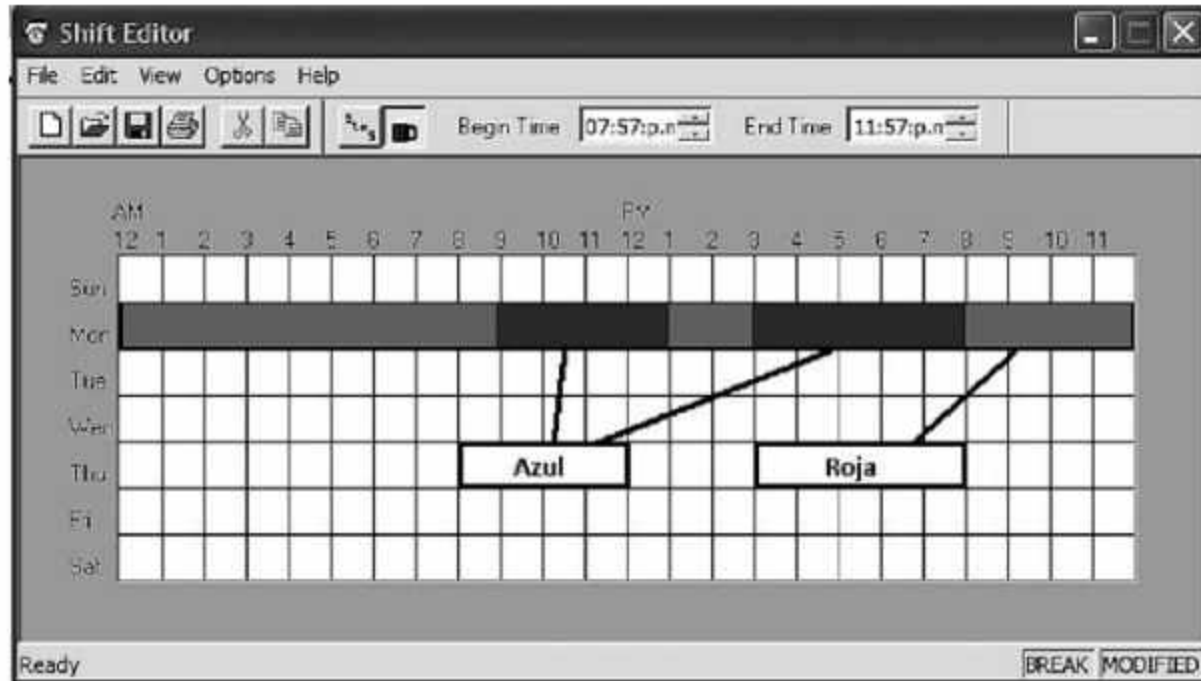


Figura 6.27 Horario de descansos del lunes.

- Repita el proceso para todos los días, activando y desactivando la taza de café, note que para desactivar la taza de café debe oprimir el botón que está a la izquierda de la taza. Si la estructura de uno de los días se repite puede usar la opción de duplicar ese día en el menú **Edit** tantas veces como se desee. Al finalizar el proceso (vea la figura 6.28) debemos guardar (**File / Save As**) nuestro archivo de turnos con extensión **.SFT**.

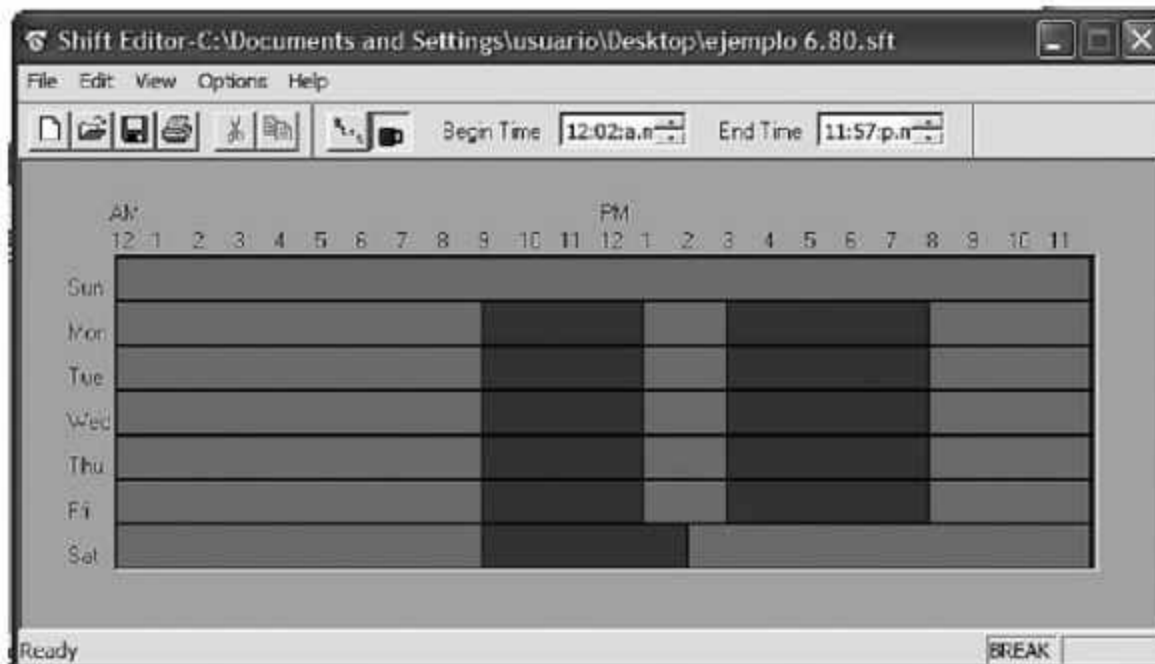


Figura 6.28 Programación completa de los turnos.

- Ahora debemos definir a cuáles localizaciones o recursos afecta el programa de turnos, por lo que activamos la opción **Shifts Assignments (Build / Shifts / Assign)**. En el campo **Locations** seleccionamos las localizaciones *Entrada y Venta*

y en el campo **Shift Files** buscamos la ruta donde guardamos el archivo SFT. En la figura 6.29 se visualiza este proceso.

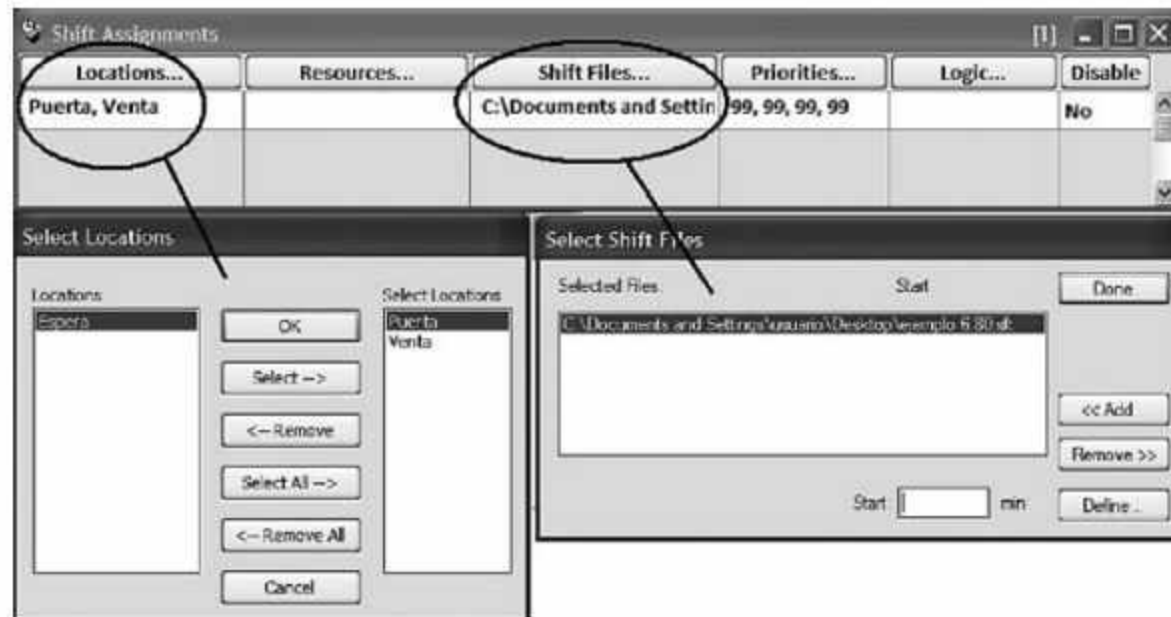


Figura 6.29 Asignación de los turnos.

- Active la ventana de **Simulation Options (Simulation / Options)** y en la sección **Run Length** seleccione la opción **Calendar Date**. En esta misma sección con los botones **Sim. Begin** y **Sim. End**, defina la fecha de inicio de simulación y la fecha de fin de simulación. Ejecute el modelo y observe que el reloj de simulación corre en un esquema semanal. Al finalizar la simulación aparecerá un aviso que indica que ocurrieron **Failed Arrivals**, lo cual es normal ya que mientras la tienda permanecía cerrada los clientes siguieron llegando y fueron rechazados.

En la figura 6.30 se despliegan los resultados finales del tiempo promedio de espera de los clientes desglosados por tipo de cliente.

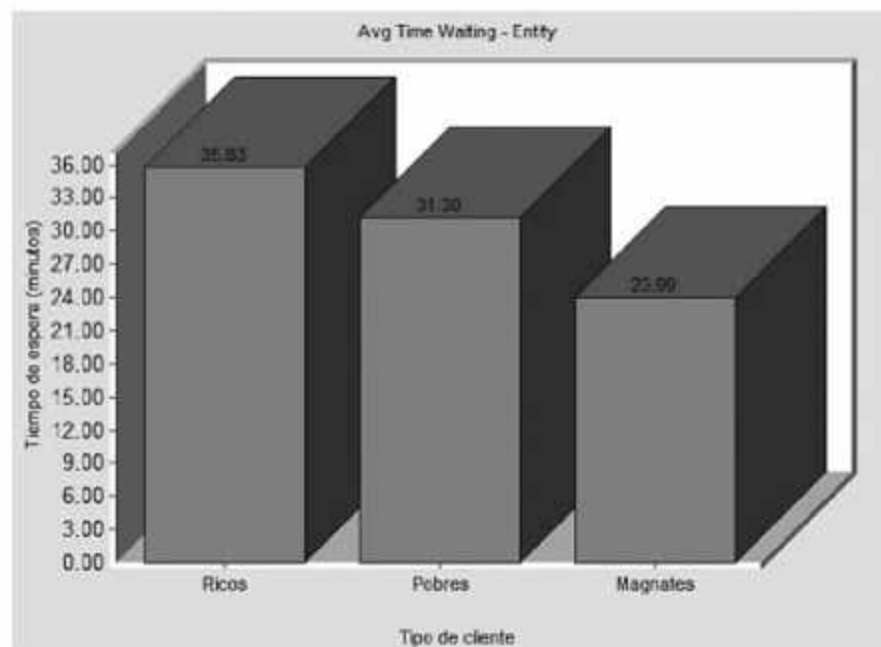


Figura 6.30 Resultados de la ejecución del modelo por tipo de cliente.

6.4 Reglas de ruteo

La ventana de edición que se ilustra en la figura 6.31 muestra una de las herramientas disponibles en ProModel. **Rule** permite gran versatilidad en la creación del proceso al construir las rutas de las entidades. Para desplegar la ventana correspondiente, abra el menú **Build** y haga clic en el comando **Processing**.

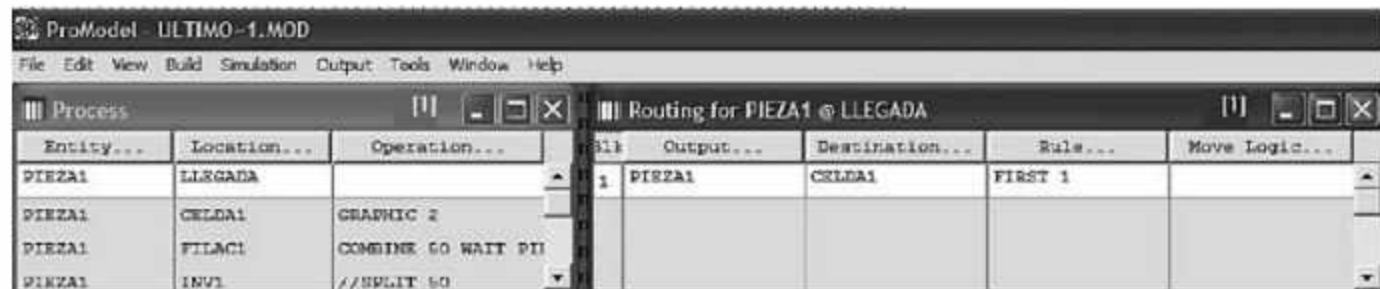


Figura 6.31 Acceso a las reglas de ruteo.

La columna **Rule** permite seleccionar la condición que se debe cumplir para que una entidad sea transferida desde la localización actual (definida en la columna **Location**) hasta la localización siguiente (determinada en la columna **Destination**).

Al oprimir el botón **Rule** se abre el cuadro de diálogo **Routing Rule**. En él se puede seleccionar cualquiera de las siguientes reglas de ruteo:

First available. Selecciona la primera localización que tenga capacidad disponible.

By turn. Rota la selección entre las localizaciones que estén disponibles.

If Join Request. Selecciona la localización que solicite una entidad para un proceso de unión (requiere el uso de la instrucción JOIN).

If Send. Selecciona la localización que solicite una entidad para un proceso de envío (requiere el uso de la instrucción SEND).

Until Full. Selecciona la localización hasta que esté llena.

Random. Selecciona aleatoriamente y en forma uniforme alguna de las localizaciones disponibles.

Most Available. Selecciona la localización que tenga la mayor capacidad disponible.

If Load Request. Selecciona la localización que solicite una entidad para un proceso de carga (requiere el uso de la instrucción LOAD).

Longest Unoccupied. Selecciona la localización que tenga el mayor tiempo desocupada.

If Empty. Selecciona la localización solamente cuando está vacía, y continuará seleccionándola hasta que esté llena.

Probability. Selecciona la localización de acuerdo con un porcentaje asignado.

User Condition. Selecciona la localización que satisfaga una condición booleana especificada por el usuario.

Continue. Se mantiene en la localización para realizar operaciones adicionales.

As Alternate. Selecciona la localización como alternativa si está disponible y ninguna de las reglas anteriores se cumple.

As Backup. Selecciona una localización como respaldo si la que tiene preferencia está descompuesta.

Dependent. Selecciona una localización solamente si la ruta inmediata anterior ya fue ejecutada.

Quantity. Es un campo adicional que permite definir el número de unidades que saldrán de esta localización por cada unidad que entre. Esta instrucción se utiliza para procesos de corte y separación.

Ejemplo 6.5

El proceso de manufactura ilustrado en la figura 6.32 consta de 2 tornos y un almacén donde las piezas esperan antes de ser procesadas. Los tiempos de proceso son de 12 y 15 min/pieza en los tornos 1 y 2, respectivamente. La tasa de entrada a este proceso es de 6 piezas/h con distribución de Poisson. Simule el proceso.

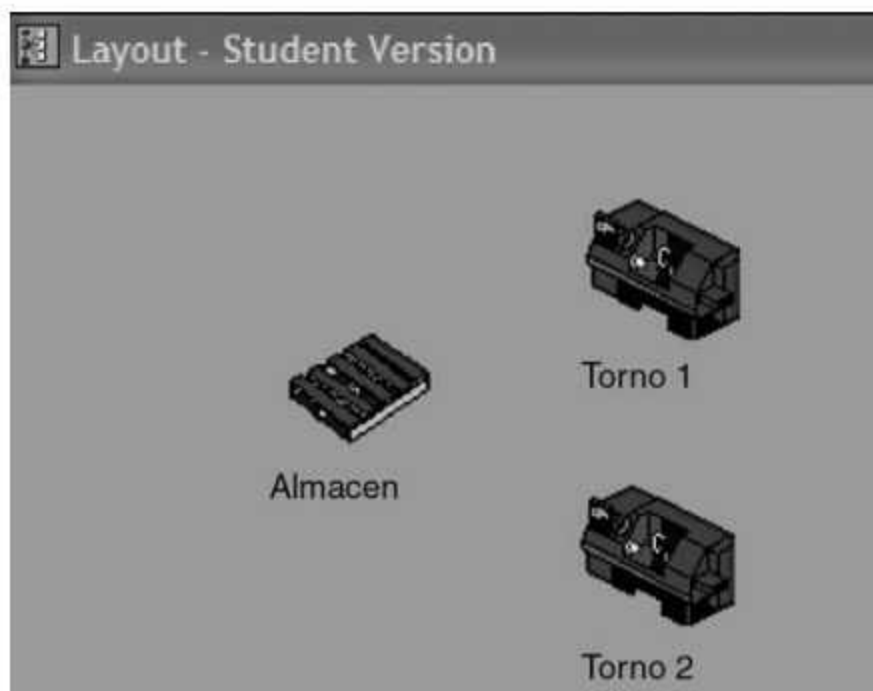


Figura 6.32
Esquemización de un proceso de torneado.

- Para empezar, definiremos las localizaciones *Almacén*, *Torno_1* y *Torno_2* en la ventana **Locations (Build / Locations)**. Especifique la capacidad de dichas localizaciones como se muestra en la figura 6.33.




Locations				
Icon	Name	Cap.	Units	DTs...
	Torno_1	1	1	None
	Torno_2	1	1	None
	Almacen	infinite	1	None

Figura 6.33 Definición de capacidad de las localizaciones.

- Defina la entidad *Pieza* en la ventana **Entities (Build / Entities)**, como se muestra en la figura 6.34.

Entities		
Icon	Name	Speed (fpm)
	Pieza	150

Figura 6.34 Definición de la entidad.

A continuación programaremos las llegadas de la *Pieza* al *Almacén*. Abra la ventana **Arrivals (Build / Arrivals)** y especifique los parámetros que se muestran en la figura 6.35.

Arrivals					
Entity...	Location...	Qty each...	First Time	Occurrences	Frequency
Pieza	Almacen	1	0	INF	E(10) min

Figura 6.35 Entrada de materia prima (*Pieza*) al punto inicial del proceso (*Almacén*).

- Despliegue la ventana **Processing (Build / Processing)** para crear la ruta de producción de las piezas a través de cualquiera de los 2 tornos (vea la figura 6.36).



Figura 6.36 Selección del torno donde se procesará la pieza.

Como puede ver en la figura 6.36, en el campo **Rule** aparece, de manera predeterminada, el valor **FIRST** para ambas salidas. En este caso la regla de ruteo indica que la pieza deberá moverse al primer torno que se encuentre disponible. Si ambos están disponibles, el torno 1 será la opción elegida. El "1" a la derecha del primer **FIRST** indica el valor del campo **Quantity** del cuadro de diálogo **Routing Rule** (vea la figura 6.37).

- Haga clic en el botón **Rule** de la ventana **Routing for** para desplegar el cuadro de diálogo **Routing Rule** (figura 6.37), en donde podrá seleccionar alguna de las reglas de ruteo para el registro activo, en este caso para *Torno_1*.

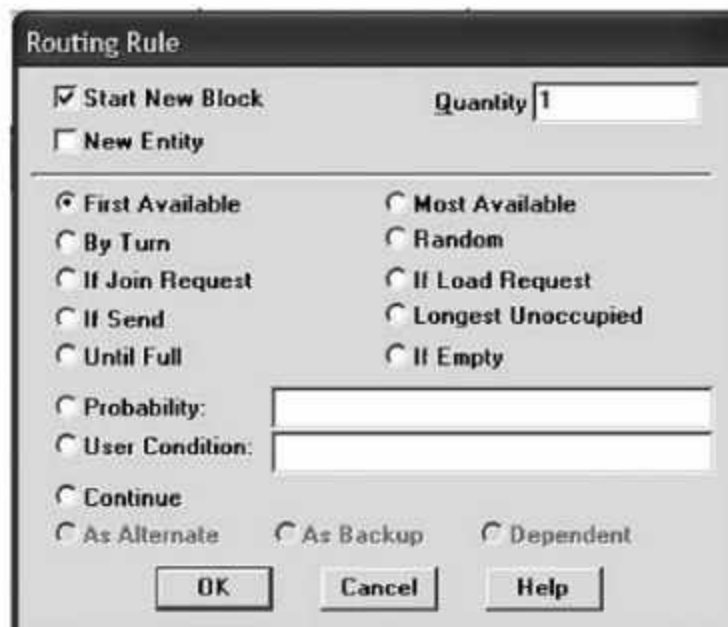


Figura 6.37 Seleccione en este cuadro de diálogo la regla de transferencia entre localizaciones.

Por ejemplo si desea cambiar la regla de manera que 30% de las piezas sean procesadas en el torno 1 y el resto en el torno 2:

- Abra primero la ventana **Routing Rule** para el torno 1, seleccionamos la opción **Probability**, y escriba 0.3 en el cuadro de texto correspondiente. Haga clic en **OK**.
- Repita los pasos anteriores para el torno 2, tenga cuidado de que la opción **Start New Block** esté desmarcada en este caso. Elija **Probability** y escriba 0.7 en el campo de texto. La figura 6.38 muestra parte de este procedimiento.

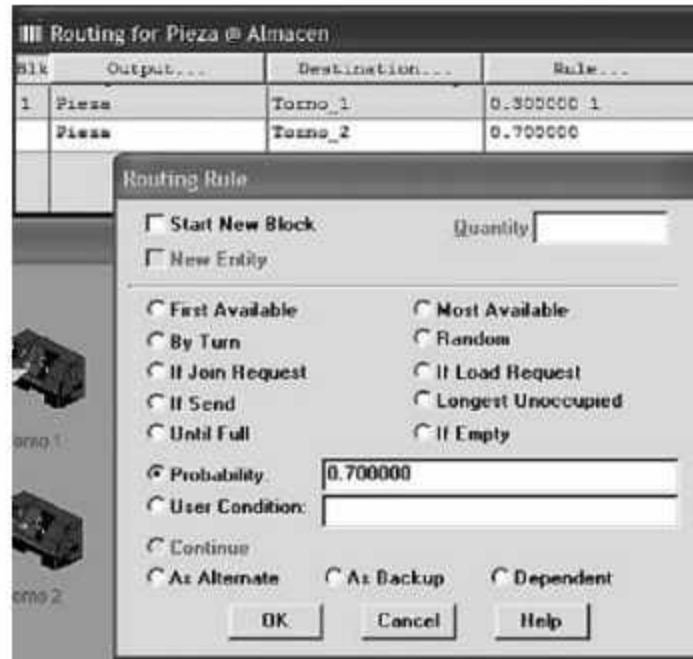


Figura 6.38 Selección del torno en función de porcentajes.

Podemos seleccionar de manera similar cualquier otra regla de ruteo. La sintaxis de programación del ejemplo (manteniendo la regla probabilística) se muestra en la figura 6.39:

```

Formatted Listing of Model:
C:\ProMod4\models\rutas.MOD
.....
Time Units:                Minutes
Distance Units:            Feet
.....
Locations
.....
Name      Cap      Units  Stats      Rules      Cost
-----
Torno_1   1          1      Time Series Oldest, ,
Torno_2   1          1      Time Series Oldest, ,
Almacen   infinite 1      Time Series Oldest, ,
.....
Entities
.....
Name      Speed (fpm)  Stats      Cost
-----
Pieza     150         Time Series
.....
Processing
.....
Process          Routing
Entity  Location Operation  Blk  Output  Destination Rule      Move Logic
-----
Pieza   Almacen                1   Pieza   Torno_1   0.300000 1
Pieza   Torno_1 wait 12    1   Pieza   Torno_2   0.700000
Pieza   Torno_2 wait 15    1   Pieza   EXIT     FIRST 1
.....
Arrivals
.....
Entity  Location Qty each  First Time Occurrences Frequency  Logic
-----
Pieza   Almacen  1          0          INF       E(10) min

```

Figura 6.39 Sintaxis completa de la programación para el ejemplo 6.5.

6.5 Ensamblajes, acumulación y agrupamiento de piezas

Un gran número de procesos de producción incluyen operaciones de separación, ensamblaje, agrupamiento y unión. A continuación se resumen varias de las funciones que ProModel tiene para modelar este tipo de procesos.

Instrucción ACCUM

La instrucción **ACCUM**, cuya sintaxis general es

ACCUM <Cantidad>,

retrasa el proceso de las entidades hasta que cierta cantidad del mismo tipo se haya acumulado en algún lugar del modelo. Una vez que se han acumulado, se permite que todas ellas continúen en forma separada. Si la entidad a procesar se define con el parámetro **ALL**, todas las entidades, sin importar su tipo, entran en el proceso de acumulación. Esta instrucción no consolida las entidades (vea la figura 6.40).

Ejemplos:

```
ACCUM 5
ACCUM 10
```

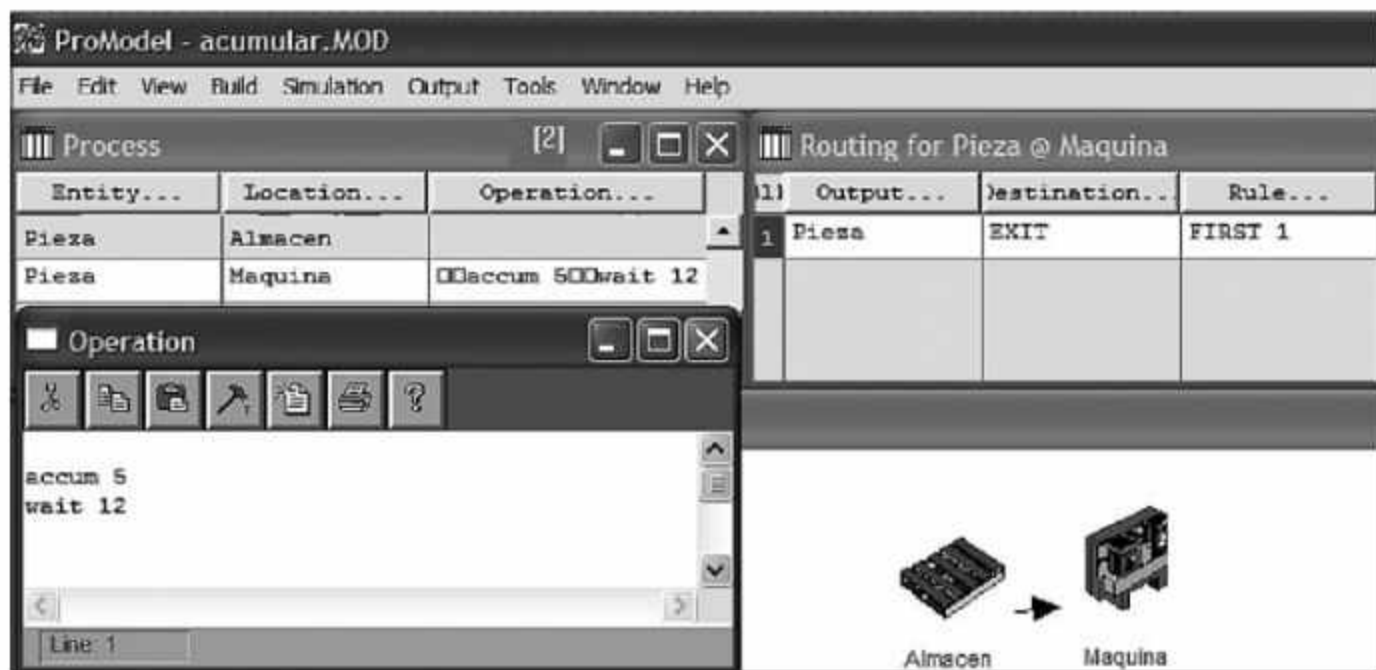


Figura 6.40 Ejemplo de un proceso de manufactura de 12 minutos, que inicia cuando se han acumulado 5 piezas en la máquina.

Instrucción GROUP

La sintaxis general de esta instrucción es:

GROUP <Cantidad> AS <Nueva entidad> ,

y su propósito es realizar una unión temporal de entidades de un mismo tipo (a menos que se utilice como parámetro la palabra **ALL**, en cuyo caso puede unir entidades de diferente tipo). Las entidades agrupadas se mantienen en ese estado hasta que encuentran una instrucción **UNGROUP**. Si se utiliza la opción **AS**, la entidad resultante de la unión puede tener un nombre diferente (vea la figura 6.41).

Ejemplos:

```
GROUP 10
GROUP 10 AS Amigos
GROUP PROD() AS Lote
```

The screenshot shows the ProModel interface with two main tables: 'Process' and 'Routing for Lote @ Almacen'. Below the tables is a diagram showing a flow from 'Almacen' to 'Maquina'. At the bottom, there is a text-based summary of the process and routing data.

Entity...	Location...	Operation...	Output...	Destination...	Rule...
Pieza	Almacen	group 5 as lote			
Lote	Almacen				
Lote	Maquina	wait 10			

Blk	Output	Destination	Rule
1	Lote	Maquina	FIRST 1


```

*****
*                               Processing                               *
*****
          Process                               Routing
Entity  Location Operation          Blk  Output  Destination Rule
-----  -
Pieza   Almacen  group 5 as lote
Lote    Almacen
Lote    Maquina  wait 10
    
```

Figura 6.41 Ejemplo de un sistema de producción de 10 min/lote, con agrupación previa de 5 piezas en el almacén (incluye la sintaxis de programación completa del proceso).

Instrucción UNGROUP

La instrucción **UNGROUP** tiene la siguiente sintaxis general:

UNGROUP

El objetivo de esta instrucción es separar las entidades agrupadas previamente con la instrucción **GROUP**. Las entidades son desagrupadas de acuerdo con el esquema "primero en entrar, primero en salir", a menos que se indique lo contrario con la expresión **LIFO** (vea la figura 6.42).

Ejemplos:

UNGROUP
UNGROUP LIFO

The screenshot shows the ProModel interface for a simulation model named 'desagrupar.MOD'. It features several windows:

- Process Table:**

Entity...	Location...	Operation...
Pieza	Almacen	group 5 as lote
Lote	Almacen	
Lote	Maquina	wait 10 ungroup
Pieza	Maquina	
- Routing Table:**

Output...	Destination..	Rule...
- Operation Window:** Shows the command 'wait 10 ungroup'.
- Diagram:** A flow diagram showing 'Almacen' (represented by a circuit board icon) leading to 'Maquina' (represented by a machine icon).
- Simulation Output:** A text window showing the following data:


```

*****
*                               Processing                               *
*****
          Process                               Routing
Entity  Location Operation                    Blk  Output  Destination Rule
-----  -
Pieza   Almacen  group 5 as lote                            1
Lote    Almacen
Lote    Maquina  wait 10 ungroup                           1  Lote   Maquina   FIRST 1
Pieza   Maquina
      
```

Figura 6.42 Ejemplo de un sistema de producción de 10 min/lote, agrupando previamente 5 piezas en el almacén y desagrupando después del proceso.

Instrucción COMBINE

La sintaxis general de la instrucción es:

COMBINE <Cantidad> AS <Nueva entidad>

Su propósito es realizar una unión definitiva de entidades de un mismo tipo, a menos que se utilice como parámetro la palabra **ALL**, en cuyo caso puede unir entidades de diferente tipo. La entidad resultante mantiene el nombre y atributos de la última entidad combinada; si se utiliza el parámetro **AS**, la entidad resultante de la unión puede tener un nombre diferente (vea la figura 6.43).

Ejemplos:

COMBINE 10
COMBINE 50 AS *Lote*

Entity...	Location...	Operation...
Pieza	Almacen	combine 10 as caja
caja	Almacen	

Output...	Destination..	Rule...
1 caja	EXIT	FIRST 1

```

*****
*                               Processing                               *
*****
          Process                               Routing
-----
Entity  Location  Operation          Blk  Output  Destination  Rule
-----
Pieza   Almacen   combine 10 as caja          1   caja    EXIT         FIRST 1
caja    Almacen
    
```

Figura 6.43 Ejemplo de un sistema de empaque en un almacén, combinando 10 piezas/caja (incluye parte de la sintaxis de programación del modelo).

Instrucción JOIN

Con la sintaxis general:

JOIN <Cantidad> <Entidad a ensamblar>

la instrucción **JOIN** ensambla entidades de cierto tipo a la entidad actual. Las entidades a ensamblar provienen de otra localización, y se dirigen mediante la regla de ruteo **IF JOIN REQUEST**. Una vez ensambladas, las entidades pierden su identidad y atributos.

Ejemplos:

```
JOIN 24 Refrescos
JOIN Variable_1 Cajas
```

Ejemplo 6.6

Como se ilustra en la figura 6.44, un operario empaqueta cajas de 25 piezas cada una. El tiempo de empaque es exponencial de 1 minuto. Las tasas de entrada por hora son constantes, de 1500 en el caso de las piezas, y de 60 en el de las cajas.



Figura 6.44 Proceso de empaque para el ejemplo 6.6.

- Para comenzar, defina las localizaciones *Almacén_Caja*, *Almacén_Pieza* y *Operario* en la ventana **Locations (Build / Locations)**.
- Especifique las entidades *Caja_vacía*, *Caja_llena* y *Pieza* en la ventana **Entities (Build / Entities)**.

- En la ventana **Arrivals (Build / Arrivals)** determine las entradas al sistema, de la *Caja_vacía* al *Almacén_Caja*, y de la *Pieza* al *Almacén_Pieza*.
- En la ventana **Processing (Build / Processing)**, cree la ruta de cada tipo de entidad.

Para modelar la unión de las entidades son necesarios dos procesos: uno para la entidad principal, *Caja_vacía*, y otro para la entidad secundaria, *Pieza*. El primer proceso se ilustra en la figura 6.45; en él haremos uso de la instrucción **JOIN** para indicar que la entidad actual, *Caja_vacía*, en la localización *Operario* solicita 25 entidades *Pieza* para llevar a cabo la unión. La entidad resultante de este proceso se identifica como *Caja_llena*, y es enviada al exterior.

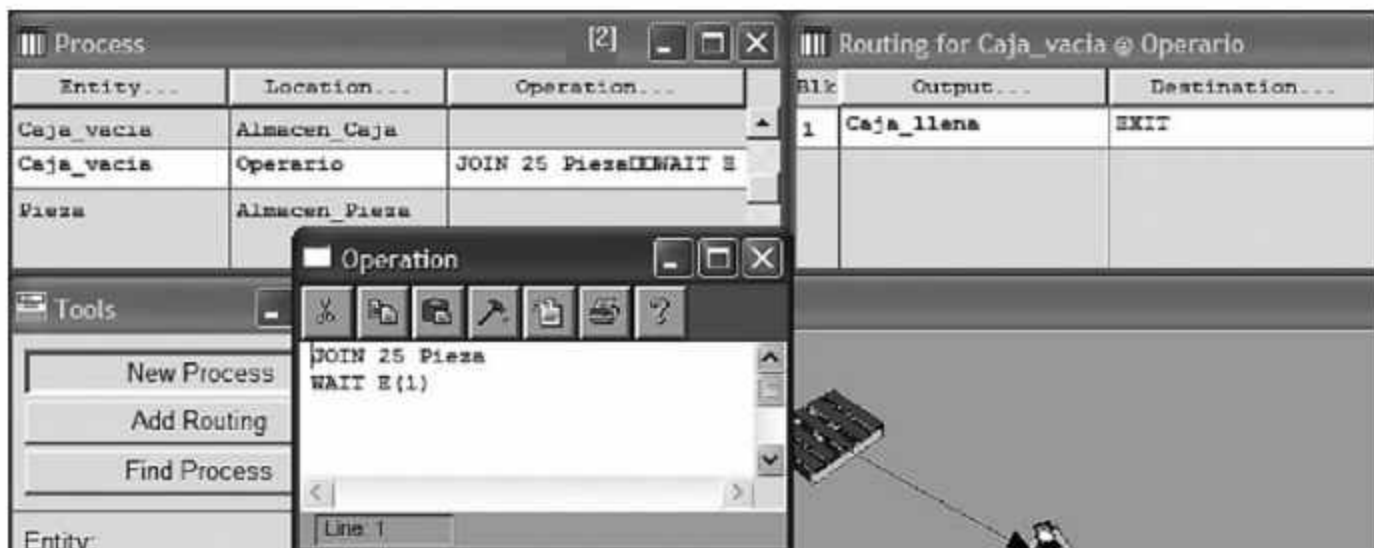


Figura 6.45 Programación de la instrucción JOIN.

El segundo proceso a programar es para la entidad *Pieza*, y consiste en modificar la regla de ruteo hacia la localización *Operario*. En la figura 6.46 se muestra cómo llevarlo a cabo mediante la selección de la opción **If Join Request** en el cuadro de diálogo **Routing Rule**.

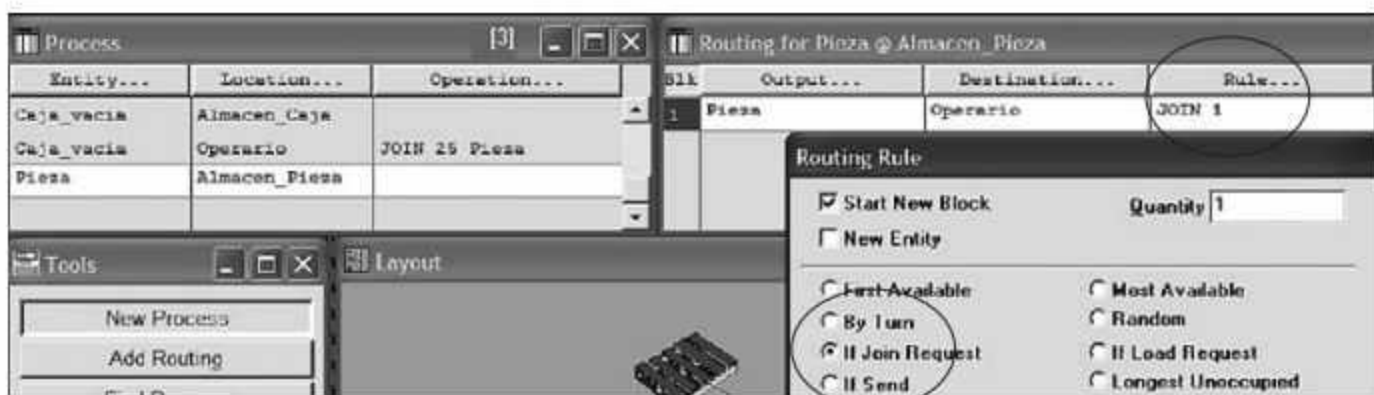


Figura 6.46 Programación de la regla de ruteo If Join Request.

La sintaxis de programación del ejemplo 6.6 se ilustra en la figura 6.47.

```

Formatted Listing of Model:
C:\ProMod4\models\Ensamble.MOD

```

```

Time Units:           Minutes
Distance Units:      Feet

```

```

Locations

```

Name	Cap	Units	Stats	Rules	Cost
Operario	1	1	Time Series	Oldest, ,	
Almacen_Caja	inf	1	Time Series	Oldest, ,	
Almacen_Pieza	inf	1	Time Series	Oldest, ,	

```

Entities

```

Name	Speed (fpm)	Stats	Cost
Caja_vacia	150	Time Series	
Pieza	150	Time Series	
Caja_llena	150	Time Series	

```

Processing

```

Entity	Location	Process		Routing		
		Operation	Blk	Output	Destination	Rule
Caja_vacia	Almacen_Caja		1	Caja_vacia	Operario	FIRST 1
Caja_vacia	Operario	JOIN 25 Pieza WAIT E(1)	1	Caja_llena	EXIT	FIRST 1
Pieza	Almacen_Pieza		1	Pieza	Operario	JOIN 1

```

Arrivals

```

Entity	Location	Qty each	First Time	Occurrences	Frequency	Logic
Caja_vacia	Almacen_Caja	1	0	INF	1/60 hr	
Pieza	Almacen_Pieza	1	0	INF	1/1500 hr	

Figura 6.47 Sintaxis completa de la programación para el ejemplo 6.6.

Instrucción LOAD

La instrucción **LOAD**, cuya sintaxis general es

LOAD <Cantidad> IFF <Condición> IN <Tiempo> ,

agrupa temporalmente entidades de cierto tipo a la entidad actual. Las entidades a agrupar provienen de otra localización, y se dirigen mediante la regla de ruteo **If Load Request**. Las entidades agrupadas mantienen su identidad y atributos, y se separan cuando encuentran una instrucción **UNLOAD**.

Ejemplos:

```
LOAD 5
LOAD 10 IFF Tipo > 2
LOAD 2 IFF Peso > 3 IN 5 min
LOAD 5 IN 10 min
```

Instrucción UNLOAD

La sintaxis general de esta instrucción es:

UNLOAD <Cantidad> IFF <Expresión>

y su propósito es permitir que una entidad descargue o desagrupe otras entidades previamente agrupadas por una instrucción **LOAD** o **GROUP** si se cumple la condición especificada.

Ejemplos:

```
UNLOAD 10
UNLOAD 2 IFF entity() = Pieza
```

Ejemplo 6.7

Un proceso requiere mover material en contenedores de un lugar a otro. Cada contenedor debe llevar 5 piezas. El tiempo para cargar el contenedor es de 1 minuto. Los contenedores se mueven a través de una banda transportadora en 30 segundos. Al final de este movimiento se separan y cada entidad continúa por separado en bandas transportadoras independientes. Vea la esquematización de este problema en la figura 6.48.

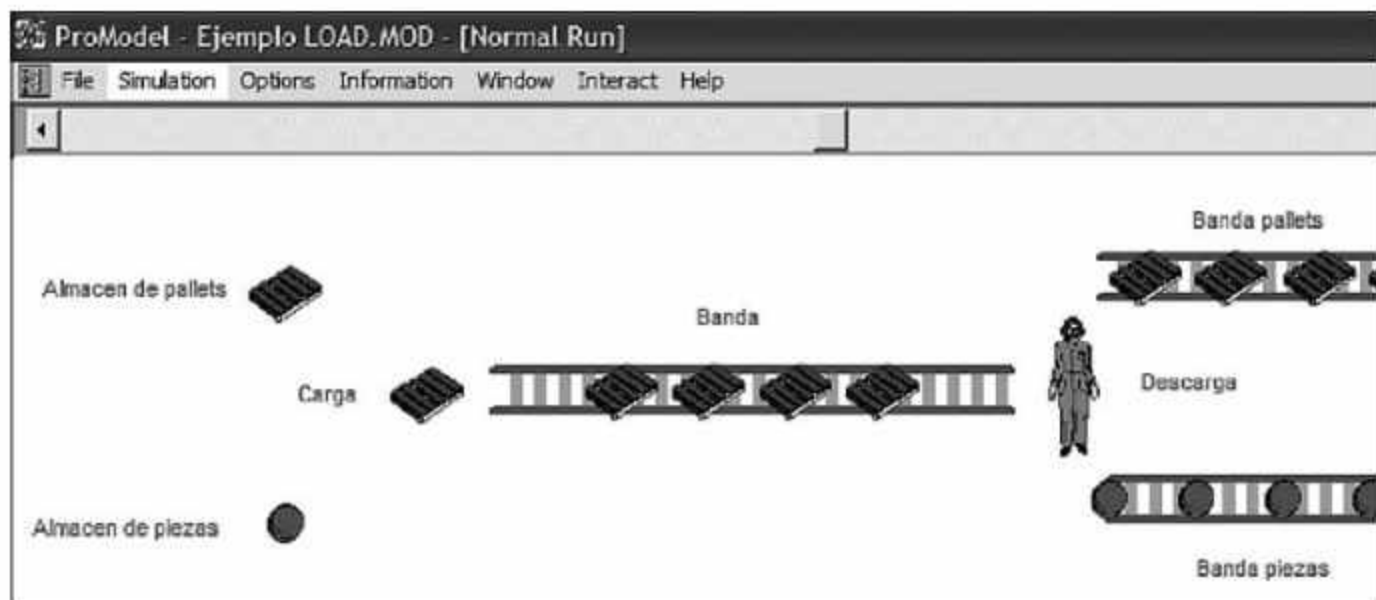


Figura 6.48 Esquematización de la simulación de un proceso de unión y separación de entidades.

Para este ejemplo, antes que nada es necesario definir las siete localizaciones que se muestran en la figura 6.48 (almacén de piezas, almacén de pallets, carga, banda, banda pallets, descarga y banda piezas), las dos entidades que se mueven a través del sistema (*pallet* y *piezas*), y la información sobre la llegada del material al sistema. Estos elementos y sus características se describen más adelante en la sintaxis de programación del modelo.

De la misma forma que en el caso de la instrucción **JOIN**, para lograr la unión o carga se requieren dos procesos: uno para la entidad principal, *Pallet*, y otro para la entidad secundaria, *Piezas*.

El primero de dichos procesos se muestra en la figura 6.49. En él se hace uso de la instrucción **LOAD** para indicar que la entidad actual, *Pallet*, solicita en la localización *Carga* 5 entidades para llevar a cabo la unión, incluyendo el tiempo necesario para hacerlo. La entidad resultante de este proceso es enviada hacia la banda transportadora *Banda*.

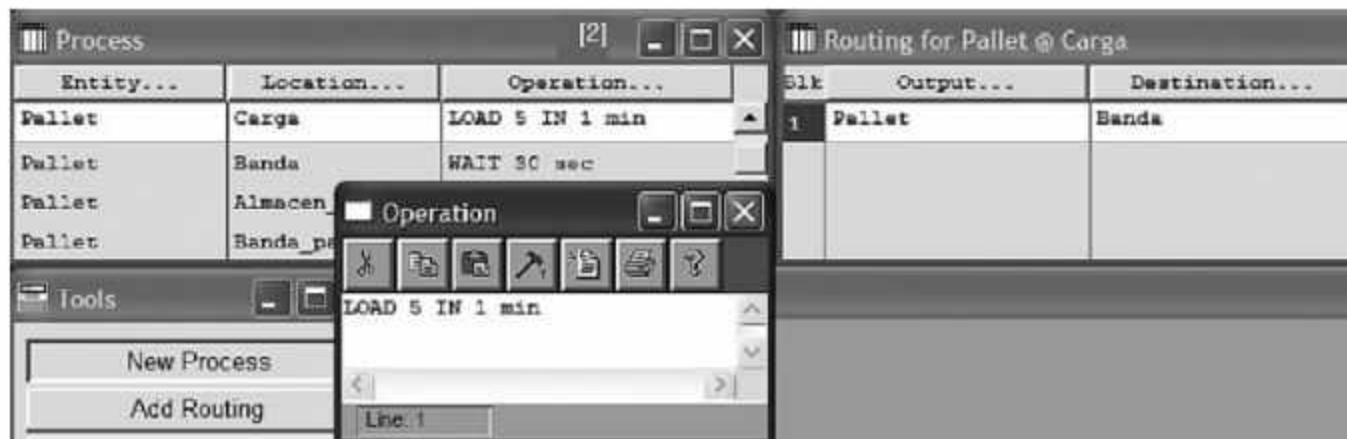


Figura 6.49 Programación de la instrucción LOAD.

El segundo proceso a programar es para la entidad *Piezas*, y consiste en modificar la regla de ruteo desde el *Almacen_de_piezas* hacia la localización *Carga*. La figura 6.50 ilustra cómo se lleva esto a cabo mediante la selección de la opción **If Load Request** en el cuadro de diálogo **Routing Rule**.

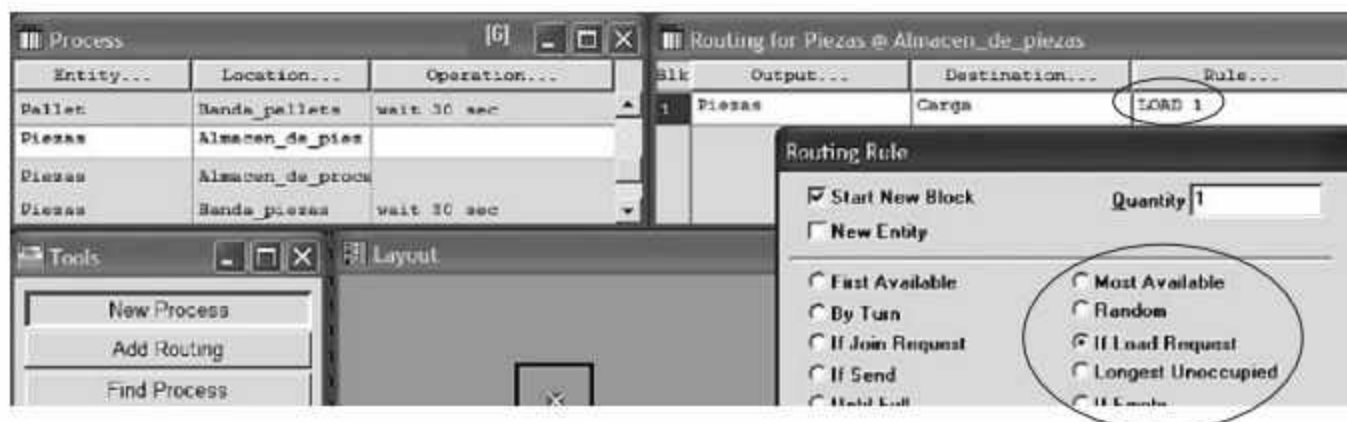


Figura 6.50 Programación de la regla de ruteo If Load Request.

Una vez transportado el *Pallet* por la banda, se lleva a cabo el proceso de separación o descarga mediante la instrucción **UNLOAD**, de manera que cada entidad sea enviada a su siguiente proceso (vea la figura 6.51).

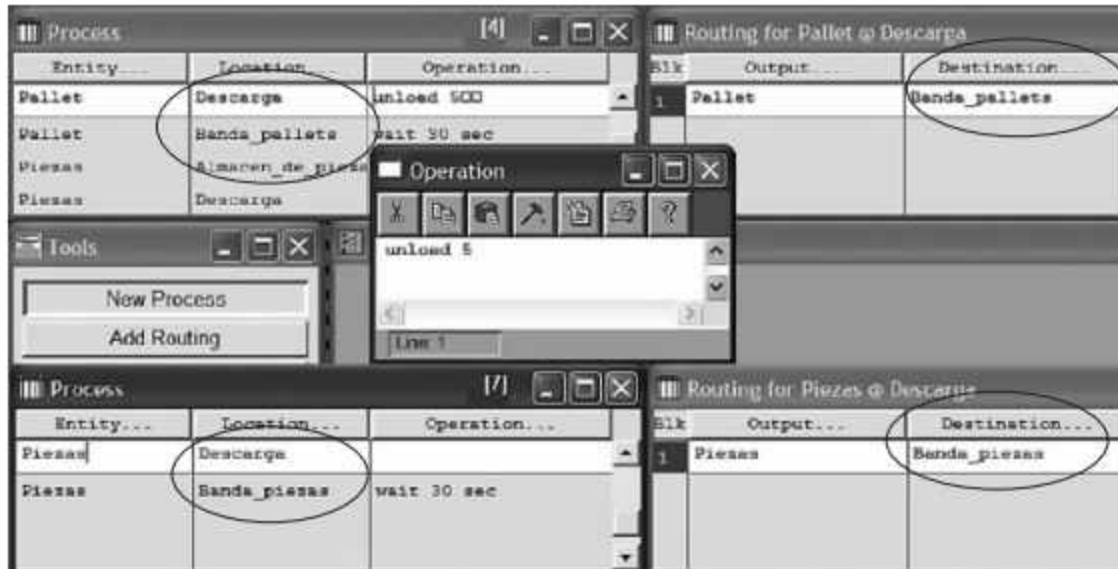


Figura 6.51 Programación de la instrucción UNLOAD.

Observe la sintaxis de programación completa para el modelo del ejemplo 6.7 en la figura 6.52.

```

*****
Formatted Listing of Model:
C:\ProMod4\models\Ejemplo LOAD.MOD
*****

Time Units:           Minutes
Distance Units:      Feet

*****
Locations
*****

Name          Cap    Units  Stats  Rules  Cost
-----
Almacen de pallets  INFINITE  1    Time Series Oldest, .
Almacen de piezas  INFINITE  1    Time Series Oldest, .
Banda         INFINITE  1    Time Series Oldest, FIFO,
Descarga       1         1    Time Series Oldest, .
Carga          1         1    Time Series Oldest, .
Banda_pallets  INFINITE  1    Time Series Oldest, FIFO,
Banda_piezas   INFINITE  1    Time Series Oldest, FIFO,

*****
Entities
*****

Name      Speed (fpm)  Stats  Cost
-----
Piezas    150          Time Series
Pallet    150          Time Series

*****
Processing
*****

Process                                Routing
Entity  Location      Operation      Blk Output  Destination  Rule
-----
Pallet  Almacen de pallets  LOAD 5 IN 1 min  1  Pallet  Carga      FIRST 1
Pallet  Carga          WAIT 30 sec    1  Pallet  Banda       FIRST 1
Pallet  Banda         unload 5       1  Pallet  Descarga    FIRST 1
Pallet  Descarga      wait 30 sec    1  Pallet  Banda_pallets  FIRST 1
Pallet  Banda_pallets  EXIT          1  Pallet  EXIT        FIRST 1
Piezas  Almacen de piezas  LOAD 1         1  Piezas  Carga       LOAD 1
Piezas  Descarga      wait 30 sec    1  Piezas  Banda_piezas  FIRST 1
Piezas  Banda_piezas  EXIT          1  Piezas  EXIT        FIRST 1

*****
Arrivals
*****

Entity  Location      Qty each  First Time Occurrences  Frequency  Logic
-----
Piezas  Almacen de piezas  1000     0      1      1
Pallet  Almacen de pallets  200      0      1      1
    
```

Figura 6.52 Sintaxis completa de la programación para el ejemplo 6.7.

Instrucción SPLIT

La sintaxis general de la instrucción **SPLIT** es:

SPLIT <Cantidad> AS <Nueva entidad>

Su propósito es dividir una entidad en varias entidades, y asignar a cada una de ellas (de manera opcional) un nuevo nombre. Las entidades resultantes de la división mantienen los mismos atributos que la entidad original.

Ejemplos:

SPLIT 10
 SPLIT 20 AS *Botella*
 SPLIT Variable_4 AS *Láminas*

El uso de esta instrucción se recomienda solamente cuando existan operaciones adicionales a las entidades dentro de la localización actual. En caso contrario es mejor utilizar la estructura de reglas de ruteo que se ilustra en la figura 6.53 para definir la cantidad de entidades a separar.

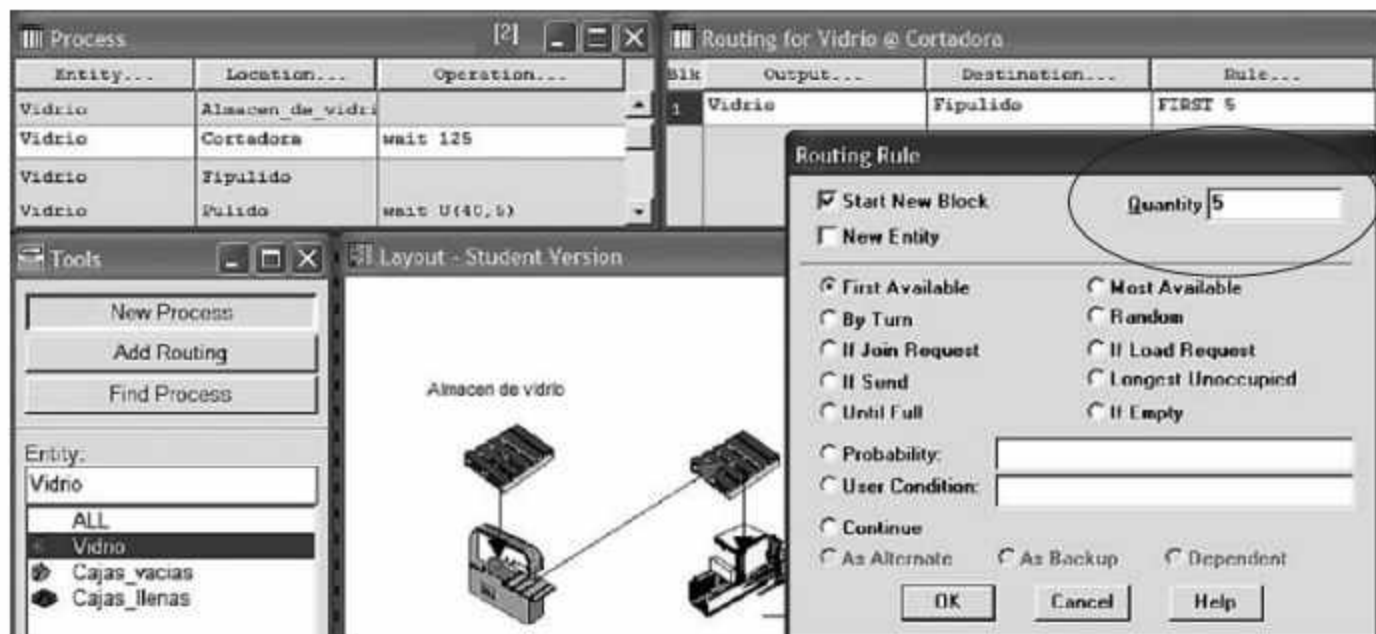


Figura 6.53 Separación de entidades mediante la opción de cantidad disponible en el cuadro de diálogo Routing Rule.

Ejemplo 6.8

A la cortadora de la figura 6.54 llegan rollos a una tasa de 5 por hora. El proceso de corte es de 30 segundos, y de cada rollo se obtienen 10 láminas, las cuales se envían al siguiente proceso a través de una banda transportadora.

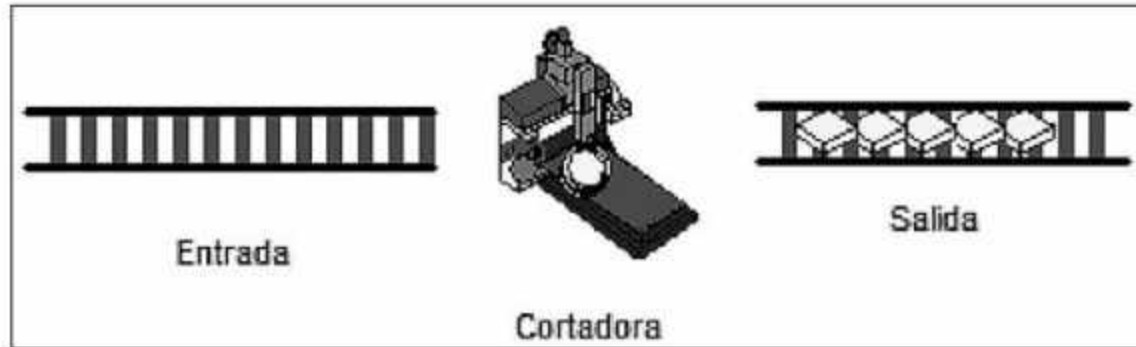


Figura 6.54
Proceso de corte.

El modelo requiere definir 3 localizaciones: *Entrada*, con capacidad infinita; *Cortadora*, con capacidad 1, y *Salida*, con capacidad infinita. Además, es preciso especificar dos entidades, llamadas *Rollo* y *Lámina*, y las llegadas de rollos al proceso con un tiempo entre llegadas de 12 minutos. La descripción completa de los elementos mencionados se encuentra en la sintaxis de programación del modelo, que se presentará más adelante.

El punto importante de este modelo se ilustra en la figura 6.55, y ocurre al crear la ruta de proceso cuando el *Rollo*, después de un tiempo de operación de 30 segundos, se transfiere hacia la localización *Salida*. Es en el cuadro de diálogo **Routing Rule** donde determinamos el valor de la variable **Quantity** en 10.

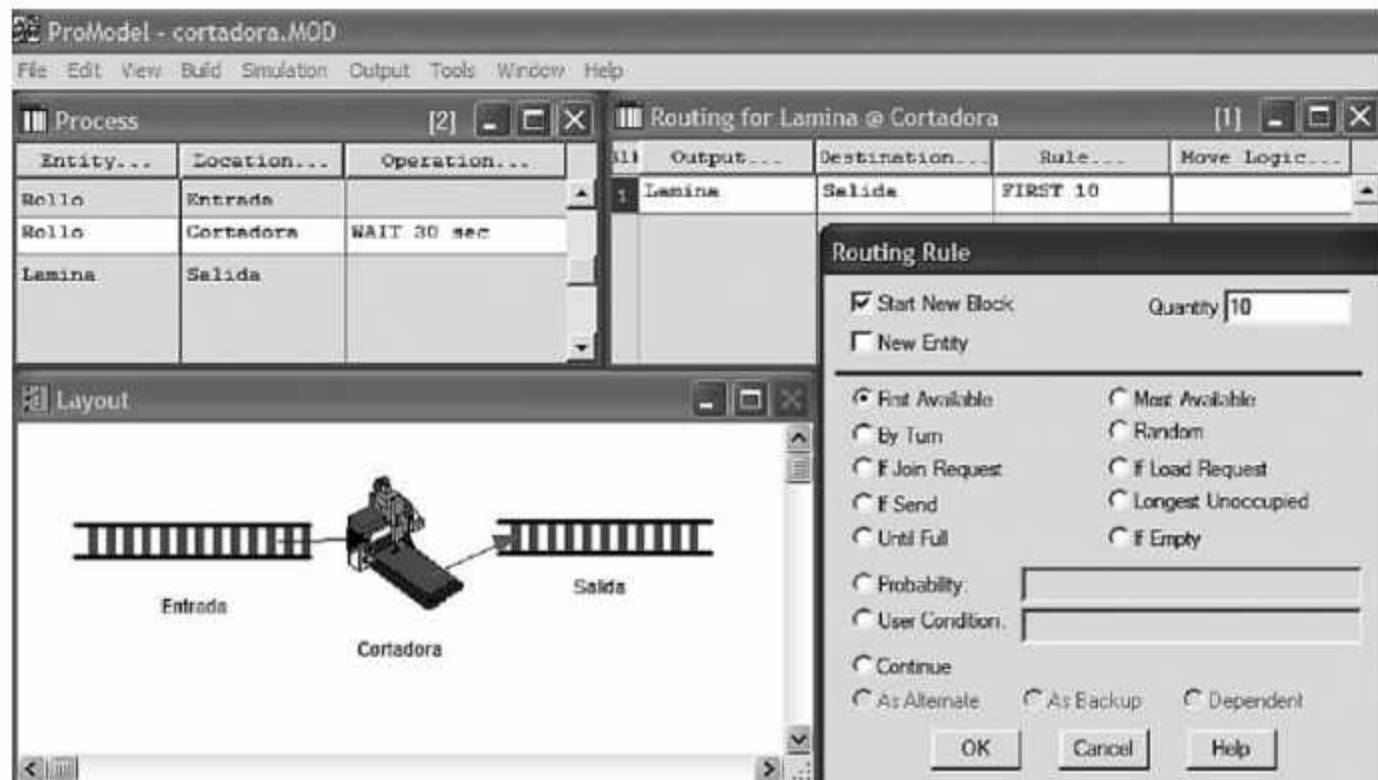


Figura 6.55 Uso del parámetro Quantity para simular procesos de corte y separación.

La sintaxis de programación completa del ejemplo 6.8 se muestra en la figura 6.56.

```

Formatted Listing of Model:
C:\Archivos de programa\ProModel\Models\cortadora.MOD
=====
Time Units:                Minutes
Distance Units:           Feet
Initialization Logic:     ANIMATE 5
=====
Locations
=====
Name      Cap      Units  Stats      Rules      Cost
-----
Entrada   INFINITE 1      Time Series Oldest, FIFO,
Cortadora 1      1      Time Series Oldest,
Salida    INFINITE 1      Time Series Oldest, FIFO,
=====
Entities
=====
Name      Speed (fpm)  Stats      Cost
-----
Rollo     150          Time Series
Lamina    150          Time Series
=====
Processing
=====
Process                                Routing
Entity  Location  Operation      Blk  Output  Destination Rule  Move Logic
-----
Rollo   Entrada  WAIT 30 sec    1    Rollo   Cortadora   FIRST 1
Rollo   Cortadora  Split 10 as Lamina 1    Lamina  Salida     FIRST 10
Lamina  Salida   EXIT           1    Lamina  EXIT       FIRST 1
=====
Arrivals
=====
Entity  Location  Qty each  First Time Occurrences Frequency  Logic
-----
Rollo   Entrada  1         0          INF       12 min

```

Figura 6.56 Sintaxis general del programa de modelado para el ejemplo 6.8.

Como se mencionó anteriormente, también es posible modelar este proceso con el uso de la instrucción **SPLIT**. Para hacerlo se deben realizar cambios sólo en la ruta del proceso. En la figura 6.57 se muestra la parte del programa que sufre modificaciones.

```

Processing
=====
Process                                Routing
Entity  Location  Operation      Blk  Output  Destination Rule  Move Logic
-----
Rollo   Entrada  WAIT 30 sec    1    Rollo   Cortadora   FIRST 1
Rollo   Cortadora  Split 10 as Lamina 1    Lamina  Salida     FIRST 10
Lamina  Salida   EXIT           1    Lamina  EXIT       FIRST 1
Lamina  Salida   EXIT           1    Lamina  EXIT       FIRST 1

```

Figura 6.57 Uso de la instrucción SPLIT para el ejemplo 6.8.

Instrucción ROUTE

La sintaxis general de esta instrucción es:

ROUTE <Bloque de ruteo>

La instrucción **ROUTE** envía la entidad hacia el proceso especificado por el número del bloque.

Ejemplos:

```
ROUTE 2
ROUTE Tipo_de_pieza
```

Ejemplo 6.9

Simule el proceso de separación de tres tipos de pieza (vea la figura 6.58). El 20% de las piezas son de tipo 1, el 50% son de tipo 2, y el resto son de tipo 3. El tiempo de transporte es de 3 minutos en las bandas de entrada, de 3 minutos en las bandas de salida de las piezas tipo 1 y 3, y de 5 minutos en la banda de salida de las piezas tipo 2. El tiempo entre llegadas al sistema es de 1 minuto/pieza, distribuido exponencialmente.

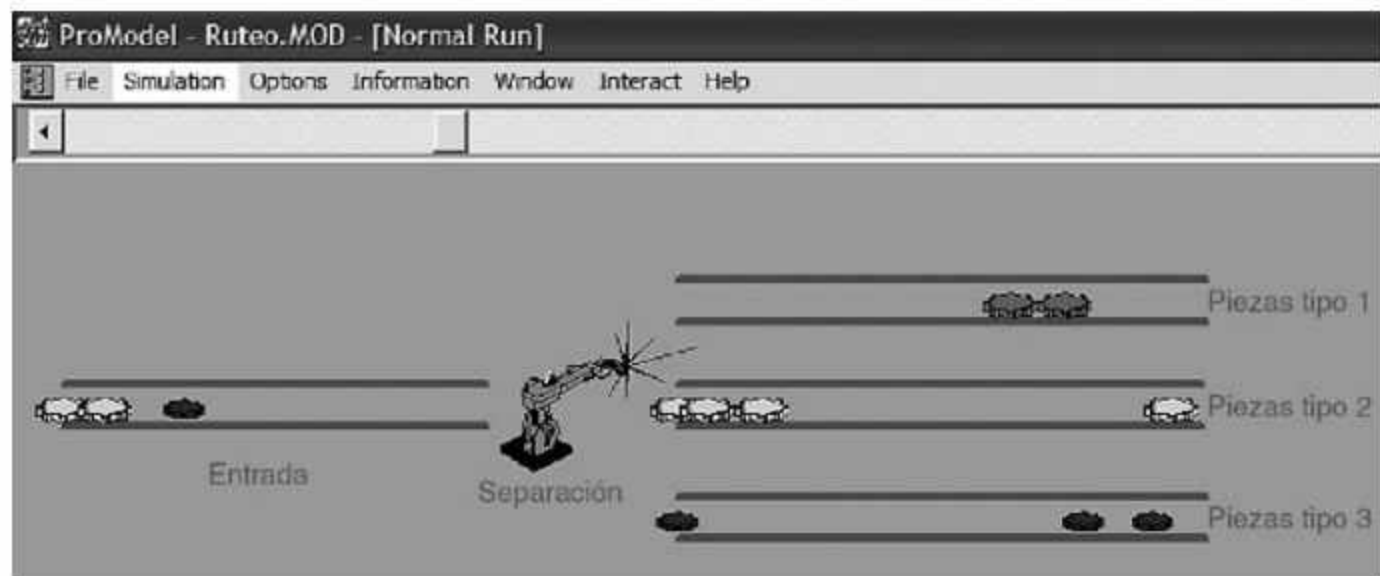


Figura 6.58 Proceso de separación de materiales.

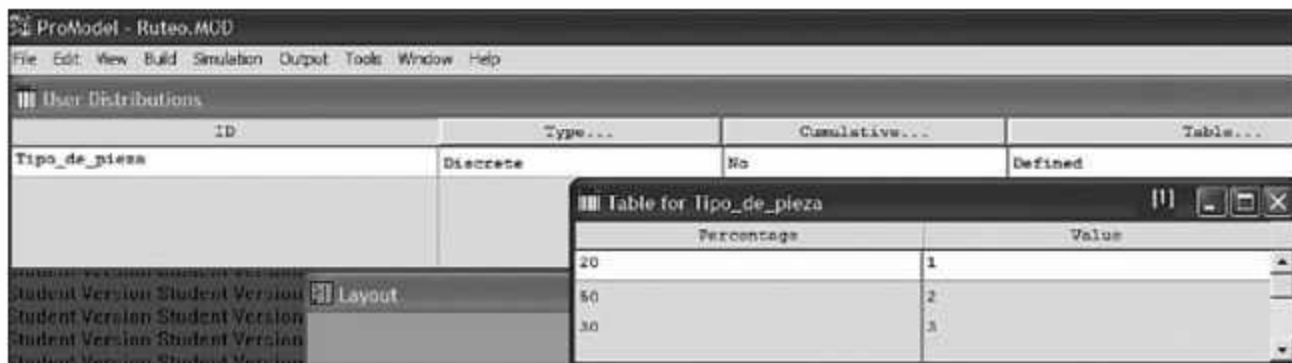
- Defina las localizaciones *Entrada*, *Separacion*, *Banda_1*, *Banda_2* y *Banda_3* en la ventana **Locations (Build / Locations)**, siguiendo el esquema de la figura 6.58.
- Defina la entidad *Gear* en la ventana **Entities (Build / Entities)**, especifique tres formas diferentes para que cada entidad pueda diferenciarse por su color (revise el procedimiento descrito en el ejemplo 5.3 de la sección 5.4.3).
- Abra la ventana **Attributes (Build / Attributes)** para definir el atributo *Ruta* que identifique la trayectoria que seguirá cada tipo de pieza (vea la figura 6.59).



ID	Type...	Classification...
Ruta	Integer	Ent

Figura 6.59 Identificación del atributo RUTA.

- En la ventana **User Distributions (Build / User Distributions)**, defina una tabla similar a la que se muestra en la figura 6.60 para determinar la mezcla de producto donde la variable *Tipo_de_pieza* tomará los valores 1, 2, 3, de acuerdo con los porcentajes definidos.

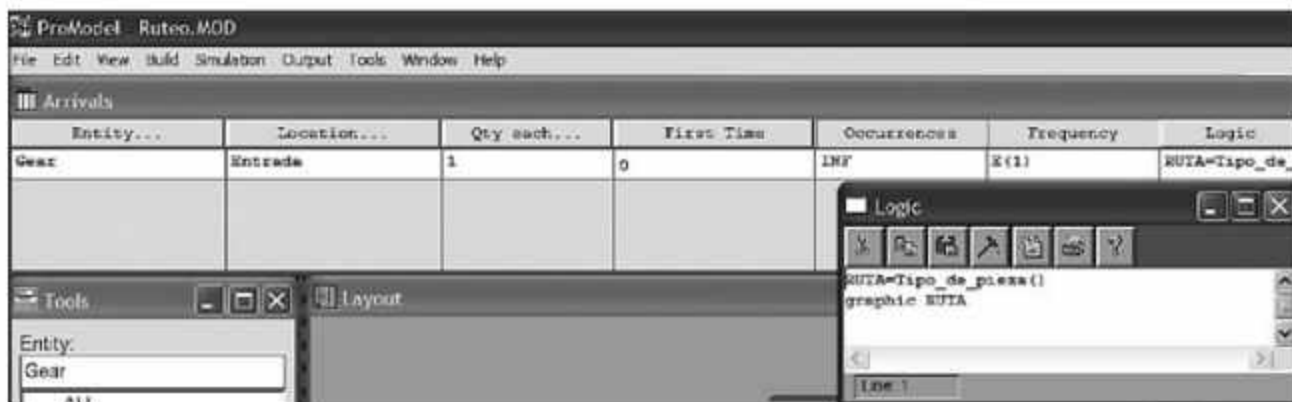


ID	Type...	Cumulative...	Table...
Tipo_de_pieza	Discrete	No	Defined

Table for Tipo_de_pieza	
Percentage	Value
20	1
50	2
30	3

Figura 6.60 Definición de la distribución de usuario *Tipo_de_Pieza*.

- Haga clic en el botón **Logic** de la ventana **Arrivals (Build / Arrivals)** para desplegar la ventana **Logic** y asignar ahí los atributos de cada pieza (vea la figura 6.61). Indique, en la columna **Frequency** de la misma figura, el tiempo entre llegadas: exponencial con media de 1 minuto/pieza. Al llegar cualquier pieza se evaluará la función *Tipo_de_Pieza*, y se asigna un valor al atributo RUTA y a la forma de la entidad (**graphic RUTA**). Estos valores permanecerán como característica de la entidad hasta que sea destruida.



Entity...	Location...	Qty each...	Fixed Time	Occurrences	Frequency	Logic
Gear	Entrada	1	0	INF	E(1)	RUTA=Tipo_de_


```

Logic
RUTA=Tipo_de_pieza()
graphic RUTA
  
```

Figura 6.61 Asignación de atributos a la entidad.

- Abra la ventana **Processing (Build / Processing)** para definir la ruta de la entidad *Gear* a través de las localizaciones. Tenga cuidado al crear el proceso de separación

que se ilustra en la figura 6.62; observe que en la localización *Separación* existen tres posibles rutas de salida, identificadas en el campo *Blk* como 1, 2, 3. Para lograrlo es necesario marcar la casilla de verificación de la opción **Start a New Block** en el cuadro de diálogo **Routing Rule**.



Figura 6.62 Segmento de la ruta de la localización Separación.

La sintaxis completa de la programación para el ejemplo 6.9 se muestra en la figura 6.63.

```

=====
*
* Locations
*
-----
Name      Cap      Units  Stats      Rules      Cost
-----
Separacion 1      1      Time Series Oldest,
Banda_1   INFINITE 1      Time Series Oldest, FIFO,
Banda_2   INFINITE 1      Time Series Oldest, FIFO,
Banda_3   INFINITE 1      Time Series Oldest, FIFO,
Entrada   INFINITE 1      Time Series Oldest, FIFO,
=====
*
* Entities
*
-----
Name      Speed (fpm)  Stats      Cost
-----
Gear      150          Time Series
=====
*
* Processing
*
-----
Process          Routing
-----
Entity Location Operation  Blk Output Destination Rule Move Logic
-----
Gear  Entrada  wait 3      1  Gear  Separacion  FIRST 1
Gear  Separacion ROUTE RUTA  1  Gear  Banda_1     FIRST 1
                                           2  Gear  Banda_2     FIRST 1
                                           3  Gear  Banda_3     FIRST 1
Gear  Banda_1  wait 3      1  Gear  EXIT        FIRST 1
Gear  Banda_2  wait 5      1  Gear  EXIT        FIRST 1
Gear  Banda_3  wait 3      1  Gear  EXIT        FIRST 1
=====
*
* Arrivals
*
-----
Entity Location Qty each  First Time Occurrences Frequency Logic
-----
Gear  Entrada  1      0      INF      E(1)  RUTA=Tipo_de_pieza()
                                           graphic RUTA
=====
*
* Attributes
*
-----
ID      Type      Classification
-----
Ruta    Integer  Entity
=====
*
* User Distributions
*
-----
ID      Type      Cumulative Percentage Value
-----
Tipo_de_pieza Discrete No      20      1
                                           50      2
                                           30      3
=====

```

Figura 6.63 Sintaxis completa de la programación del modelo para el ejemplo 6.9.

6.6 Transporte entre estaciones

En ProModel, el modelado del transporte de entidades a través de un sistema se lleva a cabo gracias a los recursos dinámicos y a la creación de rutas de transporte. El procedimiento es el siguiente:

- Definir la ruta y sus propiedades en la ventana **Path Network (Build / Path Network)**.
- Determinar el recurso en la ventana **Resource (Build / Resource)**, y asociarlo con la ruta creada previamente.
- Programar en la ventana **Processing (Build / Processing)** la captura del recurso con la instrucción **GET**, el uso del recurso con las instrucciones **MOVE WITH** o **MOVE FOR**, y la liberación del recurso con la instrucción **FREE**.

Ejemplo 6.10

Al inicio del día entran 100 barriles de 200 litros a un almacén de material en proceso. Estos barriles deben ser transportados hacia un proceso de inspección, en donde el operario, llamado Casimiro, inspecciona el producto en 5 minutos con distribución exponencial. Debido al tamaño de los barriles, un montacargas debe realizar el movimiento del producto desde el almacén hasta donde se encuentra Casimiro. El tiempo de transporte es de 2 a 3 minutos con distribución uniforme. Observe el esquema del problema en la figura 6.64.

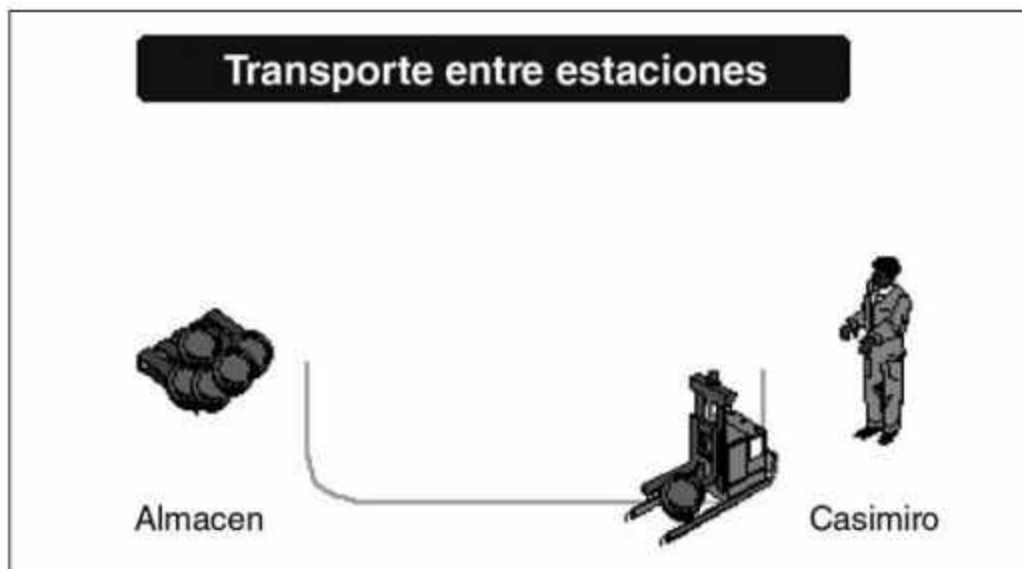


Figura 6.64
Esquematización del ejemplo 6.10.

Tomando como base el esquema de la figura 6.64, debemos:

- Definir en la ventana **Locations (Build / Locations)** dos localizaciones: *Almacén* y *Casimiro*.
- Especificar en la ventana **Entities (Build / Entities)** la entidad *Barril*.

- Definir en la ventana **Arrivals (Build / Arrivals)** las características de las llegadas del *Barril* al *Almacén*.
- En la ventana **Path Network (Build / Path Network)**, active una ventana de edición para especificar la trayectoria por donde se moverá el recurso. Al crear la ruta deberá establecer los valores de los siguientes parámetros:

Graphic. Permite seleccionar el color de la trayectoria, y ésta será visible o no durante la simulación. Si es una grúa viajera permitirá especificar sus colores, la separación del puente y la representación gráfica.

Name. Nombre de la trayectoria.

Type. Permite definir la posibilidad de rebasar dentro de la trayectoria. Seleccione la opción **Crane** si la trayectoria será usada por una grúa viajera.

T/S. Permite determinar si el movimiento es con base en el tiempo o en la velocidad.

Paths. Permite crear y editar la trayectoria y sus nodos.

Interfaces. Permite asociar los nodos con las localizaciones.

Mapping. Permite realizar el mapeo de destinos y rutas.

Nodes. Se crean automáticamente al crear la trayectoria.

Para crear la ruta y sus nodos siga este procedimiento:

- Haga clic en el botón **Paths** de la ventana **Path Networks**.
- Haga clic con el botón izquierdo del ratón sobre el punto del layout donde desee marcar el inicio de la ruta. Luego haga clic de manera sucesiva para determinar cambios de dirección. Haga clic con el botón derecho del ratón en el punto donde quiera fijar un nodo. Si desea seguir agregando nodos, repita el procedimiento anterior a partir del último nodo que haya determinado (vea la figura 6.65).
- Defina el resto de las características de la trayectoria. Como en este ejemplo deseamos mover el recurso de acuerdo con el tiempo, active la opción **Time** en la columna **T/S** y, en la ventana de edición **Paths**, establezca un tiempo U(2.5,0.5) entre el nodo *N1* y *N2* en la columna **Time**.

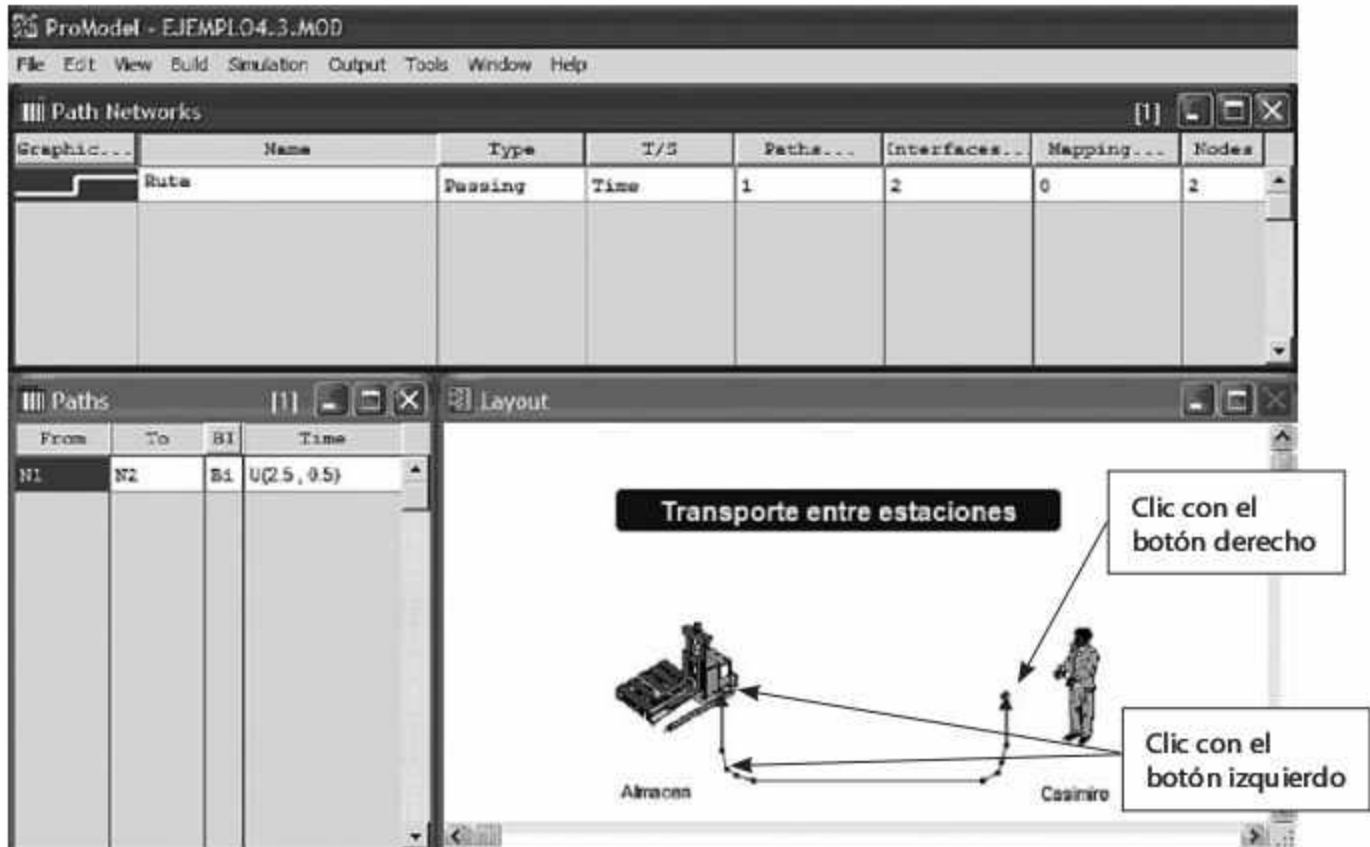


Figura 6.65 Creación de la ruta del montacargas.

- Terminado este proceso, haga clic en el botón **Interfaces** para abrir el cuadro de diálogo que se muestra en la figura 6.66. Asocie cada nodo con su localización respectiva: en este caso, asocie el nodo *N1* con *Almacén*, y el nodo *N2* con *Casimiro*.



Figura 6.66 Registre en este cuadro de diálogo la interfaz entre cada nodo y la localización asociada.

En la figura 6.67 vemos finalizado el proceso de creación de la trayectoria. Los círculos señalan la asociación; en el layout ésta se identificará mediante una línea punteada entre el nodo y la localización.

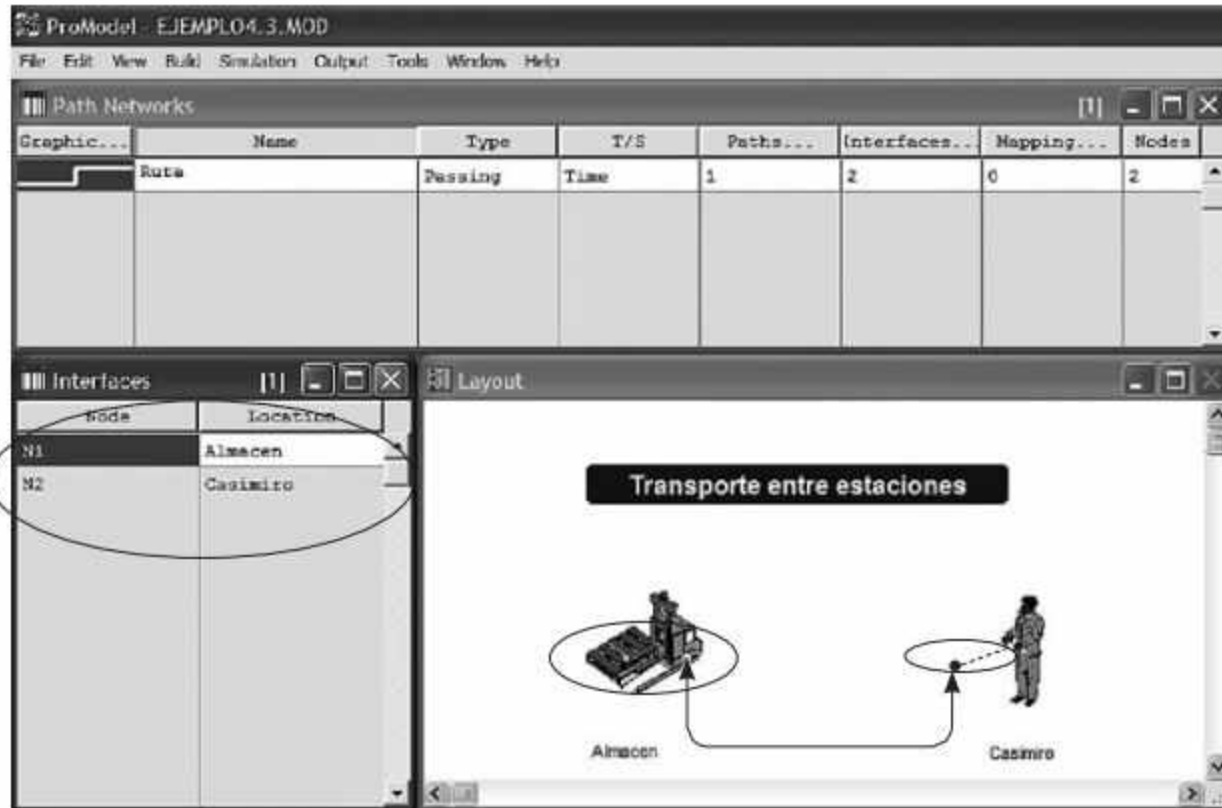


Figura 6.67 Definición final de la ruta.

Para continuar el modelado del problema:

- Abra la ventana **Resources (Build / Resources)** para especificar el recurso *Montacargas* y la cantidad de unidades que contiene (vea la figura 6.68).

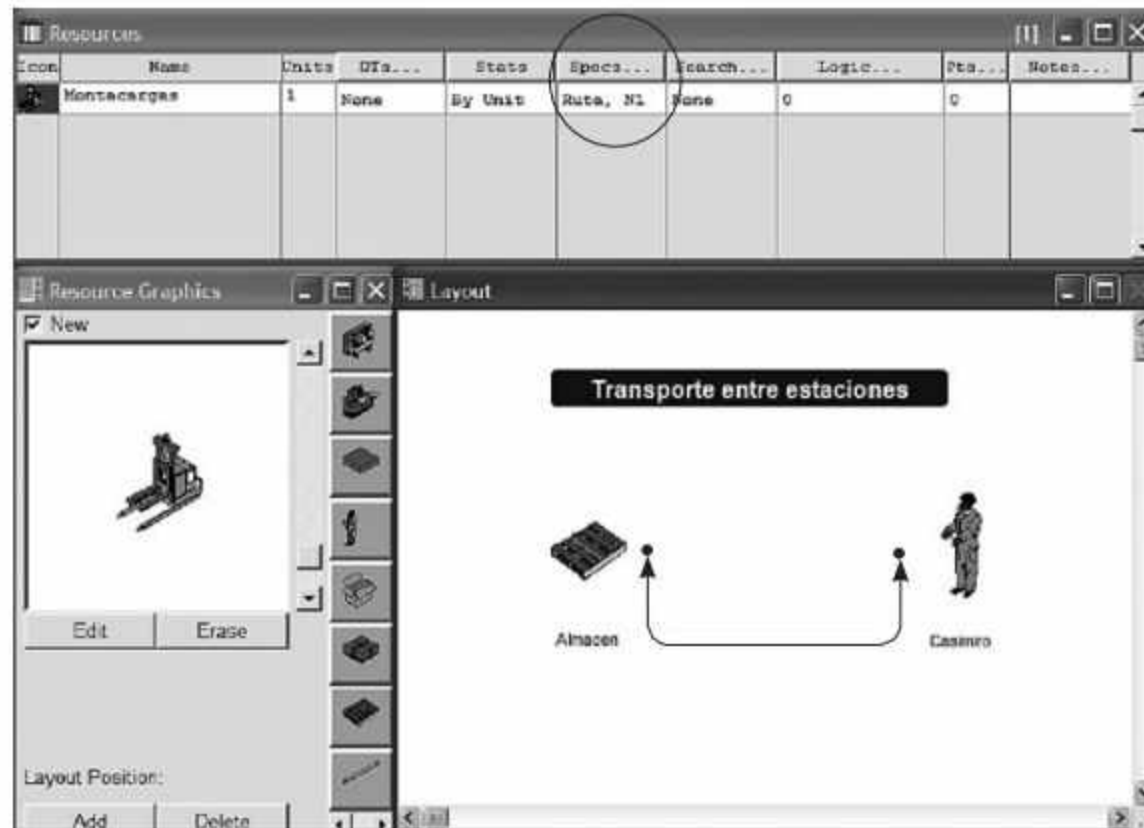


Figura 6.68 Definición del recurso Montacargas.

- Haga clic en el botón **Specs** (identificado con un círculo en la figura 6.68) para asociar el recurso con la ruta. Enseguida se abrirá el cuadro de diálogo **Specifications** (vea la figura 6.69), en donde deberá seleccionar la ruta por la que se moverá el recurso *Montacargas*. En este ejemplo, la *Ruta* tendrá asociado un nodo que será la base del recurso; también determinaremos si cuando dicho recurso esté ocioso deberá regresar a ese punto. Asimismo, podemos especificar otras propiedades, como velocidad de movimiento, aceleración, desaceleración, etcétera. (En este ejemplo dichos campos quedan en blanco, ya que el movimiento entre los nodos se definió en función del tiempo y no en función de la velocidad.)



Figura 6.69
Especificaciones del recurso y su asociación con la ruta.

- Abra la ventana **Processing (Build / Processing)** para crear la ruta, considere los detalles de la figura 6.70 que consisten en la captura del recurso en el *Almacén*, su movimiento hacia *Casimiro*, y su liberación final.

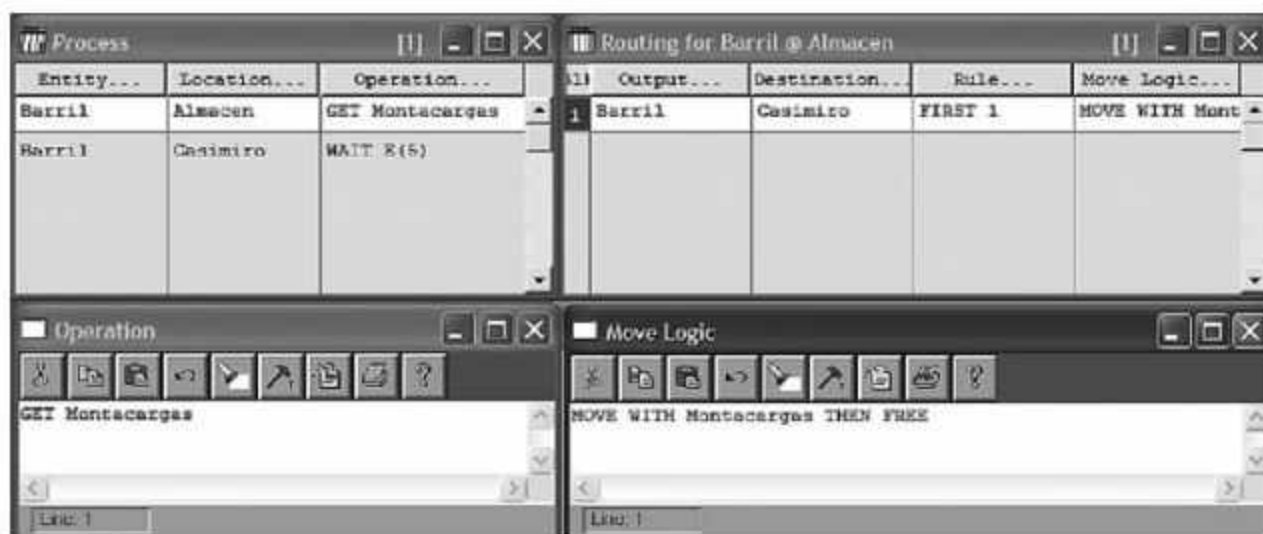


Figura 6.70 Segmento de la programación de la captura, uso y liberación del recurso dentro del proceso.

La sintaxis de programación del modelo para el ejemplo 6.10 se muestra en la figura 6.71.

```

Formatted Listing of Model:
C:\Archivos de programa\ProModel\Modelo\EJEMPLO4.S.MOD
*****
Time Units:           Minutes
Distance Units:      Feet
*****
Locations
*****
Name      Cap Units Stats      Rules      Cost
-----
Almacen   INF 1      Time Series Oldest, ,
Casimiro  1 1      Time Series Oldest, , First
*****
Entities
*****
Name      Speed (fpm) Stats      Cost
-----
Barril    150      Time Series
*****
Path Networks
*****
Name      Type      T/S      From      To      BI      Dist/Time      Speed Factor
-----
Ruta      Passing   Time     N1        N2        Bi      U(2.5,0.5)
*****
Interfaces
*****
Net      Node      Location
-----
Ruta     N1        Almacen
         N2        Casimiro
*****
Resources
*****
Name      Units Stats      Res      Ent
Search Search Path      Motion      Cost
-----
Montacargas 1      By Unit Closest Oldest Ruta
Home: N1
*****
Processing
*****
Process Routing
Entity Location Operation      Blk Output Destination Rule      Move Logic
-----
Barril  Almacen GET Montacargas 1  Barril  Casimiro  FIRST 1  MOVE WITH
Montacargas THEN FREE
Barril  Casimiro WAIT E(5) 1  Barril  EXIT  FIRST 1
*****
Arrivals
*****
Entity Location Qty Each      First Time Occurrences Frequency Logic
-----
Barril  Almacen 100 0 1

```

Figura 6.71 Sintaxis completa del modelo para el ejemplo 6.10.

ProModel permite también la construcción de sistemas de manejo de materiales a través de grúas viajeras. Estos sistemas son modelados mediante el uso de los módulos de **Path Network** y **Resources**.

Ejemplo 6.11

En un punto denominado bodega 1, en un patio de maniobras, se colocan 400 contenedores de 2 toneladas al inicio del día. Estos contenedores deben ser transportados hacia dos lugares; bodega 2 y bodega 3, en donde se depositan para su posterior operación. El 40% de los contenedores se envían a la bodega 2 y el resto a la bodega 3. Debido a su tamaño, una grúa debe realizar el movimiento entre las bodegas. La distancia entre la bodega 1 y la 2 es de 40 metros, y entre la bodega 1 y la 3 es de 50 metros. La velocidad de la grúa cuando viaja sin contenedor es de 50 metros por minuto y con contenedor es de 20 metros por minuto. El tiempo para elevar el contenedor es de 40 segundos y para depositarlo es de 50 segundos. Este sistema se puede observar en la figura 6.72.

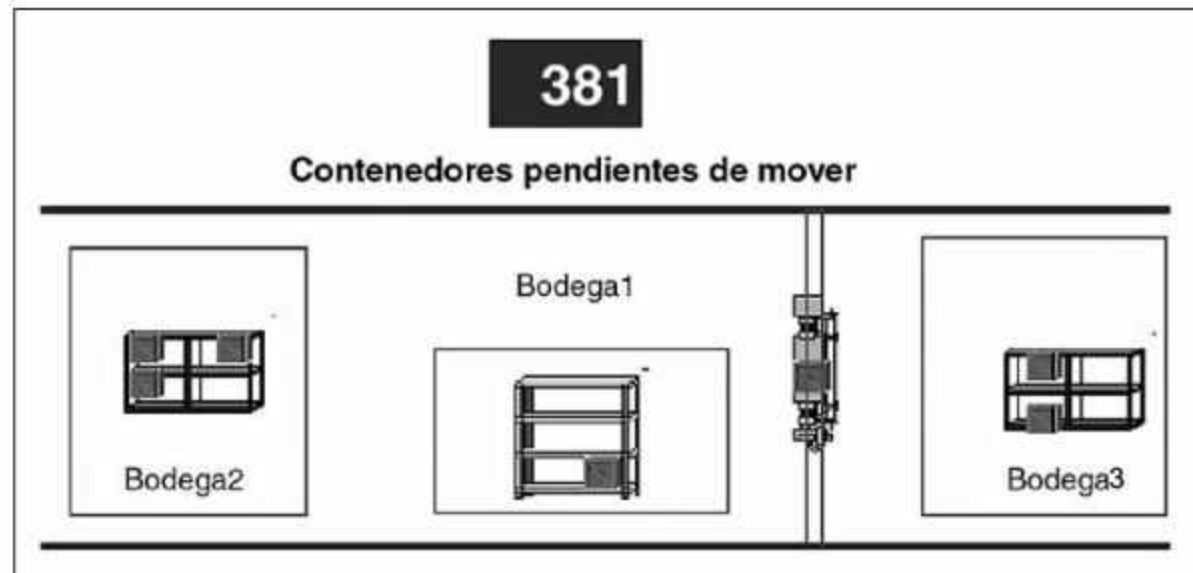


Figura 6.72 Esquematización del ejemplo 6.11.

Si se toma como base el esquema de la figura 6.72, debemos:

- Definir en la ventana **Locations (Build / Locations)** tres localizaciones: *Bodega1*, *Bodega2* y *Bodega3*.
- En la ventana **Entities (Build / Entities)** dé de alta la entidad *Contenedor*.
- Especifique en la ventana **Arrivals (Build / Arrivals)** las características de las llegadas del *Contenedor* a la *Bodega1*.
- Para crear la ruta y los nodos, en la opción **Path Network (Build / Path Network)**, active la ventana de edición para especificar la trayectoria por donde se moverá el recurso. Al crear la ruta deberá establecer los valores de los siguientes parámetros:

Graphic. Especifique los colores, la separación del puente y la representación gráfica de los rieles.

Name. Establezca el nombre a la trayectoria como *Red*.

Type. Seleccione la opción **Crane**. Aparecerán sobre el layout un par de rieles conectados por el puente sobre el que se moverá la grúa viajera y se activará la opción **Nodes**. Con el ratón podrá mover y cambiar las dimensiones de los rieles moviendo cualquiera de los 4 puntos extremos. Estos puntos denominados **Origin**, **Rail1End**, **BridgeEnd** y **Rail2End** representan las fronteras de movimiento de la grúa. Al redimensionar el tamaño de los rieles, las localizaciones deberán quedar dentro del área. En la ventana de edición **Nodes** modifique las dimensiones de estos puntos en la columna **Coords(R,B)** para que coincidan con las dimensiones del sistema, en este caso el área de movimiento será de 100 metros de largo por 50 de ancho.

Con el botón izquierdo del ratón agregue tres nodos *N1*, *N2* y *N3* dentro del área de los rieles e indique en la columna **Coords(R,B)** las coordenadas de estos nodos para definir la distancia entre cada uno de ellos de acuerdo a las dimensiones de este sistema, vea la figura 6.73 y observe que la distancia entre el nodo 1 y el 2 es de 40 metros y entre el nodo 2 y el 3 es de 50 metros.



Figura 6.73 Creación de los rieles de movimiento.

Interfaces: Haga clic en el botón **Interfaces** para abrir el cuadro de diálogo que se muestra en la figura 6.74 y asocie cada nodo con su localización respectiva: en este caso, asocie el nodo *N1* con *Bodega2*, el nodo *N2* con *Bodega1* y el nodo *N3* con *Bodega3*



Node	Location
N1	Bodega2
N2	Bodega1
N3	Bodega3

Figura 6.74 Registre en este cuadro de diálogo la interfaz entre cada nodo y la localización asociada.

T/S. Las grúas viajeras solamente se mueven con base en la velocidad por lo que este campo permanece inactivo.

Paths. Esta opción permanecerá desactivada.

Mapping. Esta opción permanecerá desactivada.

Para continuar el modelado del problema:

- Abra la ventana **Resources (Build / Resources)** para especificar el recurso *Grúa* y la cantidad de unidades que contiene es 1 (vea la figura 6.75).

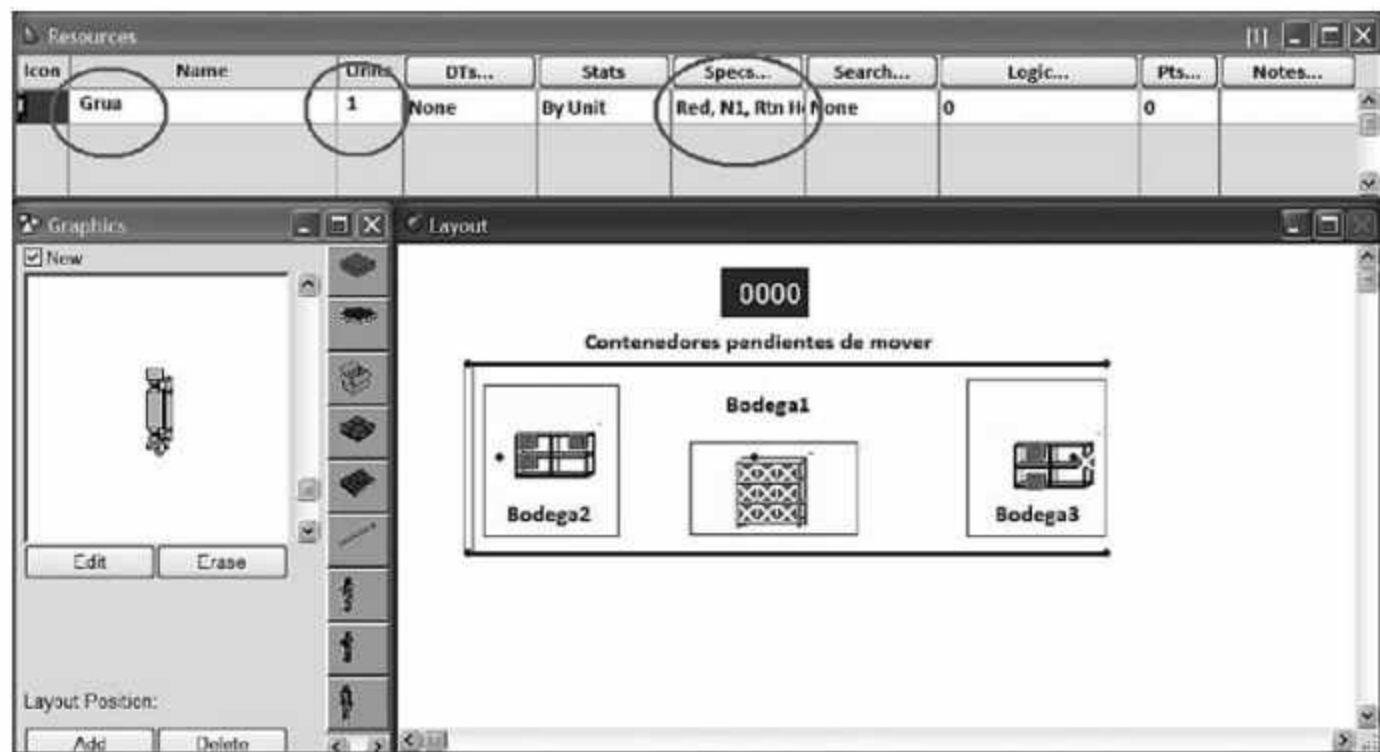


Figura 6.75 Definición del recurso Grúa.

- Haga clic en el botón **Specs** (identificado con uno de los círculos en la figura 6.75) para asociar el recurso con la ruta. Enseguida se abrirá el cuadro de diálogo

Specifications (vea la figura 6.76), en donde deberá seleccionar la ruta por la que se moverá el recurso *Grúa*. En este ejemplo, la *Red* tendrá asociado un nodo que será la base del recurso; también determinaremos que cuando dicho recurso esté ocioso deberá regresar a ese punto. Asimismo, vamos a especificar otras propiedades, como velocidad de movimiento, aceleración, desaceleración, etcétera.

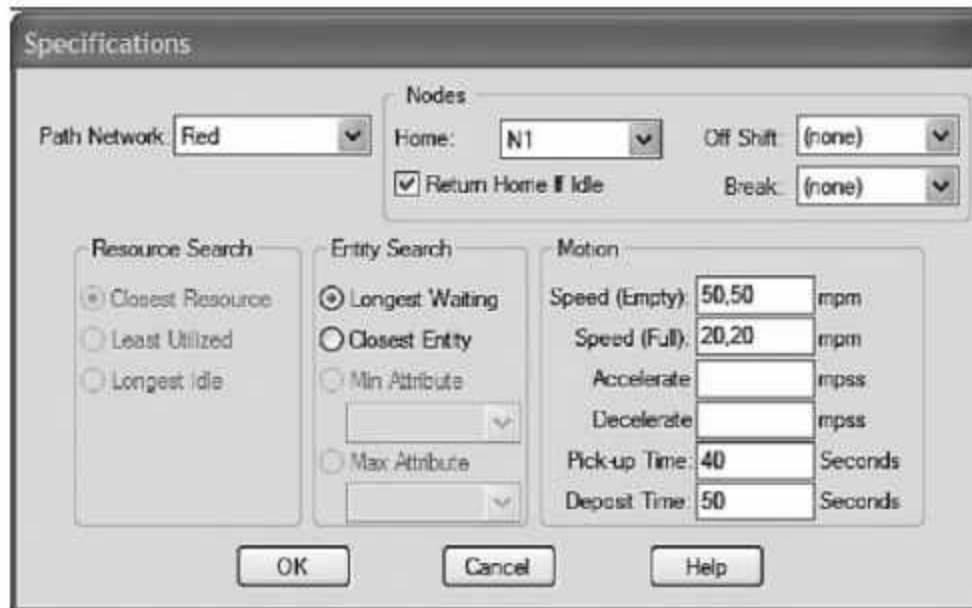


Figura 6.76
Especificaciones del recurso y su asociación con la ruta.

- Abra la ventana **Processing (Build / Processing)** para crear la ruta, considere los detalles de la figura 6.77, que consisten en la captura del recurso para transportar los contenedores desde la *Bodega1* hacia *Bodega2* o *Bodega3*, utilice la regla de ruteo probabilística, respete los porcentajes fijados inicialmente. Para hacer uso del recurso utilizamos la instrucción **MOVE WITH Grúa THEN FREE** ya utilizada en ejemplos anteriores. Note que en el ejemplo no usamos la instrucción **GET Grúa** en el campo **Operation**, lo anterior se debe a que esta instrucción está implícita en la instrucción **MOVE WITH** por lo que puede ser eliminada sin ningún problema, esto es válido también para el ejemplo anterior.

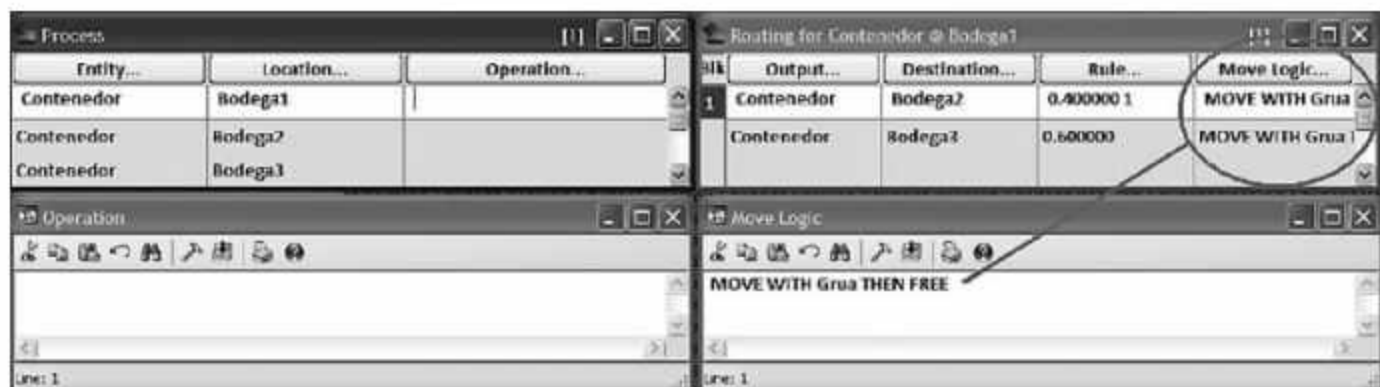


Figura 6.77 Segmento de la programación de la captura, uso y liberación del recurso dentro del proceso.

La sintaxis de programación del modelo para el ejemplo 6.11 se muestra en la figuras 6.78 y 6.79.

```

*****
*                                     *
*               Formatted Listing of Model:               *
*       C:\Program Files\ProModel\Models\Ejemplo 610.mod   *
*                                     *
*****
Time Units:                               Minutes
Distance Units:                           Meters
*****
*                                     *
*                               Locations                    *
*                                     *
*****
Name      Cap Units  Stats      Rules      Cost
-----
Bodega1   inf 1     Time Series Oldest, ;
Bodega2   inf 1     Time Series Oldest, ;
Bodega3   inf 1     Time Series Oldest, ;
*****
*                                     *
*                               Entities                    *
*                                     *
*****
Name      Speed (npm)  Stats      Cost
-----
Contenedor 150      Time Series
*****
*                                     *
*                               Path Networks              *
*                                     *
*****
Name      Type      T/S      From      To      BI      Dist/Time  Speed Factor
-----
Red      Crane Speed & Distance  Origin  Rail1End  Uni
         Origin  BridgeEnd  Uni
         BridgeEnd  Rail2End  Uni
*****
*                                     *
*                               Interfaces                *
*                                     *
*****
Net      Node      Location  Coords (R,B)
-----
Red      N1      Bodega2  5.00, 25.00
         N2      Bodega1  45.00, 25.00
         N3      Bodega3  95.00, 25.00
*****

```

Figura 6.78 Primera parte de la sintaxis del modelo para el ejemplo 6.11.

```

=====
*                               Resources                               *
=====
Name      Units  Stats      Res      Ent      Path      Motion      Cost
-----  -
Grua      1      By Unit  Closest  Oldest  Red      Empty: 50,50 mpm
Home: N1  Full: 20,20 mpm
(Return)  Pickup: 40 Seconds
Deposit: 50 Seconds
=====
*                               Processing                              *
=====
Process                                     Routing
Entity   Location Operation  Blk  Output  Destination Rule  Move Logic
-----  -
Contenedor Bodega1      1    Contenedor Bodega2  0.400000 1 MOVE WITH
                                           Contenedor Bodega3  0.600000 1 THEN FREE
                                           THEN FREE
Contenedor Bodega2      1    Contenedor EXIT    FIRST 1
Contenedor Bodega3      1    Contenedor EXIT    FIRST 1
=====
*                               Arrivals                               *
=====
Entity   Location Qty Each  First Time Occurrences Frequency  Logic
-----  -
Contenedor Bodega1  400    0      INF      1 day
=====

```

Figura 6.79 Segunda parte de la sintaxis del modelo para el ejemplo 6.11.

6.7 Instrucciones de control

En ProModel se puede hacer uso de algunas instrucciones de control con el propósito de tomar decisiones. Estas instrucciones se pueden clasificar en *decisiones*, cuando uno o más bloques de código de programación son ejecutados en función de cierta condición, o en *ciclos*, cuando una o más instrucciones son ejecutadas repetidamente hasta que cierta condición sea verdadera. A continuación se resumen las instrucciones de control disponibles en ProModel.

Instrucción IF-THEN-ELSE

La instrucción **IF-THEN-ELSE**, cuya sintaxis general es:

IF <Condición> THEN {acción 1} ELSE {acción 2}

y su propósito es que el programa ejecute la acción 1 si la Condición es verdadera y la acción 2 si es falsa.

Ejemplos:

```
IF Tipo=1 THEN {WAIT 5 min} ELSE {WAIT 10 min}
IF RAND(1)<=0.95 THEN {INC BUENAS} ELSE {INC MALAS}
IF Prod=1000 THEN {STOP}
IF CLOCK(H)<=8 THEN {ROUTE 1} ELSE {ROUTE 2}
IF CONTENTS(Fila)<=5 THEN {WAIT E(4) mi} ELSE {WAIT E(1) min}
```

Instrucción WHILE-DO

La instrucción **WHILE-DO**, tiene la siguiente sintaxis general:

WHILE <Condición> DO {acciones}

y causa que el programa ejecute las acciones mientras la Condición sea verdadera, si la Condición es falsa las acciones no se ejecutan.

Ejemplos:

```
WHILE CONTENTS(Fila)<=10 DO {WAIT 10 min}
WHILE CLOCK()<=480 DO {INC A,2}
```

Instrucción DO-WHILE

La instrucción **DO-WHILE**, cuya sintaxis general es

DO <acciones> WHILE {Condición}

causa que el programa ejecute las acciones mientras la Condición sea verdadera. Este ciclo será ejecutado al menos una vez y posiblemente en más ocasiones.

Ejemplos:

```
DO {WAIT 10 min} WHILE CONTENTS(Fila)<=10
DO {WAIT 10 min INC CICLO,1} WHILE WIP=10
```

Instrucción GO TO

La instrucción **GO TO**, con la sintaxis general:

GO TO <etiqueta>

envía la ejecución del programa al bloque de instrucciones que inician con la dirección marcada en etiqueta.

Ejemplo:

```
IF WIP>10 THEN GOTO LINEA4
WAIT 4 min
LINEA4: WAIT 2 min
```

Instrucción WAIT UNTIL

La sintaxis general de la instrucción **WAIT UNTIL** es:

WAIT UNTIL <Condición>

y detiene el flujo de una entidad en una localización hasta que la Condición sea verdadera.

Ejemplos:

```
WAIT UNTIL CONTENTS(ALM)<10
WAIT UNTIL CLOCK()>320 min
WAIT UNTIL PIEZAS=100
```

Ejemplo 6.12

El proceso de manufactura que se ilustra en la figura 6.80 consta de 1 rectificadora y un almacén donde las piezas esperan antes de ser procesadas. Los tiempos de proceso son de 10 y 8 min/pieza para la pieza tipo 1 y tipo 2, respectivamente. El tiempo entre arribos de la pieza tipo 1 a este proceso es de 12 min/pieza con distribución Exponencial, y el de la pieza tipo 2 es de 14 min/pieza con distribución Exponencial. Simule el proceso.

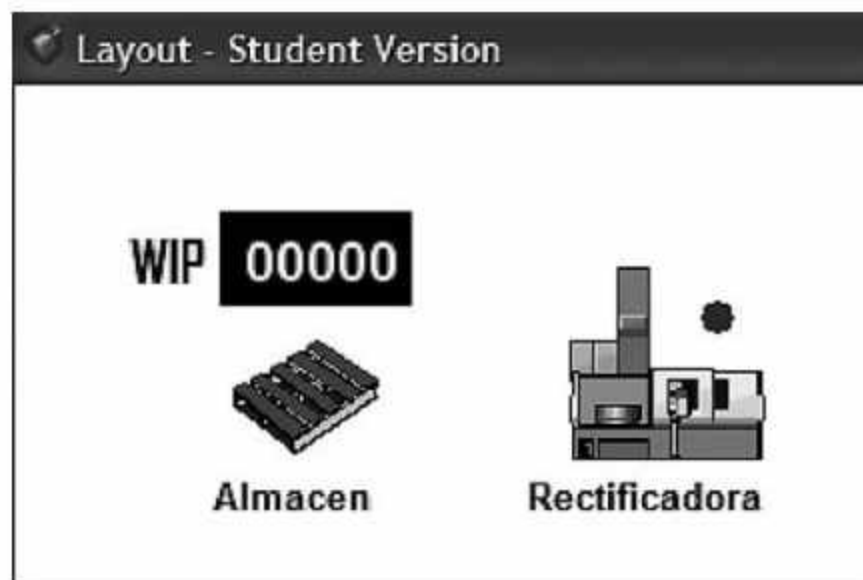


Figura 6.80
Esquematización del proceso de rectificado.

- Para empezar, definiremos las localizaciones *Almacen* y *Rectificadora* en la ventana **Locations (Build / Locations)**. Especifique la capacidad de dichas localizaciones como se muestra en la figura 6.81.



Locations			
Icon	Name	Cap.	Units
	Almacen	infinite	1
	Rectificadora	1	1

Figura 6.81 Definición de capacidad de las localizaciones.

- Defina la entidad *Pieza* en la ventana **Entities (Build / Entities)**, como se muestra en la figura 6.82.


Entities		
Icon	Name	Speed (fpm)
	Pieza	150

Figura 6.82 Definición de la entidad.

- Defina el atributo *Tipo* en la ventana **Attributes (Build / Attributes)**, como se muestra en la figura 6.83.

Attributes		
ID	Type	Classification
Tipo	Integer	Ent

Figura 6.83 Definición del atributo.

A continuación programaremos las llegadas de cada *Pieza* al *Almacén*. Abra la ventana **Arrivals (Build / Arrivals)** y especifique los parámetros que se muestran en la figura 6.84, defina en el campo **Logic** el valor del atributo *Tipo* correspondiente a cada tipo de pieza.

Entity...	Location...	Qty Each...	First Time...	Occurrences	Frequency	Logic..
Pieza	Almacen	1	0	INF	E (12) min	Tipo = 1
Pieza	Almacen	1	0	INF	E (14) min	Tipo = 2

Figura 6.84 Entrada de materia prima (*Pieza*) al punto inicial del proceso (*Almacén*).

- Despliegue la ventana **Processing (Build / Processing)** para crear la ruta de producción de las piezas a través de la rectificadora (vea la figura 6.85).

Entity...	Location...	Operation...
Pieza	Almacen	
Pieza	Rectificadora	IF Tipo = 1 THEN { WAIT 10 } ELSE {WAIT 8 }

Blk	Output...	Destination...	Rule...	Move Logic...
1	Pieza	EXIT	FIRST 1	IF ENTRIES(Rectificadora) > 800 THEN { STOP }

Figura 6.85 Programación del proceso en la rectificadora para cada tipo de pieza.

Como puede ver en la figura 6.85, en el campo **Operation** aparecen las instrucciones de control IF-THEN-ELSE con el propósito de asignar el tiempo de rectificado a las piezas en función del tipo de pieza que le fue asignado previamente en la ventana **Arrivals**, y en el campo **Move Logic** se utilizan las instrucciones de control con fin de terminar la simulación del sistema cuando el número de piezas que hayan entrado a la rectificadora sea mayor de 800 piezas.

La sintaxis de programación del ejemplo se muestra en la figura 6.86:

```

=====
Formatted Listing of Model:
=====
Time Units:          Minutes
Distance Units:     Feet
=====
Locations
=====
Name      Cap      Units Stats      Rules      Cost
-----
Almacen   infinite 1      Time Series Oldest: ;
Rectificadora 1      1      Time Series Oldest: ;
=====
Entities
=====
Name      Speed (fpm) Stats      Cost
-----
Pieza     150          Time Series
=====
Processing
=====
Process                                Routing
Entity Location      Operation      Blk Output Destination Rule Move Logic
-----
Pieza     Almacen          1              1 Pieza  Rectificadora FIRST 1
Pieza     Rectificadora IF Tipo = 1
              THEN { WAIT 10 }
              ELSE { WAIT 8 } 1 Pieza  EXIT      FIRST 1 IF
                                      ENTRIES(Rectificadora)>800
                                      THEN { STOP }
=====
Arrivals
=====
Entity Location Qty Each First Time Occurrences Frequency Logic
-----
Pieza     Almacen 1      0      INF      E( 12 ) min Tipo = 1
Pieza     Almacen 1      0      INF      E( 14 ) min Tipo = 2
=====
Attributes
=====
ID      Type      Classification
-----
Tipo    Integer   Entity
=====

```

Figura 6.86 Sintaxis completa de la programación para el ejemplo 6.12.

6.8 Optimización con Simrunner

Introducción

Una de las razones por las cuales se desarrollan modelos de simulación es por la necesidad de optimizar el sistema de estudio. La optimización es el proceso de evaluación de diferentes combinaciones de valores de un conjunto de factores que puedan ser controlables con el fin de encontrar aquella combinación que logre obtener el mejor de los resultados del sistema. En un proceso de optimización surgen preguntas tales como "¿Cuál es el número óptimo de máquinas que se deben instalar para maximizar la producción por hora?", "¿Cuántos operarios debemos de asignar a cierto departamento para minimizar el inventario promedio de piezas?". En estos ejemplos podemos distinguir a los

operarios y las máquinas como los factores o variables controlables y la producción por hora o el inventario promedio como la medida de rendimiento que nos permite seleccionar la mejor de las alternativas evaluadas. La búsqueda de la mejor solución puede ser llevada a cabo en forma manual o automática a través del uso de algoritmos específicamente diseñados para la búsqueda de una solución sin la necesidad de evaluar todas las posibles soluciones. ProModel incorpora la herramienta SimRunner con el fin de facilitar el proceso de optimización. SimRunner utiliza un algoritmo evolutivo que pertenece a un conjunto de técnicas de optimización conocidas como “Técnicas de Búsqueda Directa”, las cuales se han diseñado para encontrar los valores óptimos para las variables de decisión de un sistema con la finalidad de maximizar o minimizar las medidas de interés de dicho sistema.

La relación entre el ProModel y el SimRunner se puede visualizar como un sistema cerrado con retroalimentación en el cual el modelo de simulación se ejecuta en ProModel con alguna de las combinaciones posibles de las variables de decisión, de esta ejecución se obtiene el resultado de la variable de salida, que sirve como información de entrada al algoritmo de optimización que reside en el SimRunner y que permite generar una nueva combinación de valores de las variables de decisión para ejecutar el modelo nuevamente e ir en forma iterativa en busca de la mejor combinación de valores (vea la figura 6.87).

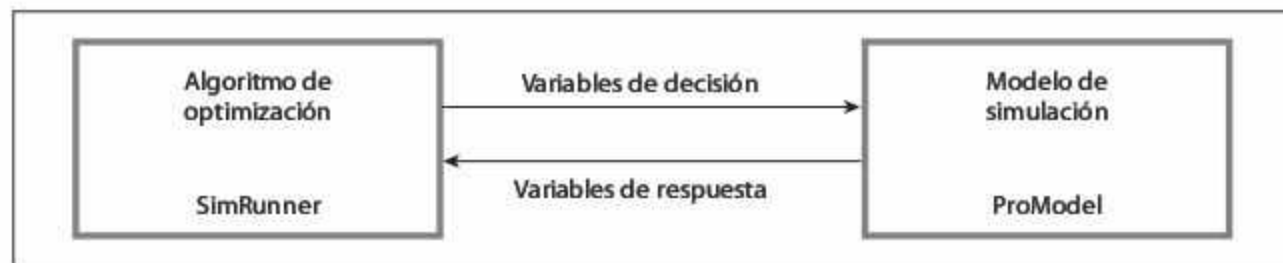


Figura 6.87 Relación entre el ProModel y el SimRunner.

SimRunner

Al conducir y analizar un experimento mediante SimRunner se debe de construir y ejecutar un proyecto. En cada proyecto SimRunner utiliza un algoritmo de búsqueda para encontrar simultáneamente valores óptimos para múltiples variables de decisión. Para cada proyecto se debe crear un modelo de simulación, identificar los factores a modificar y definir la función objetivo con la que se medirá el rendimiento del sistema. Para llevar a cabo este procedimiento se deben seguir los siguientes pasos:

Paso 1. Crear un modelo en ProModel estadísticamente válido y representativo del sistema a optimizar.

Paso 2. Seleccionar las variables de entrada o factores que se desean analizar para encontrar su valor óptimo. Estas variables son generalmente aquellos parámetros que pueden ser ajustados por el usuario, tales como el número de montacargas a asignar, el número de operarios a contratar, el número de tarimas a comprar o el precio de venta de un pro-

ducto. En este punto se deberá crear en ProModel una Macro para cada uno de estos factores.

Paso 3. Cada factor estará representado por una Macro y para cada una de ellas se define el tipo de dato (real o entero), su límite inferior (el valor más pequeño posible a analizar) y el límite superior (el valor más alto posible que se desee analizar). La cantidad de factores y el rango de valores que puede tomar cada uno afecta de manera directa el espacio de búsqueda o número de posibles soluciones del problema, por ejemplo, si se desean analizar tres factores cada uno con cuatro posibles valores, el espacio de búsqueda será de 81 combinaciones, pero si cada uno de los factores requiere de cinco valores, entonces el espacio de búsqueda será de 243 combinaciones. Incrementar el número de factores o su rango hará más difícil y más tardada la búsqueda del óptimo. Es importante seleccionar sólo aquellos factores que afecten de forma significativa las variables de respuesta.

Paso 4. Construir en SimRunner la función objetivo para medir la utilidad de cada combinación de factores. Esta función objetivo se construye por medio de los términos del reporte de salida generado al finalizar una corrida de simulación. Por ejemplo, la función objetivo puede estar basada en las estadísticas de las entidades (entities), las estadísticas de las ubicaciones (locations), las estadísticas de los recursos (resources), etcétera. Al diseñar la función objetivo se debe especificar si se desea minimizar (Min), maximizar (Max), o lograr un objetivo determinado (Target). Para crear la función objetivo se debe pensar en términos de Programación por Metas en donde podemos tener un objetivo dividido en varias metas, cada una con cierta ponderación. Por ejemplo, podemos tener en un sistema como primer objetivo o meta la de minimizar el inventario de piezas y un segundo objetivo que puede ser maximizar las producción por hora. Cada uno de ellos debe de estar acompañado por un factor de importancia relativa.

Paso 5. Seleccionar en SimRunner el perfil de optimización (optimization profile) y comenzar la búsqueda de la combinación óptima. Existen tres perfiles de optimización: agresivo, moderado y cauteloso. La selección de cada uno de ellos afecta el número de soluciones que el algoritmo de optimización va a evaluar. El perfil agresivo evalúa una menor cantidad de combinaciones que el cauteloso antes de detenerse. En forma general se puede decir que a mayor cantidad de soluciones evaluadas mayor probabilidad de encontrar la mejor, y mayor el tiempo requerido para llevar a cabo la búsqueda de la solución.

Ejemplo 6.13

A un estadio llegan aficionados con un tiempo entre arribos de 2.5 minutos con distribución Exponencial. El tiempo de viaje desde la entrada hasta las taquillas está normalmente distribuido con una media de cuatro minutos y una desviación estándar de 0.75 minutos. Frente a las taquillas, los aficionados esperan en una fila hasta que uno de los cajeros esté disponible para atenderlos. El tiempo para comprar los boletos es exponencial con una media de cinco minutos. Después de comprar los boletos los visitantes se dirigen hacia las puertas de acceso para entrar al estadio. Se desea crear un modelo de simulación para determinar el número mínimo de taquillas a abrir con el objetivo de que los aficionados esperen en promedio menos de 1 minuto.

Simule el proceso para atender 100 aficionados (vea la figura 6.88).

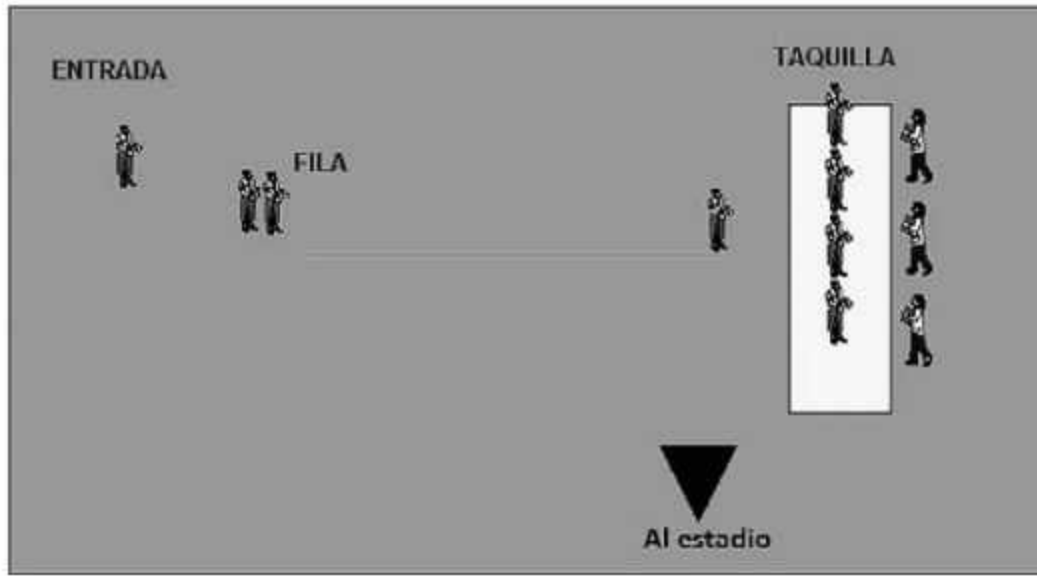


Figura 6.88
Proceso de venta de boletos.

- Como primer paso, se dará de alta una Macro que representará el factor que deseamos optimizar. Para este problema, el factor es el número de empleados a contratar para la venta de boletos. Iniciaremos con un valor de 3 empleados pero se hará una búsqueda del óptimo en el rango de 1 y 15 empleados. Definiremos la Macro *Empleados* en la ventana **Macros (Build / Macros)**. En el campo **Text** se dará el valor inicial de 3 y en el campo **Options** seleccionaremos **Scenario Parameter/Define** especificando en la ventana de diálogo el rango de valores de 1 a 15, como se muestra en la figura 6.89.

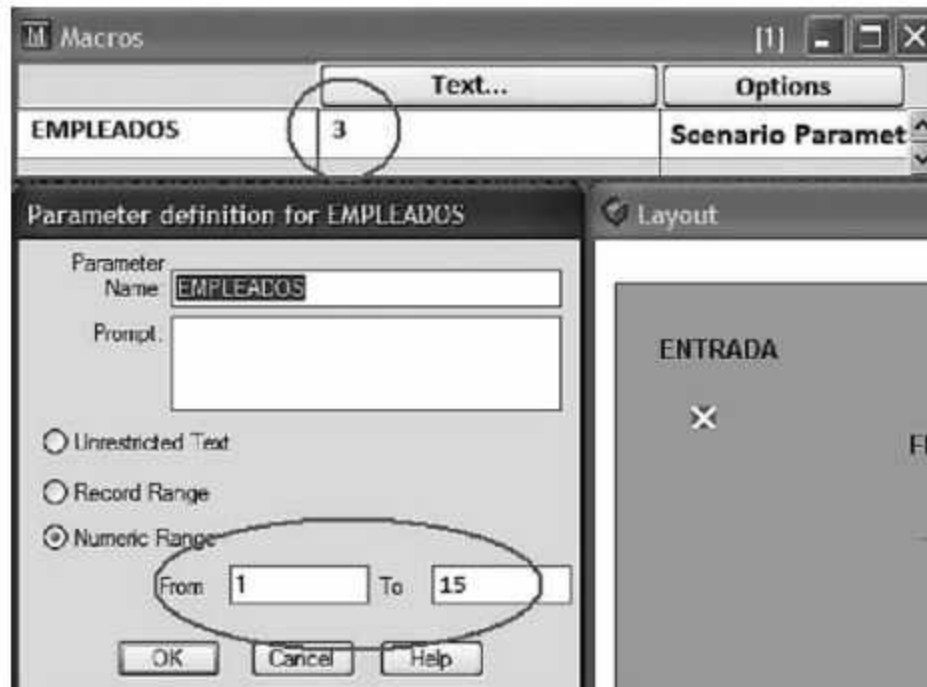


Figura 6.89
Definición de la Macro.

- Enseguida definiremos las localizaciones *Entrada*, *Fila* y *Taquilla* en la ventana **Locations (Build / Locations)**. Especifique la capacidad de dichas localizaciones como se muestra en la figura 6.90, observe que el valor numérico en la localización *Taquilla* se sustituye por la nombre de la Macro *Empleados*, de esta forma podremos ejecutar el modelo con los diferentes valores, en este caso el valor actual de la Macro es 3.

Icon	Name	Cap.	Units	DTs...
	ENTRADA	1	1	None
	FILA	INFINITE	1	None
	TAQUILLA	EMPLEADOS	1	None

Figura 6.90 Definición de capacidad de las localizaciones.

- Defina la entidad *Aficionado* en la ventana **Entities (Build / Entities)**, como se muestra en la figura 6.91.

Icon	Name	Speed (fpm)
	AFICIONADO	150

Figura 6.91 Definición de la entidad.

A continuación programaremos la llegada de cada *Aficionado* a la *Entrada*. Abra la ventana **Arrivals (Build / Arrivals)** y especifique los parámetros que se muestran en la figura 6.92.

Entity...	Location...	Qty Each...	First Time...	Occurrences	Frequency
AFICIONADO	ENTRADA	1	0	100	E(2.5) min

Figura 6.92 Entrada de entidades (*Aficionado*) al punto inicial del proceso (*Entrada*).

- Despliegue la ventana **Processing (Build / Processing)** para crear la ruta de los aficionados por las taquillas del estadio, (figura 6.93).

Entity...	Location...	Operation...	Routing for AFICIONADO @ ENTRADA				
Entity...	Location...	Operation...	Idk	Output...	Destination...	Rule...	Move Logic...
AFICIONADO	ENTRADA		1	AFICIONADO	FILA	FIRST 1	MOVE FOR N(4, 0.75) MIN
AFICIONADO	FILA						
AFICIONADO	TAQUILLA	WAIT E(5) MIN					

Figura 6.93 Programación del proceso incluyendo tiempos de traslado.

El número de aficionados que serán atendidos durante la ejecución del modelo no necesariamente asegura que los resultados lleguen a estado estable, por lo que para cada escenario se harán 30 réplicas. Para esto entramos al menú **Simulation/Options** y en la ventana de diálogo correspondiente damos de alta el valor de 30 en el campo **Number of replications**: una vez hecho esto se procede a guardar y ejecutar el modelo, con el fin de corregir cualquier error antes de iniciar con la optimización.

En el menú File/View Text podemos visualizar la sintaxis del ejemplo, el cual debe ser similar al de la figura 6.94:

```

Formatted Listing of Model:
C:\Program Files\ProModel\Models\Libro V2 Ejemplos\Ejemplo 71.mod

Time Units: Minutes
Distance Units: Feet

Locations

Name      Cap      Units Stats      Rules      Cost
-----
ENTRADA   1        1      Time Series Oldest,
FILA     INFINITE 1      Time Series Oldest, FIFO,
TAQUILLA EMPLEADOS 1      Time Series Oldest, First

Entities

Name      Speed (fpm) Stats      Cost
-----
AFICIONADO 150      Time Series

Processing

Process      Routing
Entity      Location Operation      Blk Output      Destination Rule      Move Logic
-----
AFICIONADO ENTRADA      1      AFICIONADO FILA      FIRST 1      MOVE FOR
N(4,0.75)MIN

AFICIONADO FILA
AFICIONADO TAQUILLA WAIT E(5) MIN      1      AFICIONADO TAQUILLA FIRST 1
1      AFICIONADO EXIT      FIRST 1

Arrivals

Entity      Location Qty Each      First Time Occurrences Frequency      Logic
-----
AFICIONADO ENTRADA 1      0      100      E(2.5) min

Macros

ID      Text
-----
EMPLEADOS      3
    
```

Figura 6.94 Sintaxis del ejemplo 6.13.

Para optimizar el modelo vamos a ejecutar el SimRunner desde el menú **Simulation / Simrunner** y debemos ir siguiendo las instrucciones que se presentan en la pantalla, en ocasiones hay que ser pacientes ya que este proceso puede tardar varios minutos. En caso de tener la versión estudiantil aparecerá una ventana de información en la cual se indica el número máximo de experimentos que podrán realizarse en la búsqueda del óptimo, en la versión incluida en este libro el número máximo es 25. Cuando aparece la pantalla mostrada en la figura 6.95 estaremos ya dentro del Simrunner. En esta primera ventana se dará de alta un nuevo proyecto, y por default se hará con el modelo que se encuentre activo. También es posible ejecutar proyectos realizados anteriormente.

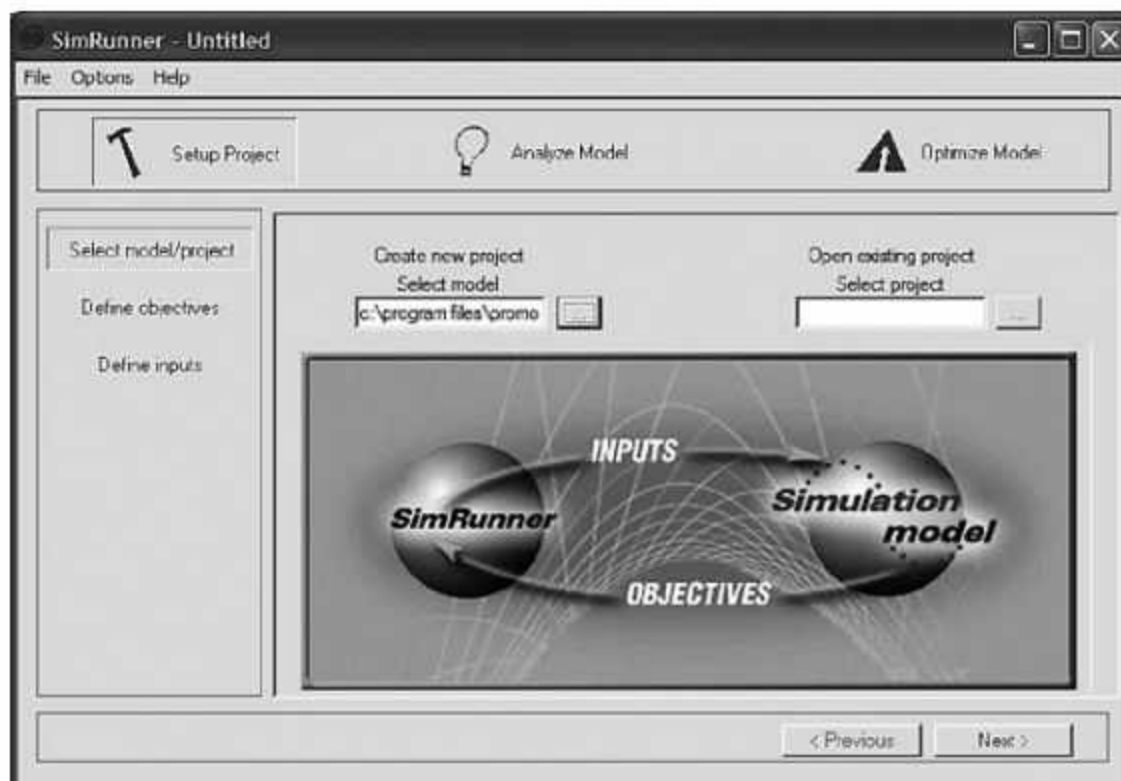


Figura 6.95 Ventana inicial del Simrunner.

Avanzamos a la pantalla siguiente con el fin de crear la función objetivo, para este ejemplo el objetivo es minimizar el tiempo promedio de espera de los aficionados en la fila, y esta variable puede ser localizada dentro del ProModel como AVG TIME/ENTRY asociada a la localización fila. En la ventana que se muestra en la Figura 6.96 seleccionamos Locations y FILA-Ave Time/Entry y mediante los botones circulados se arma y actualiza la función objetivo, la cual debe quedar activada en la ventana central, tenga cuidado de actualizar el tipo de objetivo a lograr (Min, Max, Target). Es posible dar de alta varias funciones objetivo y a cada una de ellas les puede asignar un peso relativo a las demás utilizando el campo *Weight*. En este caso sólo tenemos un objetivo, por lo que no se hará uso de este campo. La función objetivo, entonces, deberá leerse como: **Location Min 1*FILA - Avg Time/Entry**.

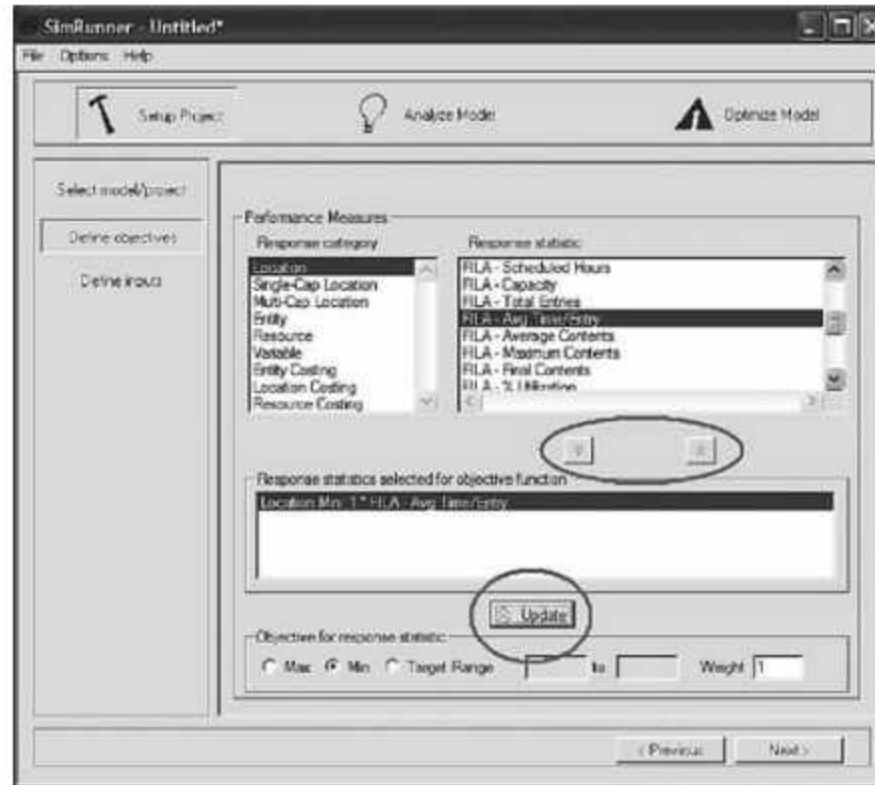


Figura 6.96
Ventana para crear la función objetivo.

Avanzamos en el proceso con el botón **Next** y ahora tendremos que seleccionar aquellos factores que el Simrunner podrá modificar para lograr la función objetivo. La figura 6.97 muestra esta pantalla en donde se puede actualizar información con respecto a los rangos y valores de los factores seleccionados sin necesidad de ir al modelo original a modificar los valores de las Macros, sin embargo, hay que tener en cuenta que modificar los valores de las Macros en este punto no modifica los valores originales del modelo, sólo se hace para este experimento. La figura 6.97 muestra la pantalla final después de haber manipulado la información, los factores activos del experimento deberán quedar activados en la pantalla central, para este ejemplo: **EMPLEADOS Default =3.00 Lower=1, Upper=15.**

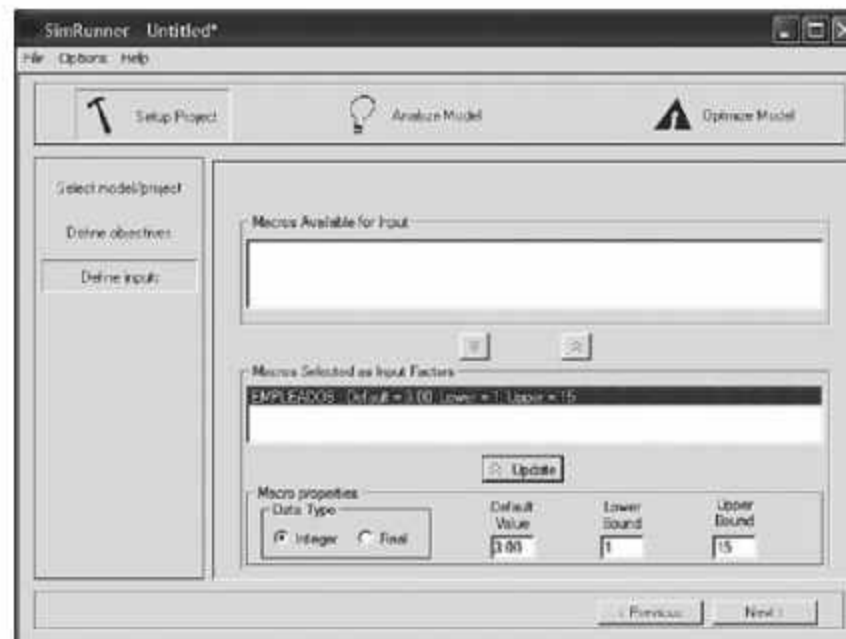


Figura 6.97
Ventana para seleccionar los factores.

Una vez hecho esto avanzamos a la siguiente pantalla en donde verificamos que las condiciones acerca de la longitud de la corrida, el número de réplicas y el nivel de confianza sea el adecuado. En este caso para poder seguir avanzando tendremos que asignar un tiempo de simulación, valor que no se dio de alta en el modelo original ya que se quería controlar por número de aficionados. Cada usuario puede dar de alta el valor que considere factible para que puedan ser atendidos 100 aficionados, en este caso se seleccionaron arbitrariamente 200 horas.

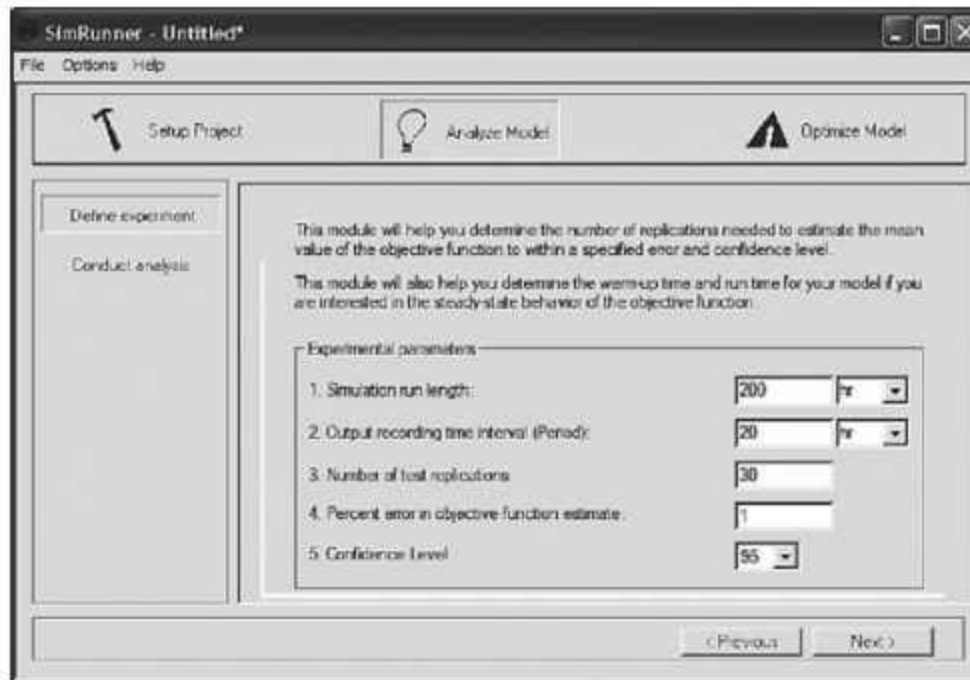


Figura 6.98
Ventana para seleccionar los parámetros de ejecución de la simulación.

A partir de la pantalla mostrada en la figura 6.98 saltaremos 2 pantallas, las cuales en este momento no son de utilidad para el análisis del problema pero se invita al lector a que experimente con los diferentes valores que se encuentran allí. En una primera instancia se recomienda seguir estos pasos para llegar hasta la pantalla que se muestra en la figura 6.99, en la cual al oprimir el botón **Run** activará el proceso de optimización.

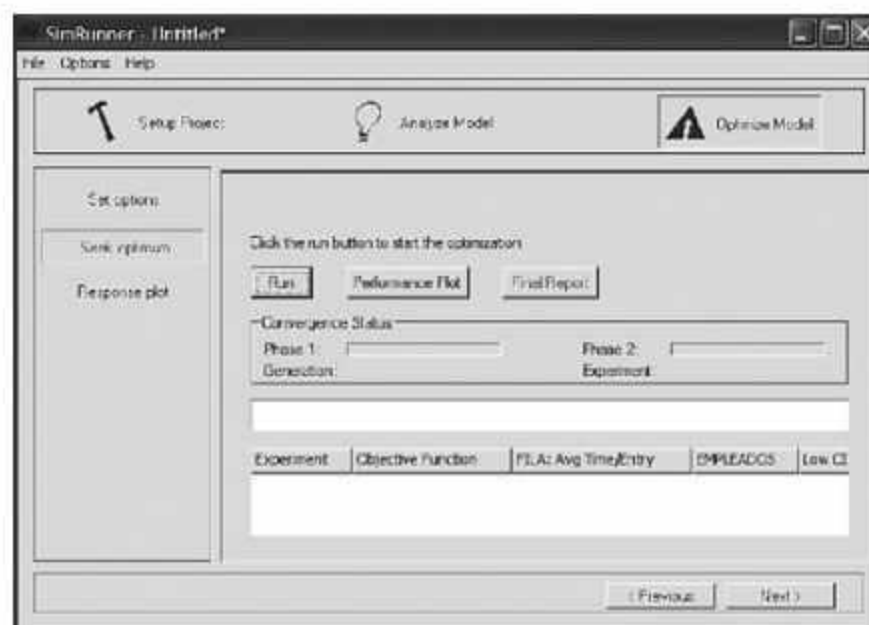


Figura 6.99
Ventana para la ejecución de la simulación.

En este proceso se ejecutarán los diferentes escenarios y al finalizar cada uno se irá actualizando la pantalla y se indicará el número del experimento, el valor de la función objetivo y sus límites de confianza superior e inferior, así como la combinación de valores de los factores analizados. El resultado de cada experimento se verá en pantalla en orden de mejor a peor.

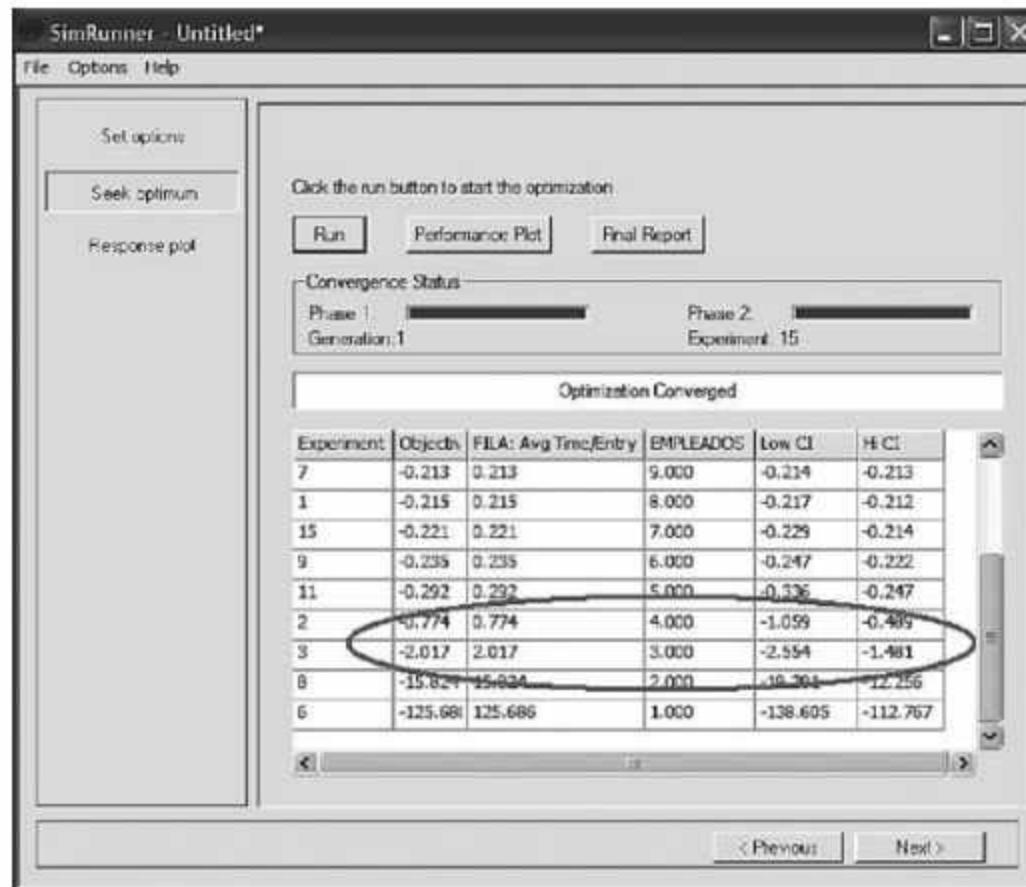


Figura 6.100
Ventana de resultados.

Para este ejemplo podemos observar el resultado final en la figura 6.100 en donde el objetivo de que los espectadores esperen menos de 1 minuto en promedio ocurre cuando el número de empleados es mayor o igual a 4. Si asignamos 4 empleados podemos esperar en la fila 0.774 minutos equivalente a 46.44 segundos, con límites entre 0.489 y 1.059 min con un intervalo de confianza del 95%.

Ejemplo 6.14

Consideremos ahora el mismo sistema descrito en el ejemplo 6.13 pero deseamos determinar el número de taquillas a instalar en función de un objetivo diferente. Si en este caso tenemos que el costo por hora de instalar cada taquilla es de \$300/h y el costo de espera por cada aficionado se estima en \$15/h, podemos plantear como nuevo objetivo: encontrar el número óptimo de taquillas que minimicen el costo del sistema.

El modelo en ProModel es idéntico al desarrollado anteriormente, el único cambio es en la definición de una nueva función objetivo la cual es:

$$\text{Min costo} = 300 * (\text{cantidad de taquillas}) + 15 * (\text{aficionados en espera})$$

Y la función objetivo se modifica en las ventanas de diálogo del Simrunner, por lo que ejecutamos nuestro modelo de Promodel siguiendo los pasos del ejemplo 6.13 hasta llegar a la ventana que se muestra en la figura 6.101 en donde comenzaremos a formar la nueva función objetivo:



Figura 6.101
Ventana para edición de la función objetivo.

En esta nueva función necesitamos dar de alta el número promedio de aficionados que permanecen en fila y su costo, por lo cual, en la ventana de la figura 6.101 seleccionamos **Locations** y **FILA-Average Contents** y mediante los botones centrales se arma y actualiza la función objetivo, que debe quedar activada en la ventana central, al actualizar el tipo de objetivo en minimizar (*Min*) y agregar en el campo *Weight* el valor de 15. Enseguida requerimos agregar el número de taquillas a instalar y su costo, para esto seleccionamos en la ventana superior la variable **Locations Taquilla-Capacity**, no olvide que la **Macro EMPLEADOS** está asociada en el modelo con la capacidad de la localización *Taquilla*, en este caso el objetivo también es minimizar (*Min*) y agregamos en el campo *Weight* el valor de 300 para finalmente actualizar con los botones centrales.

Al finalizar este proceso, en la ventana central mostrada en la figura 6.102 tendremos la función dividida en dos renglones:

Location Min 15*FILA – Average Contents
Location Min 300*TAQUILLA – Capacity

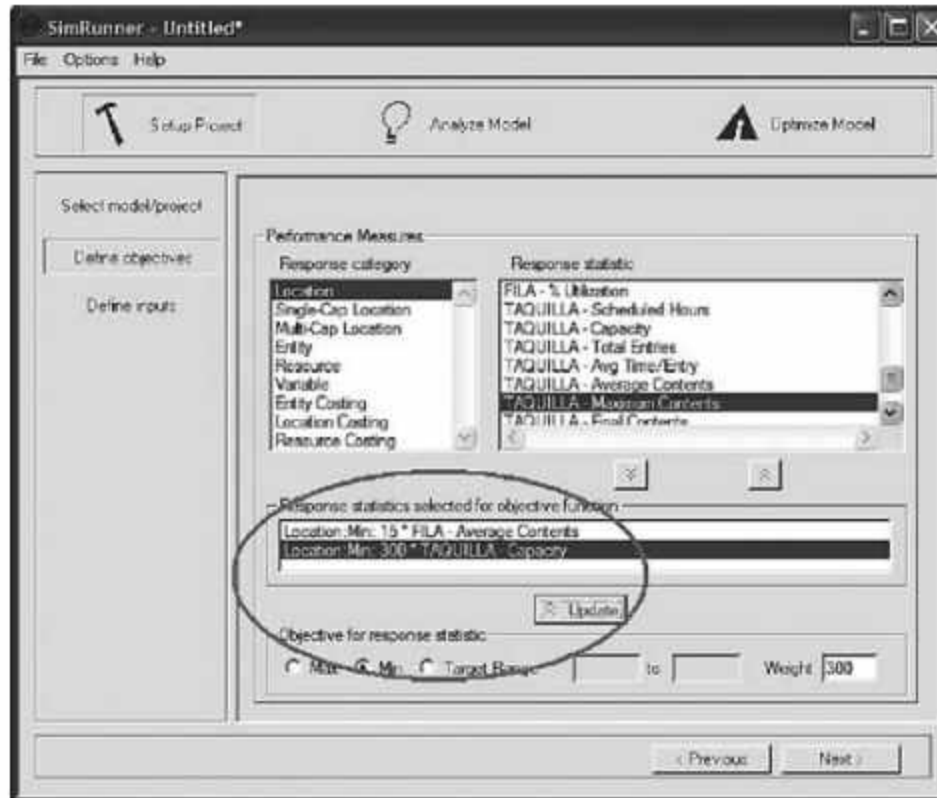


Figura 6.102
Función objetivo para minimizar costos.

Al ser éste el único cambio, continuamos con los pasos restantes de forma similar que en el ejemplo 6.13 y al final del proceso llegamos a la tabla de resultados mostrada en la figura 6.103 en donde podemos visualizar que el costo óptimo de la función objetivo es de \$667.5 /h y que el número óptimo de empleados a contratar es solamente de 1, lo que ocasiona una fila promedio de 24.5 aficionados. Podemos analizar este modelo si fijamos un costo diferente para la espera de los aficionados.

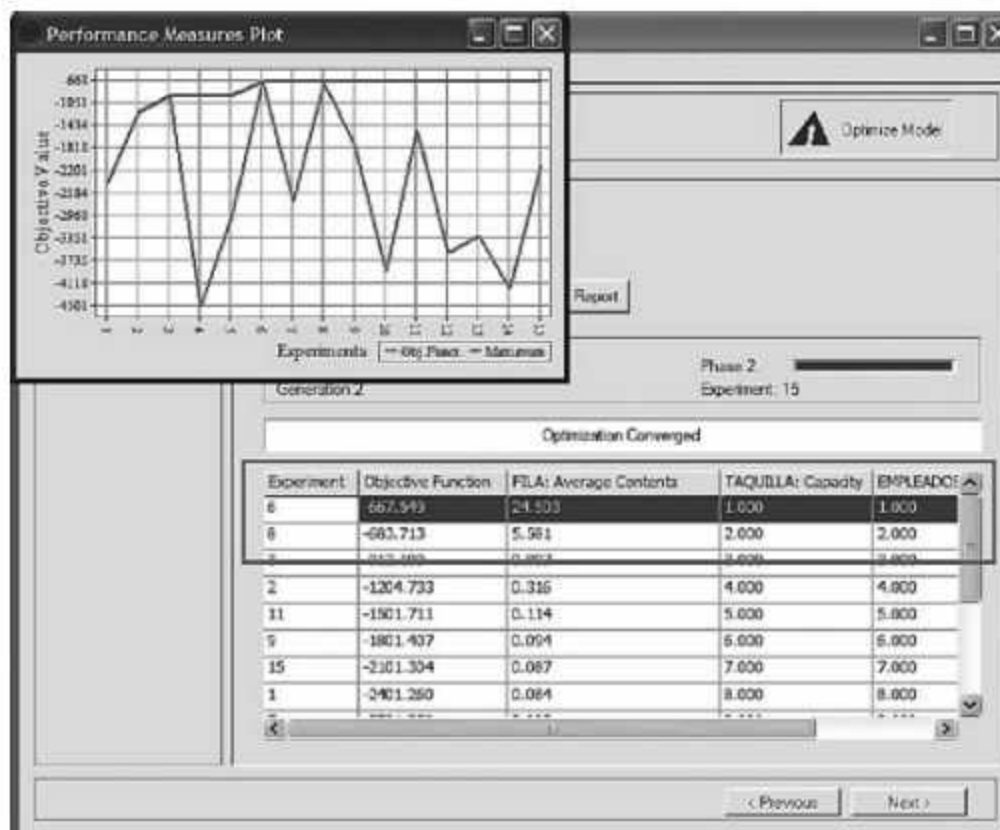


Figura 6.103
Solución al ejemplo 6.14.

Podría ser interesante hacer un análisis de sensibilidad de este modelo para determinar la influencia que tiene el costo de espera de los aficionados en el número de empleados a contratar. Por ejemplo si repetimos el experimento pero con \$30/h de espera, el resultado óptimo sería de 2 empleados con un costo por hora de \$767.4, y si fijamos el costo de espera en \$60/h, se mantiene el resultado en 2 empleados pero con un costo total de \$934.85/h. Si aumentamos en un último experimento el costo de espera a \$100/h, al ejecutar el Simrunner obtendremos el resultado mostrado en la figura 6.104, donde el costo óptimo total por hora es de \$980.7 con un número óptimo de empleados de 3.

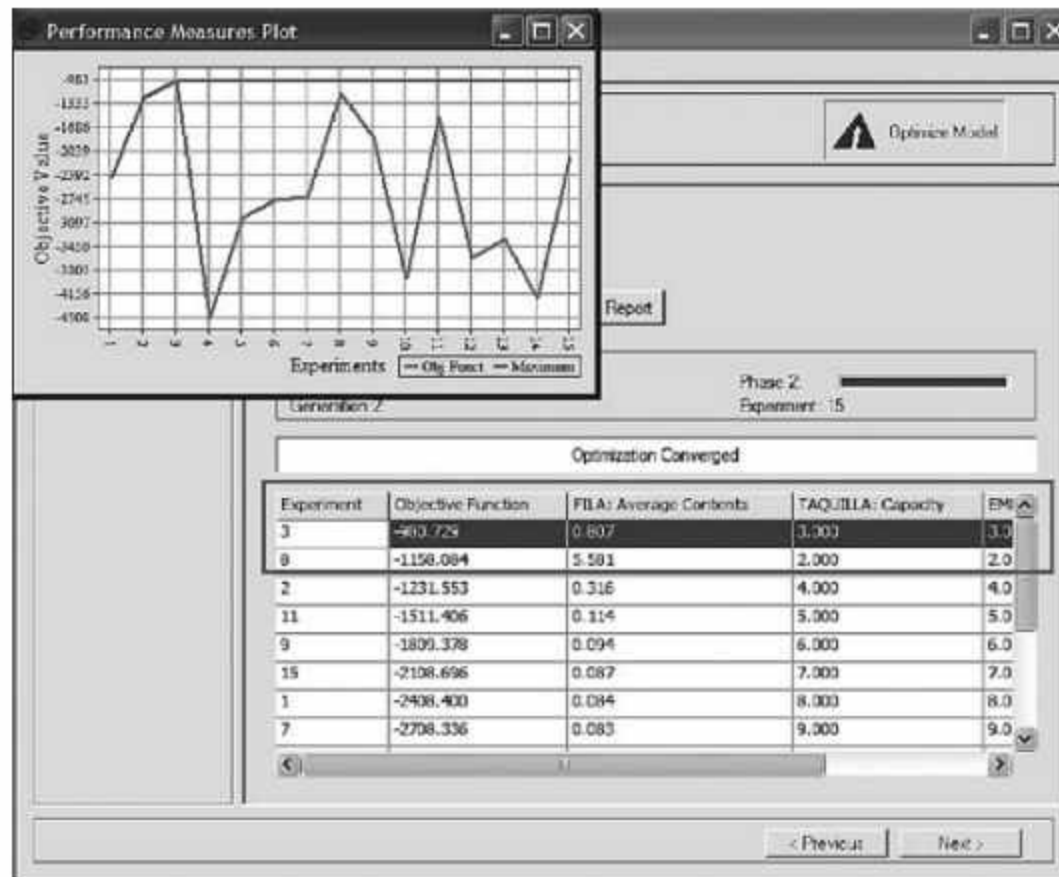


Figura 6.104
Análisis de sensibilidad del ejemplo 6.14.

6.9 Caso integrador 1

Una empresa fabricante de hilo poliéster tiene 50 máquinas estirotorcedoras en su departamento de estiraje. El índice de producción anual es de 90,000 toneladas, el cual no se está logrando debido a fallas en los motores de las estirotorcedoras. Cada máquina tiene un motor que está sujeto a descomposturas al azar, cuando un motor se descompone se reemplaza por otro de repuesto inmediatamente o cuando haya uno disponible. El motor descompuesto se envía al taller de mantenimiento y una vez reparado se convierte en un motor de repuesto.

En mantenimiento hay dos áreas, una de ellas es de capacidad para 100 motores y permite que los motores que no puedan ser reparados de inmediato esperen en ese lugar hasta que se desocupe alguno de los mecánicos y la segunda es donde los mecánicos hacen la reparación. La distancia que hay entre estiraje y el área de espera de motores antes de ser reparados es de 55 metros, entre el área de espera y el área de reparación hay

5 metros, entre reparación y el área de reserva hay 40 metros y la distancia entre el área de reserva y estiraje es 20 metros.

Los datos de la tabla 6.2 son una muestra histórica de 50 reparaciones.

Tabla 6.2 Tiempo de reparación (horas).

36.80	48.97	51.15	1.96	49.01
2.86	36.02	58.82	11.70	40.82
26.46	50.62	4.32	78.01	7.37
3.07	5.67	14.70	60.58	51.81
38.70	167.87	26.66	25.96	29.21
118.13	7.89	82.53	29.19	37.03
22.80	111.58	112.10	4.75	9.92
82.83	32.07	85.53	69.27	18.73
198.88	25.53	27.46	16.73	39.86
27.28	41.26	65.79	10.33	59.38

En la tabla 6.3 se tienen los datos históricos del tiempo que transcurre desde que un motor es instalado en la estirotorcedora hasta que sufre una descompostura.

Tabla 6.3 Tiempo de operación de los motores (horas).

437.65	80.59	355.57	141.20	182.59
98.34	263.45	294.31	97.82	200.28
378.73	130.09	311.03	87.93	195.97
264.56	78.37	80.63	48.07	318.78
143.89	324.54	343.22	118.70	191.50
65.06	134.71	183.02	208.03	119.89
330.00	140.07	111.64	147.79	157.34
203.86	121.53	70.51	109.04	112.96
61.89	102.03	439.97	183.66	290.41
187.15	416.68	256.00	38.32	79.60

Debido al volumen y peso de los motores, cualquier movimiento de un motor debe hacerse con un montacargas. La velocidad del montacargas es de 150 metros/min si va

vacío y 100 metros/min si va cargado. El tiempo para subir y o bajar un motor del montacargas es de 45 segundos.

Actualmente la empresa tiene 3 motores de repuesto, 5 mecánicos y 2 montacargas.

Se estima que cada tonelada se vende a \$2,000. El costo anual de cada mecánico es de \$150,000, el de los montacargas es de \$450,000, y cada motor ocasiona gastos anuales por \$750,000.

Construya un modelo en ProModel y mediante Simrunner determine:

- a) El índice real de producción actual.
- b) La utilidad anual actual.
- c) El mínimo número de mecánicos a contratar para trabajar en el taller, el número mínimo de montacargas a comprar para el traslado de los motores y el número mínimo de motores de repuesto, para que el departamento de estiraje esté trabajando al 90% de utilización y que le permita obtener un índice de producción anual de 81,000 toneladas. La producción por fabricada por arriba de esta cantidad no se puede vender por falta de demanda.
- d) La utilidad anual con las mejoras realizadas.

6.10 Caso integrador 2

La empresa EHLE, S.A. se dedica a la fabricación y ensamble de circuitos electrónicos (PCB) para la industria automotriz, y desea instalar una nueva celda flexible de manufactura donde se puedan ensamblar 10 nuevos tipos de circuitos electrónicos. Las operaciones para fabricar y ensamblar son las siguientes:

1. Carga de la tarjeta en el contenedor (CTC)
2. Limpieza (LI)
3. Corte con cizalla (CCZ)
4. Taladrado de orificios (TO)
5. Copia de patrones (CP)
6. Ataque químico (AQ)
7. Deposición electrolítica (DE)
8. Inspección (IN)
9. Inserción de componentes (IC)
10. Soldadura, limpieza y pruebas (SLP)
11. Descarga del circuito del contenedor (DCC)

La línea de ensamble puede trabajar de 1 a 3 turnos en función de la demanda de los clientes. Los productos se mueven en contenedores a través de la celda, y cada proceso se lleva a cabo sobre el contenedor. Existe sólo una banda transportadora por donde se mueven los contenedores, de manera que es imposible que uno rebase a otro. Hoy en día la empresa cuenta con 350 contenedores y desea saber cuántos asignar a la celda para el transporte y manejo de materiales. El diagrama 1 muestra el flujo de la celda.

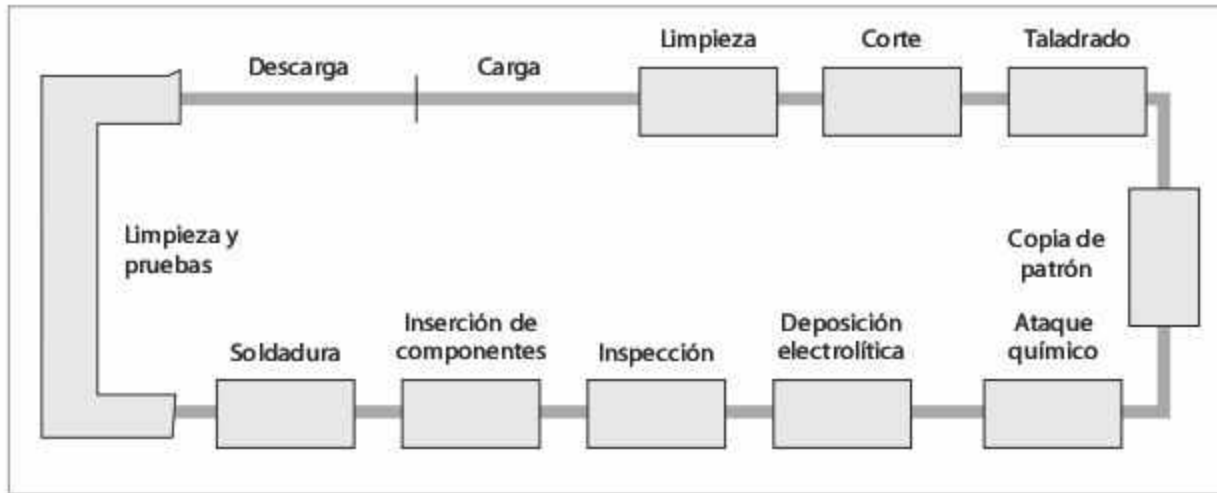


Diagrama 1. Diagrama de flujo de la línea de fabricación y ensamblaje.

En cada una de las estaciones únicamente se puede procesar un contenedor a la vez, excepto en Lavado y Limpieza; además, existen lugares de espera limitados de acuerdo con los siguientes parámetros:

Estación	Cantidad máxima de contenedores permitidos en la estación	
	En espera	En proceso
Carga de la tarjeta en el contenedor	4	1
Limpieza	0	5
Corte con cizalla	6	1
Taladrado de orificios	0	1
Copia de patrones	0	1
Ataque químico	0	1
Deposición electrolítica	9	1
Inspección	0	1
Inserción de componentes	0	1
Soldadura	4	1
Limpieza y pruebas	0	10
Descarga del circuito del contenedor	9	1

En el diagrama 2 se presenta, en forma esquemática, el proceso de producción, incluyendo los tiempos de proceso y los tiempos de transporte en cada etapa (todos los

tiempos están en minutos y por contenedor). Casi todos estos tiempos son constantes, excepto las estaciones marcadas con la expresión $f(t)$, ya que el tiempo de proceso en esas operaciones depende de cada tipo de producto y familia. En algunos casos, como en el de Taladrado, existen 2 componentes de tiempo, uno constante y otro que depende del producto.

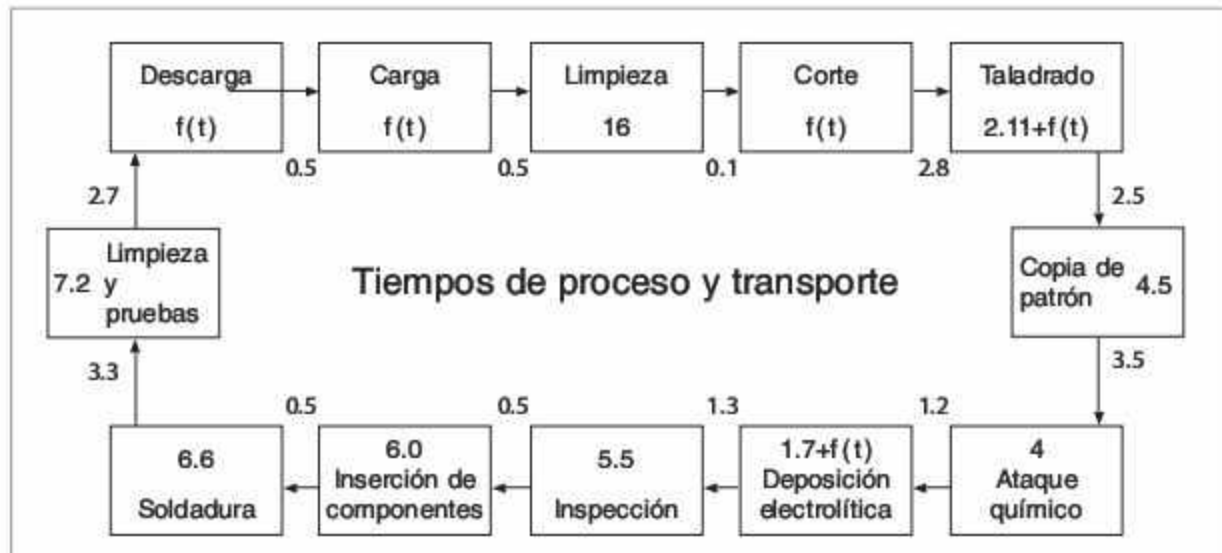


Diagrama 2. Tiempos de proceso y transporte (min/contenedor).

La celda puede procesar 10 tipos de productos diferentes, aunque en un programa de producción de un día típico de trabajo se pueden fabricar menos tipos de circuitos. Esto se debe a la distribución de probabilidad de la demanda diaria de cada producto. La columna 2 de la tabla 6.4 muestra esas distribuciones; por ejemplo, del producto 1 se pueden programar entre 8 y 10 circuitos con distribución uniforme; del producto 2 se pueden programar en promedio 8 circuitos con distribución de Poisson, y así sucesivamente. El departamento de Administración de la Producción envía este programa a Producción un día antes, de tal manera que el material lo deja disponible desde el inicio del turno, para comenzar a procesarlo durante el día.

Los circuitos se han agrupado por familias, de acuerdo con su tiempo de proceso en cada etapa. Por ejemplo, según la tabla 6.4, el circuito tipo 1 pertenece a la familia C en el proceso de Carga, a la familia A en el proceso de Corte, a la familia A en el proceso de Taladrado, a la familia B en el proceso de Deposición, y a la familia C en el proceso de Descarga.

Los contenedores están vacíos al inicio de cada turno, empiezan a cargarse con los diferentes productos y, al finalizar la producción, deben de quedar vacíos nuevamente.

Tabla 6.4 Programa de producción y definición de familias.

Tipo de circuito	Cantidad a producir por día U = Uniforme P = Poisson	Operación				
		CTC	CCZ	TO	DE	DCC
		Familia a la que pertenece el circuito				
1	U(8, 10)	C	A	A	B	C
2	P(8)	A	C	A	A	A
3	P(3)	A	C	A	B	B
4	P(2)	B	C	A	A	B
5	P(5)	B	B	A	A	C
6	U(10, 20)	C	B	B	A	B
7	P(4)	B	D	B	A	A
8	U(0, 2)	A	A	C	A	A
9	P(8)	A	A	C	A	B
10	U(20, 30)	B	A	C	A	C

En todos los casos, la componente aleatoria del tiempo de proceso, $f(t)$, se expresa con una función de densidad Erlang con un valor esperado (t) y un parámetro de forma k , como se muestra en la tabla 6.5 para cada una de las familias y procesos. Por ejemplo, todos los productos que pertenezcan a la familia A en el proceso de Carga tendrán un valor esperado de 7.55 minutos, con $k = 3$ y así sucesivamente.

Tabla 6.5 Familias y distribución de probabilidad de los tiempos de proceso: k -Erlang(t) en min/contenedor.

Operación				
CTC	CCZ	TO	DE	DCC
A 3-Er(7.55)	A 2-Er(6.37)	A 4-Er(3.91)	A 3-Er(3.12)	A 1-Er(9.45)
B 3-Er(7.70)	B 2-Er(12.1)	B 4-Er(6.56)	B 3-Er(4.44)	B 4-Er(9.36)
C 2-Er(8.28)	C 2-Er(6.96)	C 2-Er(3.64)		C 2-Er(8.12)
D 3-Er(8.82)	D 1-Er(2.27)	D 3-Er(6.15)		D 4-Er(8.52)
	E 2-Er(8.02)			

Con el propósito de modelar correctamente la celda y diseñarla tomando en cuenta todos los factores, se ha decidido incluir el comportamiento de las fallas y el tiempo de reparación de las estaciones de Limpieza e Inserción de componentes. Las tablas 6.6, 6.7, 6.8 y 6.9 muestran los comportamientos históricos de máquinas similares a las que se desea instalar. Las fallas en el resto de los equipos son despreciables.

Tabla 6.6 (MTBF) Tiempo entre fallas en Limpieza, en min/falla.

144.58	162.38	156.07	158.88	156.20	159.20	164.52	163.47	162.24	156.55
147.81	160.84	150.95	161.59	162.62	154.24	155.31	159.86	165.35	160.61
148.81	163.69	156.03	154.46	159.01	159.17	158.80	149.89	152.04	159.39
154.93	151.48	151.57	154.49	156.82	163.19	156.67	158.04	161.51	155.36
149.38	161.79	160.45	160.52	154.32	159.79	158.38	151.09	159.86	152.91
156.73	156.07	159.80	147.14	157.08	159.91	161.32	161.08	153.76	155.70
160.50	161.15	157.82	163.80	160.10	160.32	150.71	161.35	154.19	146.04
154.19	156.13	166.73	150.44	163.32	156.10	167.30	150.09	153.58	157.50
153.25	161.83	158.62	159.87	165.06	162.57	164.22	157.25	155.42	160.31
158.61	163.26	154.16	160.51	158.68	162.83	162.44	161.66	153.06	156.41
154.97	162.96	165.15	155.87	155.02	160.53	161.51	164.42	162.64	157.49

Tabla 6.7 (MTBF) Tiempo entre fallas en Inserción, en min/falla.

390.38	340.79	353.86	375.09	387.54	376.53	368.91	364.23	380.81	382.50
352.23	353.55	366.98	387.21	381.95	366.98	372.79	373.68	368.01	371.33
334.49	346.45	344.52	384.61	333.35	351.04	354.60	350.01	310.28	387.16
366.33	379.57	376.32	373.21	388.54	356.29	325.38	320.72	346.07	324.70
380.62	379.65	360.09	330.05	387.11	410.69	352.58	383.61	365.38	358.24
334.37	386.09	347.36	388.19	347.10	310.49	379.52	334.38	353.17	356.86
332.71	358.20	356.32	336.09	384.46	349.54	347.12	363.04	357.66	362.61
351.00	406.17	326.97	311.41	345.02	378.39	390.65	333.74	350.43	332.19
373.34	370.81	374.81	387.64	361.44	360.25	384.13	386.91	372.00	343.56
355.54	354.69	368.34	340.94	319.74	377.43	387.61	379.29	322.39	383.20

Tabla 6.8 (MTTR) Tiempo de reparación en Limpieza, en min/falla.

6.28	24.89	51.35	10.58	4.91	29.36	3.13	4.02	0.69	34.10
4.44	23.85	19.33	36.39	13.50	2.72	8.35	33.61	23.50	3.67
30.66	16.96	4.01	29.91	42.99	3.22	8.07	12.02	24.44	36.41
54.85	31.30	23.95	17.92	3.91	10.32	5.25	7.15	24.54	14.78
22.95	7.28	1.30	2.08	10.60	21.08	113.92	24.26	36.74	8.38
12.06	46.07	20.42	7.53	2.41	0.89	11.72	5.32	8.70	60.55
9.83	15.74	5.23	2.18	19.01	10.24	1.99	0.06	12.75	4.59
4.51	13.01	9.06	9.68	18.81	3.09	0.17	6.85	15.59	8.24
28.26	7.28	6.31	10.39	43.10	9.33	4.70	26.90	6.98	65.31
17.86	26.64	35.21	0.01	62.72	7.94	18.47	19.49	20.52	31.61

Tabla 6.9 (MTTR) Tiempo de reparación en Inserción, en min/falla.

16.73	16.50	17.67	15.16	17.84	19.15	17.23	16.39	16.35	14.68
17.70	16.12	14.94	16.89	17.97	15.91	18.01	17.39	17.72	15.97
18.37	18.34	18.28	16.29	15.40	16.83	16.92	18.18	16.08	15.34
15.29	15.19	17.19	16.36	16.89	14.87	17.25	17.27	14.97	18.05
16.14	17.09	17.97	16.55	17.55	17.69	15.86	16.36	17.48	16.87
15.39	17.82	16.40	17.99	17.26	16.00	16.68	17.32	16.28	16.49
18.08	16.98	16.51	16.84	17.40	15.08	14.34	17.30	16.94	16.77
17.73	17.55	16.71	16.49	16.21	15.81	16.24	18.20	15.94	16.70
17.20	15.24	15.99	16.85	18.57	17.89	15.58	16.11	17.24	15.22
16.66	16.83	16.25	16.00	15.10	14.49	17.79	15.09	16.94	16.84

Dentro de la empresa se han generado varias preguntas, ya que se vislumbra —en un futuro cercano— un incremento considerable en la demanda de productos. Para contestar tales interrogantes desarrolle en ProModel un modelo de simulación estadística-mente válido, que permita determinar:

- a) La operación en que se presenta un cuello de botella.
- b) La hora de terminación del programa de producción diario, considere la variación en la demanda de un día para otro, con un nivel de 90% de confianza.
- c) La cantidad óptima de contenedores que debe asignarse a la celda. Es posible justificarla mediante una gráfica donde se relacione el número de contenedores asignados contra la producción diaria lograda, observe a partir de qué punto la producción prácticamente no se incrementa.
- d) El inventario promedio de contenedores en cada operación.
- e) El tiempo de permanencia promedio de los contenedores en cada operación.
- f) El tiempo promedio de permanencia de un contenedor en la celda de manufactura.

6.11 Problemas

1. Un inspector recibe siempre 120 piezas/h. El tiempo promedio de inspección es uniforme, entre 20 y 30 s/pieza. Simule el sistema en ProModel, y calcule la utilización del inspector y el número máximo de piezas acumuladas antes del proceso de inspección.
2. Un cajero automático recibe 30 clientes/h con distribución de Poisson. Existen 5 escenarios posibles:

Escenario	Tiempo promedio de servicio (min/cliente)
1	Exponencial con media 1.8
2	2-Erlang con media 1.8
3	4-Erlang con media 1.8
4	Normal con media 1.8 y varianza 0.2
5	1.8

Simule el sistema en ProModel y calcule, para cada escenario:

- a) La utilización del cajero.
 - b) El número promedio de clientes en espera.
 - c) Si en todos los casos el tiempo promedio de servicio es el mismo, ¿a qué factor se deben las diferencias (si es que existen)?
 - d) ¿Qué diferencia existe entre el primero y el último escenario?
3. A una ferretería llegan 56 clientes/h con distribución de Poisson. El negocio cuenta con sólo dos dependientes, y los clientes deben hacer una fila. El tiempo de servicio es exponencial con media de 2 min/cliente. Simule el sistema hasta lograr el estado estable y calcule:
 - a) La utilización de los dependientes.
 - b) El tiempo promedio de espera en la fila.
 4. Una tienda de aparatos electrónicos vende 2 tipos de microcomputadoras: la E-GD y la H-GR. El tiempo entre llegadas es exponencial con media de 45 min/cliente. Se trata de una tienda pequeña, por lo que solamente requiere un empleado para atender a los clientes de acuerdo con un esquema "primero en llegar, primero en salir". Veinticinco por ciento de los clientes que entran no realizan compra alguna, y utilizan al empleado durante 15 minutos exactamente. Cincuenta por ciento de los clientes que entran compran una computadora tipo E-GD, y el tiempo que les lleva realizar la transacción sigue una distribución uniforme de entre 31 y 36 minutos. El 25% restante entra a la tienda y compra la computadora tipo H-GR; el tiempo que se requiere para la venta en este caso sigue una distribución exponencial con media de 70 minutos. Simule 8 horas y determine:
 - a) Utilización del empleado.
 - b) El tiempo promedio que un cliente tiene que esperar antes de ser atendido.

5. Un sistema de producción cuenta con 10 tornos. El tiempo de operación de cada uno sigue una distribución de probabilidad exponencial con media de 96 horas, después del cual ocurre una falla y tiene que ser enviado a mantenimiento. El tiempo de reparación es exponencial con media de 72 horas. El tiempo de traslado de los tornos entre producción y mantenimiento es uniforme, con parámetros de 60 ± 15 minutos. Se desea simular el sistema hasta alcanzar el estado estable, con 1, 2, 3, 5, ... , 10 mecánicos para determinar en cada caso la utilización de los mecánicos y el número promedio de tornos en espera de reparación.
6. La empresa LECAR dispone de 100 telares. El tiempo de operación de cada uno antes de que ocurra una rotura sigue una distribución de probabilidad exponencial, y el telar se detiene en espera de que un obrero repare la rotura y reinicie la operación. El tiempo de reparación es de 2-Erlang con media de 5 minutos. El tiempo promedio entre roturas depende de la calidad del hilo, de acuerdo con los siguientes escenarios:

Escenario	Calidad del hilo	Tiempo promedio entre roturas (min/rotura)
Optimista	Primera	40
Promedio	Segunda	20
Pesimista	Dudosa	5

Cree un modelo de simulación para cada escenario, y determine el número de obreros que deben asignarse a la reparación de roturas con el objetivo de mantener en operación los telares 95% del tiempo.

7. A una biblioteca llega un promedio de 104 personas/año con distribución de Poisson, para pedir prestado cierto libro. La persona que logra encontrarlo lo regresa en promedio 10 días después, con distribución exponencial. Las personas que solicitan el libro pero no lo reciben por estar en préstamo, se van y nunca regresan. Simule el proceso durante un año, con 1, 2, 3 y 4 ejemplares del libro, y determine en cada caso el número esperado de personas que podrán leer el libro.
8. A la operación de empaqueo de bolsas de detergente entran bolsas a una velocidad de 20 por minuto. Cuando las bolsas entran al sistema son colocadas en una banda que las transporta hasta la mesa de un operario de empaque. El tiempo de transporte en la banda es de 20 s/bolsa. Una vez que la bolsa llega al final de la banda, cae por gravedad hacia una mesa donde se va acumulando con otras. Un operario toma las bolsas de la mesa y las introduce en una caja con capacidad de 30 unidades; el tiempo que le lleva al operario tomar una bolsa y colocarla dentro de la caja es de 1 segundo/bolsa. Una vez que la caja se llena, el operario la lleva al almacén de cajas; allí la deja y recoge una caja vacía para repetir el procedimiento de llenado. El tiempo que le toma en llevar la caja llena y traer una vacía sigue una distribución exponencial con media de 3 minutos.

Simule el sistema anterior en ProModel, para obtener la gráfica del número de bolsas en la mesa, el número promedio de bolsas y el tiempo promedio de espera en la mesa a lo largo del tiempo.

9. A un proceso de troquelado llegan láminas de acuerdo con una distribución de Poisson con promedio de 82 láminas/min. Se cuenta con 7 troqueladoras, cada una de las cuales es capaz de procesar una lámina en 5 segundos. Hay un almacén de grandes dimensiones para materia prima, de manera que todas las piezas que no puedan ser troqueladas inmediatamente podrán esperar ahí. El costo de operación de las máquinas se estima en \$10/h-troqueladora, y el costo de mantener una lámina en inventario se estima en \$0.500/h-lámina. Simule el sistema para determinar el inventario promedio de láminas y el costo total/h.
10. A un proceso de troquelado llegan láminas de acuerdo con una distribución de Poisson con promedio de 82 láminas/min. Se cuenta con 7 troqueladoras, cada una de las cuales es capaz de procesar una lámina en 5 segundos. Hay un almacén para materia prima con capacidad de 5 láminas; si una lámina llega y no puede entrar al proceso o al almacén de materia prima, debe de ser enviada a otro lugar de la planta para la realización del troquelado. El costo de operación de las máquinas se estima en \$10/h-troqueladora; el costo de mantener una lámina en inventario se estima en \$0.500/h-lámina, y el costo de enviar a las láminas hacia otro lugar es de \$0.800 por lámina. Simule el sistema para determinar el inventario promedio de láminas en el almacén antes de troquelado, y el costo total/h.
11. A un centro de maquinado llegan tres diferentes tipos de piezas. Antes de llegar pasan por un almacén de producto en proceso, con capacidad prácticamente infinita. El tiempo de operación y la tasa de entrada de las piezas son las siguientes:

Tipo de pieza	Tasa de entrada Poisson (piezas/h)	Tiempo de maquinado exponencial (min/pieza)
1	2	3
2	4	5
3	2	10

Simule el sistema en ProModel durante 100 horas, y determine:

- Uso del centro de maquinado.
 - Número total de piezas producidas.
 - Tiempo promedio de espera de las piezas en el almacén.
 - Número promedio de piezas en el almacén.
12. A un operario de limpieza le entregan cada hora 60 piezas en forma simultánea. El tiempo de limpieza es uniforme, de 50 ± 10 s/pieza. Simule el proceso anterior durante 500 horas para determinar:
- Uso del operario.
 - Tiempo promedio de permanencia de las piezas en todo el proceso.
 - Tiempo promedio de espera de las piezas antes de ser limpiadas.

13. Un sistema de pintura tiene dos procesos en serie: pintura y horneado. El tiempo de pintura es exponencial, de 10 min/pieza, y el de horneado es triangular (3,6,15) min/pieza. Para ambos procesos hay dos pintores y un horno. La tasa de entrada es de 7 piezas/h en el caso de la pieza tipo 1, y de 3 piezas/h en el caso de la pieza tipo 2. El tiempo para moverse de un proceso a otro es de 30 segundos. Simule el sistema 5 días para determinar:
- Uso de cada operación.
 - Tiempo promedio de permanencia de las piezas en todo el proceso.
 - Tiempo promedio de espera de las piezas antes de pintura y antes del horneado.
14. A un centro de copiado llegan tres tipos de trabajos. Si un trabajo no puede ser iniciado de inmediato, espera en una fila común hasta que esté disponible alguna de las tres copiadoras con las que cuenta el centro. El tiempo de copiado y la tasa de entrada de los trabajos son como siguen:

Tipo de trabajo	Tasa de entrada (trabajos/h)	Tiempo de copiado (min/trabajo)
1	4	Exponencial(12)
2	Poisson(8)	Normal(15,2)
3	16	3-Erlang(1)

Después del proceso de copiado los trabajos son inspeccionados por un empleado en un tiempo exponencial con media de 3, 6, 10 minutos para los trabajos 1, 2 y 3, respectivamente. Simule el sistema en ProModel durante 50 horas, y determine:

- Uso del empleado y las copiadoras en la situación propuesta.
 - Número de empleados y copiadoras mínimos necesarios para asegurar el flujo de los trabajos.
15. Un tipo de pieza entra a una línea de producción. El proveedor entrega en forma exponencial con media de 2 min/pieza. La línea consta de 3 operaciones con una máquina en cada operación. Los tiempos de proceso son:

Operación	1	2	3
Tiempo (min/pieza)	Constante (2)	Exponencial (1)	Uniforme (0.5 ± 0.1)

El tiempo para moverse entre estaciones es de 0.0625 minutos. La animación debe incluir un contador de las piezas producidas. Simule en ProModel el proceso de 500 piezas para determinar:

- Tiempo total de simulación.
- Uso de cada operación.
- Tiempo de espera antes de la primera operación.
- Porcentaje del tiempo que la pieza estuvo bloqueada.

16. A un proceso de perforado llegan piezas a partir de las 8:00 a.m. El tiempo entre llegadas es exponencial, con media de 30 s/pieza. La perforación se puede realizar en cualquiera de las 3 máquinas existentes; sin embargo, debido al tamaño del transformador de energía eléctrica, las máquinas no pueden trabajar al mismo tiempo. Se ha decidido que la producción se realice en la máquina 1 las primeras 3 horas del turno, en la máquina 2 las 3 horas siguientes, y en la máquina 3 las 2 últimas horas. Por razones de programación, la máquina 1 inicia siempre a las 8:15 a.m. Los tiempos de perforado en cada máquina se muestran a continuación:

Máquina	Tiempo promedio de proceso (s/pieza)
1	Exponencial (20)
2	Exponencial (10)
3	Uniforme (15 ± 5)

Realice un modelo en ProModel para simular 30 días de 8 horas cada uno, y obtenga:

- Producción al final de cada uno de los días.
 - Distribución de probabilidad de la producción diaria.
17. A una oficina llegan 2 tipos de clientes. La tasa de llegadas de los clientes tipo I sigue una distribución uniforme (100-150) min/cliente; la tasa del segundo tipo sigue una distribución constante con media de 120 min/cliente. Solamente existe un servidor que tiene que atender a ambos tipos de clientes de acuerdo con un esquema de "primero en llegar, primero en salir". El tiempo que tarda en atender a los clientes tipo I sigue una distribución exponencial con media de 25 min/cliente, mientras que el tiempo de servicio del segundo tipo de clientes sigue una distribución 2-Erlang con media de 35 min/cliente. Simule hasta que hayan sido atendidos 500 clientes de tipo II y determine:
- Tiempo total de simulación.
 - Número de clientes tipo I que fueron atendidos.
 - Tiempo promedio de estancia en la oficina por cada tipo de cliente.
 - Número máximo de clientes en la oficina.
18. En un banco hay 2 cajas, cada una con su propia fila de tamaño infinito: la caja 1 para clientes "rápidos", con tiempo de 2 minutos, y la caja 2 para clientes "lentos" con tiempo de 16.4 minutos, en ambos casos con distribución exponencial. Los clientes "lentos" y "rápidos" llegan de acuerdo con una función exponencial, con media de 8 y 20 min/cliente, respectivamente. Algunos días, en particular, solamente viene a trabajar 1 cajera que tiene que atender ambas filas según la siguiente secuencia cíclica: 5 clientes "rápidos" y 2 clientes "lentos". El tiempo para moverse entre las cajas es de 0.33 minutos. Simule en ProModel el proceso por 80 horas, y determine:
- Número promedio de clientes en espera en cada fila.
 - Secuencia óptima de atención para minimizar el tiempo promedio de espera, considere ambos tipos de clientes.

19. A un sistema de limpieza entran intercambiadores de calor en forma exponencial, con media de 5 h/pieza. El proceso consta de 3 operaciones con una máquina y en cada operación. Los tiempos de proceso son:

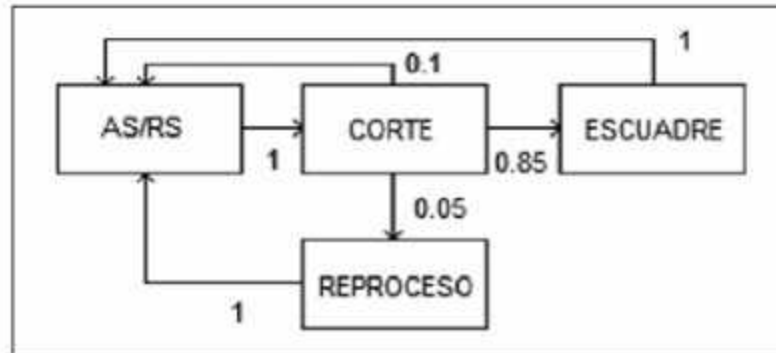
Operación	Tiempo promedio de proceso (h/pieza)
1	Constante (4.9)
2	Exponencial (4)
3	Uniforme (4.7 ± 0.3)

Se tiene un montacargas para todos los movimientos entre estaciones. El tiempo de traslado entre estaciones es 3-Erlang con media de 30 min/intercambiador. La animación debe incluir un contador de las piezas producidas.

- I. Simule en ProModel el proceso de 500 intercambiadores para determinar:
 - a) Tiempo total de simulación.
 - b) Utilización de cada operación.
 - c) Tiempo de espera antes de la primera operación.
 - d) Porcentaje del tiempo que la pieza estuvo bloqueada.
 - e) ¿Recomendaría tener otro montacargas? Justifique su respuesta.

 - II. Simule el modelo con un escenario donde el montacargas falle en forma exponencial cada 15 horas de trabajo y sea reparado con una distribución normal con media de 30 minutos y desviación estándar de 5 minutos. En este caso, ¿sería apropiado tener dos o más montacargas? Justifique su respuesta.
20. Un promedio de 10 personas/h con distribución de Poisson intentan entrar al jacuzzi de un hotel. En promedio, cada persona permanece 20 minutos con distribución exponencial. En el jacuzzi existen solamente 6 espacios, por lo que si una persona llega y están ocupados todos los lugares, se retira enojada y no regresa. Simule el proceso durante 100 horas.
- a) ¿Cuál es la utilización del jacuzzi?
 - b) En promedio, ¿cuántas personas se encuentran en el jacuzzi?
 - c) En promedio, ¿cuántas personas por hora *no* pueden usar el jacuzzi?
 - d) ¿Cuántos lugares debería tener el jacuzzi para asegurar que 95% de las personas que llegan puedan entrar?
21. Determine mediante ProModel el número de máquinas que requiere un sistema para procesar las piezas, tenga como restricción mantener una utilización entre 60 y 80%. Las piezas llegan de acuerdo con una distribución de Poisson, con media de 1.5 clientes por minuto. Cuando una pieza entra y todas las máquinas están ocupadas, permanece en una sola fila común a las máquinas. El tiempo de proceso sigue una distribución exponencial con media de 3 minutos. Incluir dentro de los resultados los valores de la utilización, tiempo de espera promedio y número de piezas promedio en espera.

22. El diagrama muestra el proceso y ruta de las piezas dentro de una celda de manufactura. Todas las piezas deben entrar y salir de la celda a través de la máquina de carga y descarga (AS/RS). Los números en las flechas dentro del diagrama representan el porcentaje de flujo que es enviado a otros equipos; por ejemplo, 10% de lo que entra a corte se envía al AS/RS, 85% a escuadre, y 5% a reproceso. Un sistema de control automático mantiene siempre 10 piezas dentro del sistema (observe el siguiente diagrama).

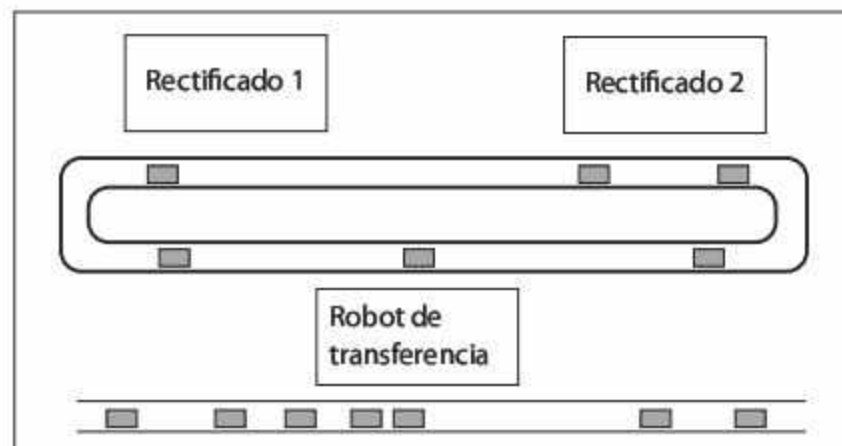


El tiempo de proceso en cada operación es:

Operación	[Min/pieza]
AS/RS	1.000
CORTE	0.465
ESCUADRE	0.500
REPROCESO	6.666

Simule en ProModel y determine para cada operación:

- Utilización.
 - Número máximo de piezas.
 - Número promedio de piezas.
23. En el sistema cerrado que se muestra en el diagrama se mantienen siempre 20 piezas en proceso. Cada una entra desde una banda principal a través de un robot de transferencia, y debe visitar las 2 estaciones de rectificado antes de regresar al robot de transferencia para ser enviada de nuevo hacia la línea principal. Las piezas se mueven a través de bandas transportadoras (observe el siguiente diagrama).



Los tiempos de producción son:

Operación	Tiempo de proceso [min/pieza]
Robot	Normal (4,1)
Rectificado 1	Normal (6,0.3)
Rectificado 2	Exponencial (4)
Transporte entre estaciones	1

Simule durante 200 horas y determine:

- Producción por hora.
- Uso de las estaciones.
- Inventario promedio en proceso en cada estación.
- Tiempo promedio de permanencia de las piezas en el sistema cerrado.

24. Se utilizan cinco camiones para transportar concreto de un lugar a otro. Solamente se tiene una tolva de carga, y el tiempo para hacerlo sigue una distribución exponencial con media de 20 minutos. El tiempo para transportar el concreto y regresar por más material sigue una distribución exponencial con media de 180 minutos. Simule el movimiento de los camiones para contestar:

- En promedio, ¿cuánto espera un camión en la fila de la tolva?
- ¿Qué fracción del tiempo no se utiliza la tolva?
- Si se han programado 500 viajes, ¿cuánto tiempo tomará hacerlos?

25. A un sistema de producción de la empresa EHYPESA llegan piezas de tipo 1 cada 5 ± 3 minutos, y piezas tipo 2 cada 3 ± 2 minutos. Las piezas tipo 1 pasan por limpieza en un tiempo de 8 ± 3 minutos; al salir, 25% deben limpiarse de nuevo, y el 75% restante sale del sistema para su venta. Las piezas tipo 2 pasan primero por verificación en un tiempo de 9 ± 3 minutos, y después por limpieza en un tiempo de 3 ± 1 minutos. Al salir de limpieza, 5% deben limpiarse de nuevo, y el 95% restante sale del sistema para su venta. Simule el sistema un mes y determine el número mínimo de operarios de verificación y limpieza que permitan maximizar la producción por hora. Indique número de piezas de cada tipo que se produjeron durante el mes.

26. En el hospital HELEED, los pacientes entran a las salas de emergencia a una tasa Poisson con media de 4 persona/h. Ochenta por ciento de los pacientes entran directamente a una de las 5 salas; el 20% restante son llevados con una secretaria para un proceso de registro de datos. El tiempo en registro es de 5 ± 0.5 min/paciente, luego, pasan a las salas de emergencia. Se tiene de guardia a dos médicos en las salas de emergencia, y ellos tardan en revisar a los pacientes un tiempo 3-Erlang con media de 30 min/paciente. Después de este tiempo el paciente es trasladado a otras áreas del hospital, de acuerdo con su estado.

Realice un modelo en ProModel para simular 3 réplicas de 48 horas de la situación anterior y determine, con un intervalo de confianza, de 95%:

- a) Tiempo promedio de espera de los pacientes en el proceso de registro.
 b) Número máximo de clientes que estuvieron en todo el hospital.
 c) Utilización de los médicos, las salas de emergencia y la secretaria.
27. Cierta pieza requiere 2 operaciones: soldadura en 6 minutos, y esmerilado en 6.4 minutos. La llegada de piezas es uniforme entre 10 y 15 min/pieza. Existen almacenes de tamaño infinito antes de cada operación. Se tiene solamente un operario para hacer ambas operaciones, de acuerdo con la siguiente secuencia cíclica: 5 piezas en soldadura y 5 piezas en esmerilado. El tiempo para moverse entre las máquinas es de 1 minuto. Simule en ProModel el proceso 480 horas, y determine el tiempo promedio de espera de las piezas en los almacenes antes de cada operación.
28. Una máquina empacadora es alimentada por dos bandas transportadoras. Por la primera entran dulces a una razón constante de 2000 dulces/h. Por la segunda entran bolsas. La máquina empaca 50 dulces por bolsa y coloca el producto empacado en una tercera banda, para su transportación hacia el almacén de producto terminado. El tiempo total de empaque es de 20 segundos. La velocidad de las bandas es de 150 pies por minuto. Cada 3 horas la empacadora se detiene para ajuste y limpieza; el tiempo para llevar a cabo estas operaciones es exponencial con media de 10 minutos. Mientras la empacadora es ajustada, las bolsas y los dulces siguen entrando al sistema. Desarrolle un modelo en ProModel y determine la tasa promedio de entrada de las bolsas.
29. A la oficina de la SRE llegan clientes para recibir su pasaporte a una tasa de Poisson con media de 18 clientes/h. Al entrar toman una ficha que permite atenderlos en el orden de llegada. Noventa por ciento de los clientes entra directamente a la sala de espera, y el resto llena primero algunas formas en un tiempo uniforme (6 ± 2) minutos. Existen 2 servidores para atender a los clientes de la sala de espera; el tiempo promedio de atención es exponencial con media de 6 minutos.
- La sala de espera dispone de 40 sillas. Si un cliente llega y todas las sillas están ocupadas, permanece de pie y se sienta cuando se desocupa alguna de ellas.
- Cada vez que se expiden 10 pasaportes, el servidor deja de atender la fila y conduce a los clientes atendidos a la parte posterior, después de lo cual sigue atendiendo la fila; el tiempo en ir y venir sigue una función exponencial con una media de 5 minutos.
- Haga un modelo en ProModel para simular durante 8 horas la situación actual, de manera que, al finalizar la simulación, se indique en pantalla los siguientes resultados:
- a) Tiempo promedio de espera en la fila.
 b) Número promedio de personas sentadas.
 c) Número promedio de personas de pie.
 d) Número máximo de personas en la sala de espera.
 e) Utilización de los servidores.

30. En la compañía PESYPINSA, las piezas entran a un proceso de pintura a una tasa de Poisson con media de 1.0 min/pieza. Antes de pintarse, las piezas deben pesarse en cualquiera de las dos básculas existentes: el tiempo en la báscula es uniforme, de (2 ± 0.5) minutos. Existe una persona que realiza la operación de pintura en 12 minutos por pieza, con distribución exponencial.

Hay también un pintor de reserva, que entra a proceso cada vez que el número de piezas en el almacén de producto en proceso entra a las básculas y la pintura excede de 10, y permanece trabajando durante una hora, después de lo cual deja de pintar y regresa a su posición de pintor de reserva.

Haga un modelo en ProModel para simular durante 8 horas la situación actual, de manera que, al finalizar la simulación, se indique en pantalla:

- a) Tiempo promedio de espera en el almacén de pintura.
 - b) Número máximo de piezas que se llegaron a acumular en todo el sistema.
 - c) Utilización de la báscula y de los pintores.
31. La empresa MOLGADE Co. fabrica moldes y desea simular en ProModel una parte de su proceso. Los moldes inicialmente se encuentran en el almacén de materia prima, en cantidad suficiente para no detener la producción por falta de material. La primera operación es una limpieza con chorro de arena, con duración de 4 minutos. Enseguida, uno de seis inspectores verifica la dimensión de los moldes; cada inspector tarda 3-Erlang con media de 18 minutos en la verificación. Luego, los moldes son transportados a un horno en lotes de 5, mediante una grúa viajera. El tiempo de transporte es de 6 minutos, y la empresa cuenta con dos grúas para este movimiento. En el horno se cargan 4 lotes para iniciar el proceso de calentamiento; este tiempo es de una hora. Al salir del horno, los moldes se enfrían al aire libre en 3 ± 1 hora. Una vez terminado el proceso, se empacan en cajas de 25 moldes. El tiempo de empaque es triangular de (2,3,6) min/caja, y los moldes se transportan en contenedores de 8 cajas cada uno hasta el área de producto terminado en un AGV. El tiempo requerido para este movimiento es de 10 minutos.
- Se trabajan tres turnos de ocho horas cada uno. Modele la operación durante un mes para:
- a) Determinar la utilización de los equipos, el inventario promedio en proceso y el tiempo de permanencia de un molde en el sistema.
 - b) Detectar problemas y proponer el aumento o disminución de recursos.
 - c) Determinar, para la solución propuesta, la utilización de los equipos, el inventario promedio en proceso, y el tiempo de permanencia de un molde en el sistema.
32. Use SimRunner para determinar el número de máquinas que debe haber en un centro de copiado, su objetivo es que los clientes no esperen más de 20 minutos en la fila. Los clientes llegan de acuerdo a una distribución Poisson con una media de 4 clientes/min. El tiempo que lleva atender a cada cliente sigue una distribución exponencial con media de 7.5 minutos. Cuando un cliente llega y todas las copadoras están ocupadas tiene que esperar en una fila común.

33. A una tienda llegan 10 clientes/h con distribución Poisson. El tiempo de atención de los clientes sigue una función 3-Erlang con media de 15 min/cliente. Cada servidor puede atender sólo a un cliente simultáneamente y si todos los servidores están ocupados el cliente espera en una fila única. El costo de servicio es \$500/h-servidor y el costo de la espera se estima en \$1000/h-transacción. Se desea determinar el número de servidores a contratar utilizando SimRunner.
34. A una celda de manufactura llegan 10 engranes al día con distribución Poisson. El tiempo de proceso está distribuido de manera uniforme entre 1 y 2 horas. La celda opera 16 horas por día. Simule este sistema 30 días y encuentre la utilización promedio de la celda, el tiempo promedio de residencia de los engranes y el número de engranes promedio en el proceso. La celda sólo puede procesar una orden a la vez.
35. Un inspector recibe una computadora cada 5 minutos en promedio. El tiempo entre arribos de las computadoras se distribuye exponencialmente. Cada computadora requiere de cinco operaciones consecutivas (cuando una computadora entra a la estación de inspección se deben completar las cinco tareas antes de procesar la computadora siguiente). El tiempo para una operación individual es exponencial con media de 0.9 minutos. Realice una simulación con 3 réplicas de 6 meses cada una y encuentre la utilización promedio del inspector, el tiempo promedio de residencia de las computadoras y el número de computadoras promedio en el proceso.
36. Un vendedor recibe un cliente cada 5 minutos en promedio. El tiempo entre arribos de los clientes se distribuye exponencialmente. El tiempo de atención por cliente es 5-Erlang con media de 4.5 minutos. Realice una simulación con 3 réplicas de 1 mes cada una y encuentre la utilización promedio del vendedor, el tiempo promedio de espera de los clientes y el número de clientes promedio en el proceso.
37. Un grupo de 50 estudiantes organizaron una fiesta para ellos solos. Compraron una canasta con 1000 tacos. Cada uno toma un plato y se sirve 4 ± 1 tacos. El tiempo para servirse un plato es Exponencial con media de 1 minuto. El 30% de ellos termina su plato en un tiempo Normal con media 10 minutos y desviación estándar 2 minutos, el 70% se termina el plato en un tiempo Uniforme entre 10 y 15 minutos. Una vez que se termina su plato regresa a servirse de nuevo. Sólo un estudiante a la vez puede servirse de la canasta, y debe esperar en una fila hasta que se desocupe la canasta.
Codifique en ProModel el sistema anterior para que se detenga cuando se hayan acabado los tacos y determine el tiempo que tardaron los estudiantes en lograrlo.
38. Una línea aérea requiere del diseño de su sistema de reparación de sus aviones. La tasa de llegada de aviones a mantenimiento es Poisson con media de 8 aviones/mes. Cuando un avión requiere de este servicio, se asigna una cuadrilla de trabajadores para realizar estas tareas, el tiempo de reparación de un avión es exponencial con media de 3 días por avión. Encuentre el número óptimo de cuadrillas si:
- a) El costo de mantenimiento es \$100,000/mes-cuadrilla y el costo de que un avión no esté volando es \$1,000,000/mes-avión.

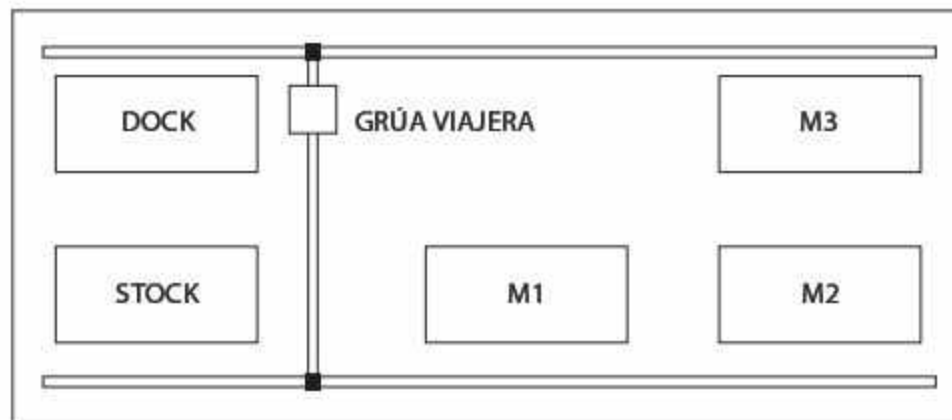
39. Una oficina de recaudación de impuestos requiere determinar el número de cajas a instalar. La tasa de llegadas es Poisson con media de 24 personas/h. El tiempo de atención es exponencial con media de 2 minutos por cliente. Las personas que pueden ser atendidas inmediatamente esperan en una fila común. Encuentre el número óptimo de cajas si:
- El costo de atención es \$200/h-caja y el costo de espera \$20/h-cliente.
 - El costo de dar servicio es \$200/h-caja y el costo de que un cliente espere es \$200/h-cliente.
 - El costo de servicio es \$20/h-caja y el costo de que un cliente espere es \$200/h-cliente.
40. Una oficina dispone actualmente de 3 copiadoras. Los empleados llegan al área de copiado cada 1.5 minutos con distribución 2-Erlang. El tiempo que cada empleado necesita una copiadora es de 4.2 minutos con distribución 3-Erlang. Debido a quejas de los empleados del tiempo perdido hasta que se desocupa una de las copiadoras, la empresa ha decidido evaluar la posibilidad de contratar un mayor número de copiadoras. Desarrolle un modelo de simulación en ProModel y con el uso de Simrunner, encuentre el número de copiadoras a instalar si la renta de cada copiadora es de \$20,000/año y el costo de productividad perdida por el tiempo que pasa un empleado en la sección de copias es \$200/h-empleado.
41. Un banco recibe clientes cada 3 minutos con distribución Exponencial. Los clientes hacen una fila común antes de ser atendidos por cualquiera de las 2 cajas disponibles. El tiempo de atención al cliente sigue una distribución Normal con media de 2 minutos y desviación estándar de 0.5 minutos por cada transacción que el cliente vaya a realizar. Un estimado del número de transacciones por cliente indica que el 20% de los clientes requiere 1 transacción, el 30% hacen 2 transacciones, el 40% solicitan tres transacciones y el 10% restante realiza 4 transacciones.
- El banco opera de 9:00 a 16:00 de lunes a viernes y los sábados de 9:00 a 13:00. Los clientes que se encuentran en el banco al cerrar deben ser atendidos.
- Realice 3 réplicas de un mes cada una y determine:
- Utilización promedio de las cajas.
 - Tiempo promedio de espera de los clientes.
 - Si el banco desea que sus clientes no esperen en promedio más de 5 minutos en la fila, ¿deberá abrir más cajas?, en caso afirmativo ¿cuántas?
42. Un taller de reparación de neumáticos tiene la intención de proporcionar ese servicio de manera eficiente y tener suficientes empleados para que la espera sea mínima. Se estima que el costo de esperar es \$0.60/min-cliente y que el costo de cada empleado es de \$18/h. La tasa promedio de entrada es Poisson con una media de 5 autos/h. Y el tiempo necesario para reparar un neumático es 4-Erlang con media de 20 minutos. Cree un modelo y ejecute con Simrunner 100 réplicas de 8 horas cada una con el fin de encontrar el número de empleados a contratar para minimizar el costo total esperado por hora.

43. Un taller tiene 3 operarios que realizan un proceso de inspección y rectificación de discos del sistema de frenos. Cada operario está asignado solamente a un vehículo, si un automóvil llega al taller y todos los operarios están ocupados, el auto espera en una fila común hasta que se desocupe algún operario. El tiempo entre arribos de los coches es una distribución exponencial con una media de 10 por hora. Cada vehículo requiere de la verificación de los 4 discos y la posible rectificación de alguno de ellos. El proceso inicia con una inspección de uno de los discos, el tiempo para esta primera inspección es de 3 ± 2 minutos, si el disco está en mal estado el operario realiza la rectificación en un tiempo exponencial con media de 4 minutos, y una vez que termina el rectificado hace una segunda inspección con una duración de 2 ± 1 minuto. Si el disco está en buen estado ya no realiza ninguna operación adicional. Una vez terminado con este primer disco, pasa con el segundo repitiendo el proceso de forma similar y así sucesivamente hasta terminar el automóvil. De datos previos se estima que el 35% de los discos inspeccionados requieren de rectificado. Simule el sistema hasta que considere que las variables están en estado estable, replique al menos 3 veces y determine:
- Utilización de los operarios.
 - Número promedio de autos en espera.
 - Longitud máxima de la fila.
 - Tiempo de estancia de un auto en el taller.
 - Número mínimo de operarios con el fin de que un auto espere en promedio menos de una hora en el taller.
44. Se cuenta con dos máquinas para el perforado de dos tipos de partes (Tipo 1 y Tipo 2). Las partes Tipo 1 llegan a una razón de una cada 15 ± 2 minutos y las de Tipo 2, de una cada 5 ± 2 minutos. Las piezas Tipo 1 deben ser perforadas en la máquina EDW. Las partes Tipo B deben ser procesadas por la máquina PGM pero podrán ser procesadas por la máquina EWD en caso de que la máquina PGM esté en reparación. La máquina PGM se descompone cada 8 horas con distribución exponencial y el tiempo de reparación es de 3 ± 1 horas. El tiempo de proceso de las piezas tiene una duración Normal con media de 4 minutos y desviación estándar de 1 minuto. Realice un modelo en ProModel y simule 10 réplicas de un mes cada una para estimar el número promedio de piezas de cada tipo esperando a ser perforadas.
45. Un sistema flexible de manufactura tiene cinco tornos en paralelo. Las piezas llegan en forma Poisson con una media de 18 por hora y pueden ser procesadas en cualquiera de los cinco tornos. El tiempo de torneado es exponencial con una tasa de 4 por hora por torno. Al finalizar el proceso las piezas son enviadas a cualquiera de dos operarios de empaque, el tiempo de empaque es exponencial con media de 6 min/pieza. Simule el sistema con 10 réplicas de una semana cada una y determine la utilización promedio de los tornos y los operarios de empaque, el tiempo promedio de espera de las piezas antes de cada operación y el número de máximo de piezas que se acumularon en el proceso.

46. Un sistema de producción consiste en dos operaciones en serie (operación A y operación B). Las piezas llegan en lotes de tamaño 5, el tiempo entre arribos es exponencial con media 15 min/lote, las piezas se procesan primeramente en la operación A y al salir, el 10% de las piezas salen defectuosas y tienen que repetir la operación A para después ir a la operación B, el 90% restante pasa directamente de la operación A a la operación B.

En la Operación A hay 2 máquinas y en la Operación B hay 3, cada máquina sólo puede procesar una pieza a la vez. El tiempo de operación B es uniforme entre 15 y 25 min/pieza. El tiempo de operación A es Exponencial con media 8 minutos. Antes de cada operación pasan por un almacén de capacidad 10. Cree un programa en ProModel que simule este sistema con 15 réplicas de 1000 piezas cada réplica y determine:

- Utilización de cada operación.
 - Número promedio de piezas en los almacenes.
 - Tiempo promedio de permanencia de las piezas en el sistema.
 - Si fuera necesario, ¿cuántas máquinas agregaría o eliminaría en cada operación?
47. En un sistema de producción se fabrican 4 tipos de válvulas para la industria petrolera, la materia prima es tomada del almacén DOCK y el producto final es depositado en el almacén STOCK. Ambos almacenes tienen capacidad ilimitada. Además, cada máquina M1, M2 y M3 tiene su propio almacén de tamaño 20, 20 y 30 respectivamente. Todos los movimientos de material son realizados por la grúa viajera que se mueve a una velocidad de 40 ft/h cuando transporta una válvula y a 50 ft/h cuando está vacía. En el siguiente diagrama se muestra un esquema de la distribución de planta.



Las distancias (en metros) entre estaciones son las siguientes:

	DOCK	STOCK	M1	M2	M3
DOCK	0	10			20
STOCK	10	0	10	20	
M1		10	0	10	
M2		20	10	0	10
M3	20			10	0

El resto de las distancias se pueden obtener mediante el teorema de Pitágoras.

La siguiente tabla muestra la secuencia de operación, la máquina donde se realiza y el tiempo de proceso de cada tipo de válvula. En todas las máquinas el tiempo de proceso es constante.

Operación	Máquina - Tiempo de proceso (h/pieza)			
	Válvula-1	Válvula-2	Válvula-3	Válvula-4
1	M1 - 10	M2 - 12	M1 - 5	M1 - 2
2	-	M3 - 7	-	M3 - 5
3	M3 - 5	M2 - 2	M3 - 10	M2 - 10

La demanda de cada uno de los productos es:

Producto	Válvula-1	Válvula-2	Válvula-3	Válvula-4
Demanda semanal	10	40	10	20

El proveedor entrega toda la materia prima necesaria para la producción de la semana los lunes a las 6.00 am.

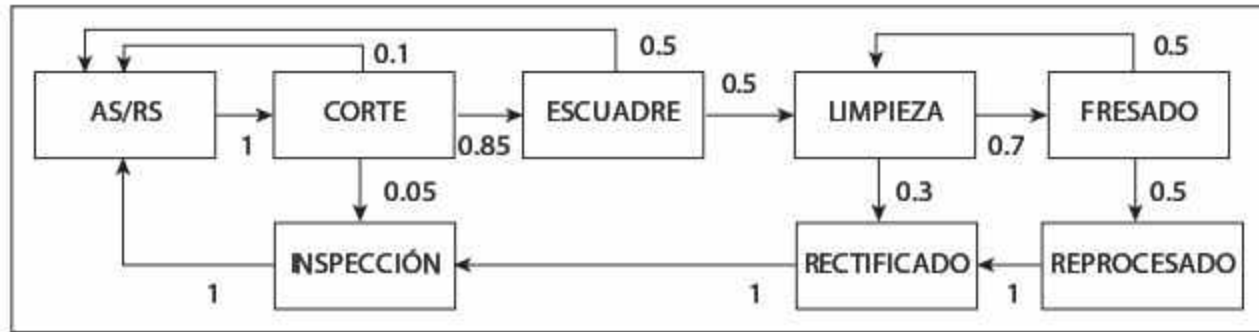
La empresa trabaja solamente dos turnos al día de lunes a viernes.

Mediante ProModel y Simrunner determine:

- a) El número mínimo de máquinas y/o grúas a instalar para cumplir con la demanda.

Y para la solución propuesta, encuentre:

- b) Utilización de cada estación.
 c) La estación cuello de botella.
 d) Número de válvulas promedio en todo el sistema por tipo de producto.
 e) Número de válvulas promedio en cada máquina, incluyendo las que están en los almacenes respectivos.
 f) Tiempo de permanencia en el sistema de cada tipo de válvula.
48. En una celda de manufactura se realizan 8 operaciones. Las operaciones y los flujos de producción se muestran en el diagrama. Los números que se encuentran dentro del diagrama representan la proporción de piezas que salen de una operación y se dirigen a la siguiente. Por ejemplo, del total de las piezas que se procesan en Escuadre, un 50% son enviadas al AS/RS y el otro 50% a Limpieza, del total de piezas que se procesan en Rectificado el 100% son enviadas a Reproceso.
- Un sistema de 50 contenedores permite controlar el número de piezas (1 pieza en cada contenedor) dentro del sistema.
- La piezas entran al sistema en la estación AS/RS y de allí son transportadas dentro de su contenedor a través de la celda, una vez que la pieza es procesada regresa a la estación AS/RS y sale como producto terminado.

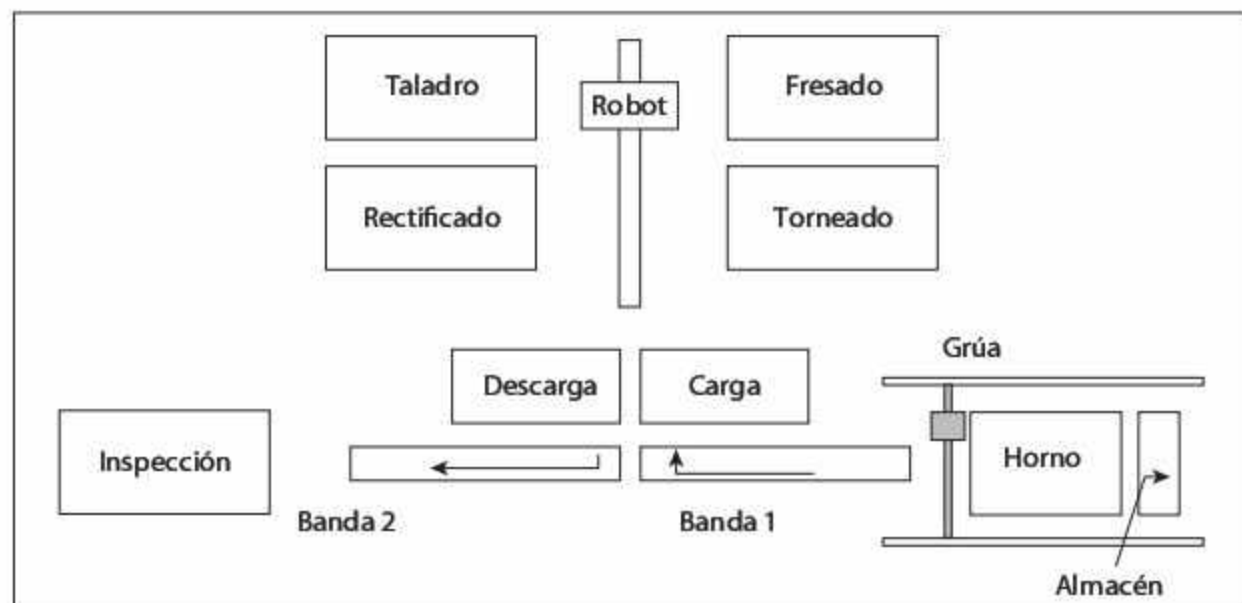


La capacidad de producción de cada equipo es:

Operación	(Piezas/min)
AS/RS	1.00
Corte	2.15
Escuadre	2.00
Reproceso	0.15
Limpieza	0.25
Inspección	1.50
Rectificado	0.40
Fresado	0.50

Simule el proceso de 3 réplicas de un año cada una y determine el índice de producción anual de esta celda.

49. La empresa Engranajes de Acero SA fabrica piezas automotrices en una celda flexible de manufactura. El diagrama muestra la distribución de los equipos en la celda. Las piezas entran del exterior a un almacén de materia prima a una tasa de 12 piezas/h con distribución Poisson y posteriormente son procesados en el horno, la capacidad del horno es de 10 piezas e inicia su operación cuando está lleno.



El flujo de producción es el siguiente:

1. Horno
2. Carga
3. Torneado
4. Fresado
5. Taladrado
6. Rectificado
7. Descarga
8. Inspección

Para cada movimiento entre las operaciones se requiere de las bandas transportadoras, el robot o la grúa viajera de acuerdo con el diagrama anterior. Los tiempos de proceso son:

Operación	Tiempo promedio de proceso (min/pieza)	Número actual de máquinas
Carga y descarga	0.5	Espacio muy grande
Torneado	5.2	1
Fresado	9.17	1
Taladrado	1.6	1
Rectificado	2.85	1
Inspección	Exponencial (3)	1
Horno	100	1

La velocidad de los equipos para el movimiento de materiales es:

Equipo	Velocidad (pies/min)	Cantidad actual de equipos
Robot	45	1
Bandas (la longitud de ambas es de 30ft)	30	2
Grúa viajera	25	1

Las distancias entre los diferentes procesos son las siguientes:

Estación	Estación	Distancia (pies)
Almacén inicial	Horno	10
Horno	Banda 1	15
Torneado y rectificado	Fresado y taladrado	15
Carga y descarga	Torneado y rectificado	20

Con esta información desarrolle un modelo de simulación. Realice 3 réplicas de 1000 horas cada uno y determine:

- a) La utilización de los equipos.
- b) La producción de la celda.
- c) Identifique la estación cuello de botella.

Determine el número mínimo de máquinas que deberían ser instaladas en cada estación para maximizar la producción y que la estación considerada como cuello de botella trabaje a una utilización lo más cercana al 95%.



Anexo 1

Distribuciones de probabilidad

A.1 Distribuciones continuas

Distribución de Erlang

Codificación	$k\text{-Er}(1/\lambda)$
Función de densidad	$f(x) = \frac{\lambda(\lambda x)^{k-1}}{(k-1)!} e^{-\lambda x}$ para $x \geq 0$
Distribución acumulada	$F(x) = 1 - \frac{\Gamma(k, x\lambda)}{\Gamma(k)}$ para $x \geq 0$
Parámetros	Parámetro de localización: 0 Parámetro de forma: k Parámetro de escala: $\frac{1}{\lambda} \geq 0$
Rango	$[0, \infty)$
Media	$k \left(\frac{1}{\lambda} \right)$
Varianza	$k \left(\frac{1}{\lambda} \right)^2$
	<p style="text-align: center;">Erlang (0, 3, 5)</p>

Distribución exponencial

Codificación	$E(1/\lambda)$
Función de densidad	$f(x) = \begin{cases} \lambda e^{-\lambda x} & \text{si } x \geq 0 \\ 0 & \text{en otro caso} \end{cases}$
Distribución acumulada	$F(x) = \begin{cases} 1 - e^{-\lambda x} & \text{si } x \geq 0 \\ 0 & \text{en otro caso} \end{cases}$
Parámetros	Parámetro de escala: $\frac{1}{\lambda} \geq 0$
Rango	$[0, \infty)$
Media	$\frac{1}{\lambda}$
Varianza	$\frac{1}{\lambda^2}$
	<p style="text-align: center;">Exponencial (2)</p> <p>El gráfico muestra la función de densidad $f(x) = 2e^{-2x}$ para $x \geq 0$. El eje horizontal x tiene marcas en 0.0, 2.0, 4.0, 6.0, 8.0 y 10.0. El eje vertical $f(x)$ tiene marcas en 0.00, 0.25 y 0.50. La curva parte en el punto (0, 0.50) y se aproxima asintóticamente al eje x a medida que x aumenta.</p>

Distribución gamma

Codificación	$G(\alpha, \beta)$
Función de densidad	$f(x) = \frac{\beta^{-\alpha} x^{\alpha-1}}{\Gamma(\alpha)} e^{-x/\beta}$ para $x \geq 0$
Distribución acumulada	$F(x) = \begin{cases} - & \text{para } \alpha \text{ no entera} \\ 1 - e^{-x/\beta} \sum_{j=0}^{\alpha-1} \frac{(x/\beta)^j}{j!} & \text{para } \alpha \text{ entera} \end{cases}$
Parámetros	Parámetro de forma: α Parámetro de escala: $\beta \geq 0$
Rango	$[0, \infty)$
Media	α, β
Varianza	α, β^2
	<p style="text-align: center;">Gamma (3.86, 5.25)</p>

Distribución log-logística

Codificación	$LL(\alpha, \beta)$
Función de densidad	$f(x) = \frac{\alpha(x/\beta)^{\alpha-1}}{\beta[1+(x/\beta)^\alpha]^2}$ para $x > 0$
Distribución acumulada	$F(x) = \frac{1}{1+(x/\beta)^\alpha}$ para $x > 0$
Parámetros	Parámetro de escala: $\beta > 0$ Parámetro de forma: $\alpha > 0$
Rango	$[0, \infty)$
Media	$\frac{\pi\beta}{\alpha} \operatorname{cosecante}\left(\frac{\pi}{\alpha}\right)$
Varianza	$\frac{\pi\beta^2}{\alpha} \left\{ 2 \operatorname{cosecante}\left(\frac{2\pi}{\alpha}\right) - \frac{\pi}{\alpha} \left[\operatorname{cosecante}\left(\frac{\pi}{\alpha}\right) \right]^2 \right\}$
	<p style="text-align: center;">Log-Logística (3, 1)</p>

Distribución lognormal

Codificación	$LN(\mu, \sigma^2)$
Función de densidad	$f(x) = \frac{1}{x\sqrt{2\pi\sigma^2}} e^{-\frac{(\ln x - \mu)^2}{2\sigma^2}}$ para $x > 0$
Distribución acumulada	–
Parámetros	Parámetro de escala: μ Parámetro de forma: $\sigma > 0$
Rango	$[0, \infty)$
Media	$e^{\mu + \sigma^2/2}$
Varianza	$(e^{\sigma^2} - 1)e^{2\mu + \sigma^2}$
	<p>Lognormal (4, 0.792)</p>

Distribución normal

Codificación	$N(\mu, \sigma^2)$
Función de densidad	$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$ para $x > -\infty$
Distribución acumulada	–
Parámetros	Parámetro de localización: μ Parámetro de escala: $\sigma > 0$
Rango	$(-\infty, \infty)$
Media	μ
Varianza	σ^2
	<div style="text-align: center;">Normal (10, 2)</div>

Distribución triangular

Codificación	$T(a, b, c)$
Función de densidad	$f(x) = \begin{cases} \frac{2(x-a)}{(b-a)(c-a)} & \text{si } a \leq x \leq c \\ \frac{2(b-x)}{(b-a)(b-c)} & \text{si } c \leq x \leq b \end{cases}$
Distribución acumulada	$F(x) = \begin{cases} \frac{(x-a)^2}{(b-a)(c-a)} & \text{si } a \leq x \leq c \\ 1 - \frac{(b-x)^2}{(b-a)(b-c)} & \text{si } c \leq x \leq b \end{cases}$
Parámetros	Parámetro de localización: a Parámetro de escala: $b - a$ Parámetro de forma: c Valor mínimo: a Valor máximo: b Moda: c
Rango	$[a, b]$
Media	$\frac{1}{3}(a + b + c)$
Varianza	$\frac{1}{18}(a^2 + b^2 + c^2 - ab - ac - bc)$
	<p style="text-align: center;">Triangular (1,7,3)</p>

Distribución uniforme

Codificación	$U(a, b)$
Función de densidad	$f(x) = \begin{cases} \frac{1}{b-a} & \text{si } a \leq x \leq b \\ 0 & \text{en otro caso} \end{cases}$
Distribución acumulada	$F(x) = \begin{cases} 0 & \text{si } x < a \\ \frac{x-a}{b-a} & \text{si } a \leq x \leq b \\ 1 & \text{si } b < x \end{cases}$
Parámetros	<p>a y b son números reales con $a < b$; Parámetro de localización: a Parámetro de escala: $b - a$ Valor mínimo: a Valor máximo: b</p>
Rango	$[a, b]$
Media	$\frac{1}{2}(a+b)$
Varianza	$\frac{1}{12}(b-a)^2$
	<p style="text-align: center;">Uniforme (5, 10)</p>

Distribución de Weibull

Codificación	$W(\gamma, \alpha, \beta)$
Función de densidad	$f(x) = \begin{cases} \alpha\beta^{-\alpha}(x-\gamma)^{\alpha-1}e^{-\left(\frac{x-\gamma}{\beta}\right)^\alpha} & \text{si } x \geq \gamma \\ 0 & \text{en otro caso} \end{cases}$
Distribución acumulada	$F(x) = \begin{cases} 1 - e^{-\left(\frac{x-\gamma}{\beta}\right)^\alpha} & \text{si } x \geq \gamma \\ 0 & \text{en otro caso} \end{cases}$
Parámetros	Parámetro de localización: γ Parámetro de escala: $\beta > 0$ Parámetro de forma: $\alpha > 0$
Rango	$[\gamma, \infty)$
Media	$\gamma + \beta\Gamma(1 + \alpha^{-1})$
Varianza	$\beta^2 [\Gamma(1 + 2\alpha^{-1}) - \Gamma^2(1 + \alpha^{-1})]$
	<p style="text-align: center;">Weibull (4,2,1)</p>

A.2 Distribuciones discretas

Distribución de Bernoulli

Codificación	$BE(p)$
Distribución de probabilidad	$p(x) = \begin{cases} 1-p & \text{si } x=0 \\ p & \text{si } x=1 \\ 0 & \text{otro caso} \end{cases}$
Distribución acumulada	$P(x) = \begin{cases} 0 & \text{si } x < 0 \\ 1-p & \text{si } 0 \leq x < 1 \\ 1 & \text{si } x \geq 1 \end{cases}$
Rango	$\{0, 1\}$
Parámetros	$p \in (0, 1)$
Media	p
Varianza	$p(1-p)$
	<p style="text-align: center;">Bernoulli (0.7)</p> <p>El gráfico muestra la función de probabilidad $p(x)$ para una distribución de Bernoulli con $p=0.7$. El eje horizontal x tiene marcas en -1.0, 0.0, 1.0, 2.0, 3.0, 4.0 y 5.0. El eje vertical $p(x)$ tiene marcas en 0.00, 0.35 y 0.70. Se observan dos barras: una en $x=0$ con altura 0.30 (representando $1-p$) y una en $x=1$ con altura 0.70 (representando p).</p>

Distribución binomial

Codificación	$BI(N, p)$														
Distribución de probabilidad	$p(x) = \begin{cases} \frac{N!}{x!(N-x)!} p^x (1-p)^{N-x} & \text{si } x = 0, 1, 2, \dots, N \\ 0 & \text{otro caso} \end{cases}$														
Distribución acumulada	$P(x) = \begin{cases} 0 & \text{si } x < 0 \\ \sum_{i=0}^{\lfloor x \rfloor} \frac{N!}{i!(N-i)!} p^i (1-p)^{N-i} & \text{si } 0 \leq x < 1 \\ 1 & \text{si } x > 1 \end{cases}$														
Rango	$\{0, 1, \dots, N\}$														
Parámetros	N es un número entero $p \in (0, 1)$														
Media	Np														
Varianza	$Np(1-p)$														
	<p style="text-align: center;">Binomial (5, 0.5)</p> <table border="1" style="display: none;"> <caption>Datos del gráfico Binomial (5, 0.5)</caption> <thead> <tr> <th>x</th> <th>p(x)</th> </tr> </thead> <tbody> <tr><td>0</td><td>0.031</td></tr> <tr><td>1</td><td>0.156</td></tr> <tr><td>2</td><td>0.312</td></tr> <tr><td>3</td><td>0.312</td></tr> <tr><td>4</td><td>0.156</td></tr> <tr><td>5</td><td>0.031</td></tr> </tbody> </table>	x	p(x)	0	0.031	1	0.156	2	0.312	3	0.312	4	0.156	5	0.031
x	p(x)														
0	0.031														
1	0.156														
2	0.312														
3	0.312														
4	0.156														
5	0.031														

Distribución geométrica

Codificación	$GEO(p)$																								
Distribución de probabilidad	$p(x) = \begin{cases} p(1-p)^x & \text{si } x = 0, 1, \dots \\ 0 & \text{otro caso} \end{cases}$																								
Distribución acumulada	$P(x) = \begin{cases} 1 - (1-p)^{x+1} & \text{si } x = 0, 1, \dots \\ 0 & \text{otro caso} \end{cases}$																								
Rango	$\{0, 1, \dots\}$																								
Parámetros	$p \in (0, 1)$																								
Media	$(1-p)p^{-1}$																								
Varianza	$(1-p)p^{-2}$																								
	<p style="text-align: center;">Geométrica (0.5)</p> <table border="1" style="display: none;"> <caption>Datos del gráfico Geométrica (0.5)</caption> <thead> <tr> <th>x</th> <th>p(x)</th> </tr> </thead> <tbody> <tr><td>0</td><td>0.50</td></tr> <tr><td>1</td><td>0.25</td></tr> <tr><td>2</td><td>0.125</td></tr> <tr><td>3</td><td>0.0625</td></tr> <tr><td>4</td><td>0.03125</td></tr> <tr><td>5</td><td>0.015625</td></tr> <tr><td>6</td><td>0.0078125</td></tr> <tr><td>7</td><td>0.00390625</td></tr> <tr><td>8</td><td>0.001953125</td></tr> <tr><td>9</td><td>0.0009765625</td></tr> <tr><td>10</td><td>0.00048828125</td></tr> </tbody> </table>	x	p(x)	0	0.50	1	0.25	2	0.125	3	0.0625	4	0.03125	5	0.015625	6	0.0078125	7	0.00390625	8	0.001953125	9	0.0009765625	10	0.00048828125
x	p(x)																								
0	0.50																								
1	0.25																								
2	0.125																								
3	0.0625																								
4	0.03125																								
5	0.015625																								
6	0.0078125																								
7	0.00390625																								
8	0.001953125																								
9	0.0009765625																								
10	0.00048828125																								

Distribución de Poisson

Codificación	$P(\lambda)$																																												
Distribución de probabilidad	$p(x) = \begin{cases} \frac{\lambda^x}{x!} e^{-\lambda} & \text{si } x = 0, 1, 2, \dots \\ 0 & \text{otro caso} \end{cases}$																																												
Distribución acumulada	$P(x) = \begin{cases} 0 & \text{si } x < 0 \\ \sum_{l=0}^{\lfloor x \rfloor} \frac{\lambda^l}{l!} e^{-\lambda} & \text{si } x \geq 0 \end{cases}$																																												
Rango	$\{0, 1, \dots\}$																																												
Parámetros	$\lambda > 0$ es un número entero																																												
Media	λ																																												
Varianza	λ																																												
	<p style="text-align: center;">Poisson (4)</p> <table border="1"> <caption>Data for Poisson (4) distribution</caption> <thead> <tr> <th>x</th> <th>p(x)</th> </tr> </thead> <tbody> <tr><td>0</td><td>0.0183</td></tr> <tr><td>1</td><td>0.0733</td></tr> <tr><td>2</td><td>0.1465</td></tr> <tr><td>3</td><td>0.1954</td></tr> <tr><td>4</td><td>0.1954</td></tr> <tr><td>5</td><td>0.1563</td></tr> <tr><td>6</td><td>0.1042</td></tr> <tr><td>7</td><td>0.0625</td></tr> <tr><td>8</td><td>0.0390</td></tr> <tr><td>9</td><td>0.0253</td></tr> <tr><td>10</td><td>0.0162</td></tr> <tr><td>11</td><td>0.0108</td></tr> <tr><td>12</td><td>0.0072</td></tr> <tr><td>13</td><td>0.0048</td></tr> <tr><td>14</td><td>0.0032</td></tr> <tr><td>15</td><td>0.0021</td></tr> <tr><td>16</td><td>0.0014</td></tr> <tr><td>17</td><td>0.0009</td></tr> <tr><td>18</td><td>0.0006</td></tr> <tr><td>19</td><td>0.0004</td></tr> <tr><td>20</td><td>0.0003</td></tr> </tbody> </table>	x	p(x)	0	0.0183	1	0.0733	2	0.1465	3	0.1954	4	0.1954	5	0.1563	6	0.1042	7	0.0625	8	0.0390	9	0.0253	10	0.0162	11	0.0108	12	0.0072	13	0.0048	14	0.0032	15	0.0021	16	0.0014	17	0.0009	18	0.0006	19	0.0004	20	0.0003
x	p(x)																																												
0	0.0183																																												
1	0.0733																																												
2	0.1465																																												
3	0.1954																																												
4	0.1954																																												
5	0.1563																																												
6	0.1042																																												
7	0.0625																																												
8	0.0390																																												
9	0.0253																																												
10	0.0162																																												
11	0.0108																																												
12	0.0072																																												
13	0.0048																																												
14	0.0032																																												
15	0.0021																																												
16	0.0014																																												
17	0.0009																																												
18	0.0006																																												
19	0.0004																																												
20	0.0003																																												

Distribución uniforme discreta

Codificación	$UD(i, j)$
Distribución de probabilidad	$p(x) = \begin{cases} \frac{1}{j-i+1} & \text{si } x \in \{i, i+1, \dots, j\} \\ 0 & \text{en otro caso} \end{cases}$
Distribución acumulada	$P(x) = \begin{cases} 0 & \text{si } x < i \\ \frac{\lfloor x \rfloor - i + 1}{j - i + 1} & \text{si } i \leq x \leq j \\ 1 & \text{si } x > j \end{cases}$
Rango	$\{i, i+1, \dots, j\}$
Parámetros	i y j son números enteros con $i < j$; Parámetro de localización: i Parámetro de escala: $j - i$
Media	$\frac{1}{2}(i + j)$
Varianza	$\frac{1}{12}[(j - i + 1)^2 - 1]$
	<p style="text-align: center;">Uniforme discreta (1, 6)</p> <p style="text-align: center;">x</p>



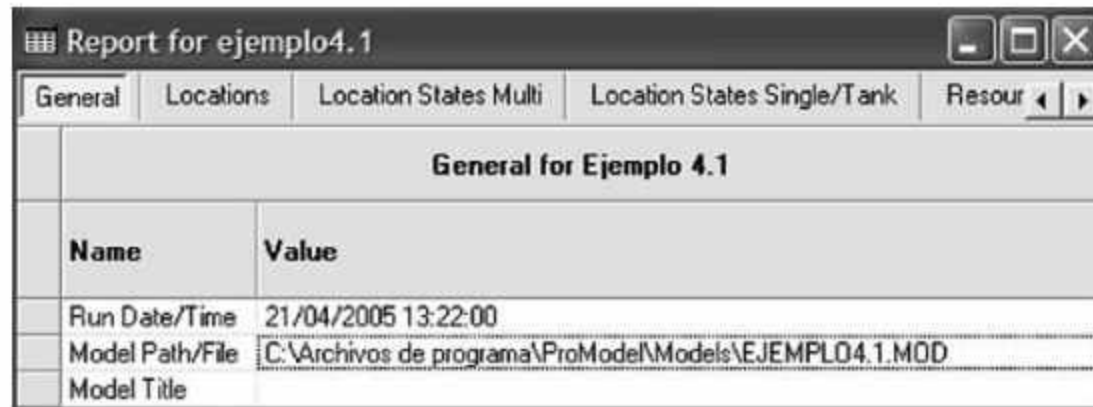
Anexo 2

Reportes estadísticos en ProModel
(formato Viewer 2.0)

En ProModel los reportes con formato Viewer 2.0(3DR) contienen los resultados numéricos de la simulación, los cuales se presentan en un formato de hoja de cálculo.

El reporte contiene la información separada en las siguientes hojas:

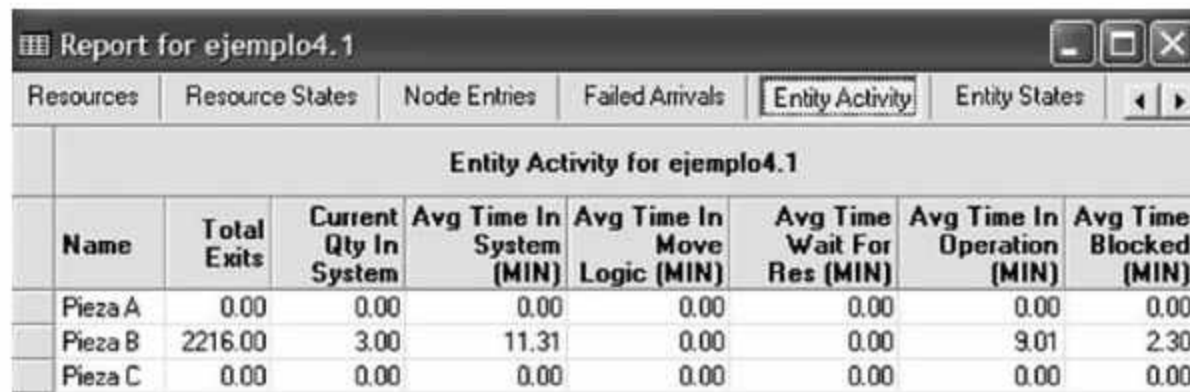
General



Name	Value
Run Date/Time	21/04/2005 13:22:00
Model Path/File	C:\Archivos de programa\ProModel\Models\EJEMPLO4.1.MOD
Model Title	

En esta sección se encuentran los datos generales del modelo como nombre, fecha de ejecución y el lugar donde se encuentra el archivo.

Entity Activity



Name	Total Exits	Current Qty In System	Avg Time In System (MIN)	Avg Time In Move Logic (MIN)	Avg Time Wait For Res (MIN)	Avg Time In Operation (MIN)	Avg Time Blocked (MIN)
Pieza A	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Pieza B	2216.00	3.00	11.31	0.00	0.00	9.01	2.30
Pieza C	0.00	0.00	0.00	0.00	0.00	0.00	0.00

- **Total Exits.** Número de entidades que abandonaron el sistema.
- **Current Quantity In System.** Número de entidades que permanecen en el sistema al finalizar la simulación.
- **Avg Time In System.** Tiempo promedio que las entidades permanecieron en el sistema simulado.
- **Avg Time In Move Logic.** Tiempo promedio que las entidades permanecieron viajando entre las localizaciones.
- **Avg Time Wait For Res.** Tiempo promedio que las entidades permanecieron esperando por un recurso o por otra entidad.
- **Avg Time In Operation.** Tiempo promedio que las entidades permanecieron en operación o en una banda transportadora.

- **Avg Time Blocked.** Tiempo promedio que las entidades permanecieron esperando por una localización desocupada.

Entities Costing

General Report (Baseline - Rep. 1)				
Location States Single		Resources	Resource States	Ni
Mfg_cost.MOD (Baseline - Rep. 1)				
Name	Explicit Exits	Total Cost Dollars	% Total Cost	
Pallet	93.00	0.00	0.00	
Blank	0.00	0.00	0.00	
Cog	391.00	6433.84	72.06	
Reject	167.00	2494.48	27.94	
Bearing	0.00	0.00	0.00	

- **Name.** Nombre de la entidad.
- **Explicit Exits.** Número de entidades que abandonaron el sistema.
- **Total Cost.** Costo total de las entidades.
- **% Total Cost.** Porcentaje del costo de las entidades correspondiente a la entidad activa.

Entity States By Percentage

Report for ejemplo4.1				
Location States Multi		Location States Single/Tank	Resources	Resource Sta
Entity States for ejemplo4.1				
Name	% In Move Logic	% Wait For Res	% In Operation	% Blocked
Pieza A	0.00	0.00	0.00	0.00
Pieza B	0.00	0.00	79.70	20.30
Pieza C	0.00	0.00	0.00	0.00

- **Name.** Nombre de la entidad.
- **% In Move Logic.** Porcentaje del tiempo que las entidades permanecieron viajando entre localizaciones.
- **% Wait For Res.** Porcentaje del tiempo que las entidades permanecieron esperando por un recurso o por otra entidad.
- **% In Operation.** Porcentaje del tiempo promedio que las entidades permanecieron en operación o en una banda transportadora.
- **% Blocked.** Porcentaje del tiempo promedio que las entidades permanecieron esperando por una localización desocupada.

Failed Arrival

Report for ejemplo4.1		
General	Locations	Location States Multi
Failed Arrivals for ejemplo4.1		
Entity Name	Location Name	Total Failed
Pieza B	Almacen 1	0.00

- **Entity Name.** Nombre de la entidad.
- **Location Name.** Localización donde ocurre la llegada de la entidad.
- **Total Failed.** Número de entidades que no entraron a la localización por falta de capacidad.

Locations Costing

Report for deaerco						
Node Entries	Failed Arrivals	Entity Activity	Entity States	Variables	Location Costing	
Location Costing for deaerco						
Name	Operation Cost Dollars	% Operation Cost	Resource Cost Dollars	% Resource Cost	Total Cost Dollars	% Total Cost
Aisle 1	0.00	0.00	30.89	4.62	30.89	4.62
Aisle 2	0.00	0.00	34.36	5.13	34.36	5.13
Aisle 3	0.00	0.00	45.93	6.86	45.93	6.86
Aisle 4	0.00	0.00	30.04	4.49	30.04	4.49
Aisle 5	0.00	0.00	40.87	6.11	40.87	6.11
Aisle 6	0.00	0.00	26.65	3.98	26.65	3.98
Aisle 7	0.00	0.00	40.10	5.99	40.10	5.99

- **Name.** Nombre de la localización.
- **Operation Cost.** Costo del tiempo de operación.
- **% Operation Cost.** Porcentaje del costo de la localización sobre la suma de los costos de operación.
- **Resource Cost.** Costo por el uso de un recurso operando dentro de la localización.
- **% Resource Cost.** Porcentaje del costo de recurso en la localización sobre la suma de los costos del resto de los recursos.
- **Total Cost.** Costo de operación más costo del recurso.
- **% Total Cost.** Porcentaje del costo total de la localización sobre la suma de los costos totales de las localizaciones.

Location States By Percentage (Multiple Capacity)

Report for ejemplo4.1						
General	Locations	Location States Multi	Location States Single/Tank	Resources		
Location States Multi for ejemplo4.1						
Name	Scheduled Time (MIN)	% Empty	% Part Occupied	% Full	% Down	
Almacen 1	19200.40	82.91	17.09	0.00	0.00	
Almacen 2	19200.40	99.38	0.62	0.00	0.00	

- **Name.** Nombre de la localización.
- **Scheduled Time.** Porcentaje de tiempo en operación.
- **% Empty.** Porcentaje de tiempo que la localización estuvo vacía.
- **% Partially Occupied.** Porcentaje de tiempo que la localización estuvo parcialmente llena.
- **% Full.** Porcentaje de tiempo que la localización estuvo llena.
- **% Down.** Porcentaje de tiempo que la localización estuvo no disponible por paros no programados.

Location States By Percentage (Single Capacity/Tank)

Report for ejemplo4.1								
Locations	Location States Multi	Location States Single/Tank	Resources	Resource				
Location States Single/Tank for ejemplo4.1								
Name	Scheduled Time (MIN)	% Operation	% Setup	% Idle	% Waiting	% Blocked	% Down	
Lavadora	19200.40	46.15	0.00	53.85	0.00	0.00	0.00	
Inspector.1	19200.40	41.32	0.00	58.68	0.00	0.00	0.00	
Inspector.2	19200.40	16.59	0.00	83.41	0.00	0.00	0.00	
Inspector	38400.80	28.95	0.00	71.05	0.00	0.00	0.00	

- **Name.** Nombre de la localización.
- **Scheduled Time.** Tiempo total programado.
- **% Operation.** Porcentaje de tiempo que la localización estuvo en operación.
- **% Setup.** Porcentaje de tiempo que la localización estuvo en preparación.
- **% Idle.** Porcentaje de tiempo que la localización estuvo ociosa por falta de entidades.
- **% Waiting.** Porcentaje de tiempo esperando por un recurso o por otra entidad para agruparse, unirse, etcétera.
- **% Blocked.** Porcentaje de tiempo que las entidades permanecieron bloqueadas en la localización.
- **% Down.** Porcentaje de tiempo por paros no programados.

Locations

Locations for ejemplo4.1									
Name	Scheduled Time (MIN)	Capacity	Total Entries	Avg Time Per Entry (MIN)	Avg Contents	Maximum Contents	Current Contents	% Utilization	
Almacen 1	19200.40	999999.00	2219.00	2.25	0.26	5.00	1.00	0.00	
Lavadora	19200.40	1.00	2218.00	4.00	0.46	1.00	1.00	46.15	
Almacen 2	19200.40	999999.00	2217.00	0.05	0.01	2.00	0.00	0.00	
Inspector.1	19200.40	1.00	1584.00	5.01	0.41	1.00	1.00	41.32	
Inspector.2	19200.40	1.00	633.00	5.03	0.17	1.00	0.00	16.59	
Inspector	38400.80	2.00	2217.00	5.02	0.29	2.00	1.00	28.95	

- **Name.** Nombre de la localización.
- **Scheduled Time.** Tiempo total programado de la localización.
- **Capacity.** Capacidad de la localización.
- **Total Entries.** Total de entidades que entraron a la localización.
- **Avg Time Per Entry.** Tiempo promedio de permanencia en la localización.
- **Avg Contents.** Número promedio de entidades en la localización.
- **Maximum Contents.** Número máximo de entidades en la localización en el transcurso de la simulación.
- **Current Contents.** Número de entidades en la localización al final de la simulación.
- **% Utilization.** Porcentaje de utilización.

Logs

Logs for ejemplo4.1				
Name	Number Observations	Minimum Value	Maximum Value	Avg Value

- **Name.** Identificador de registro.
- **Numbers Observations.** Número de entidades que ejecutaron la instrucción LOG durante la simulación.
- **Minimum Value.** El mínimo valor de tiempo registrado por la instrucción LOG.
- **Maximum Value.** El máximo valor de tiempo registrado por la instrucción LOG.
- **Avg Value.** El valor promedio de tiempo registrado por la instrucción LOG.

Node Entries

Path Name	Node Name	Total Entries	Blocked Entries
-----------	-----------	---------------	-----------------

- **Path Name.** Nombre de la trayectoria.
- **Node Name.** Número del nodo.
- **Total Entries.** Total de entradas de un recurso a un nodo.
- **Blocked Entries.** Número de veces que el nodo estuvo bloqueado por otro recurso.

Resources Costing

Name	Units	NonUse Cost Dollars	% NonUse Cost	Usage Cost Dollars	% Usage Cost	Total Cost Dollars	% Total Cost
Filler.1	1.00	0.80	0.45	99.18	12.10	99.98	10.04
Filler.2	1.00	0.80	0.45	99.08	12.09	99.88	10.03
Filler.3	1.00	0.80	0.45	98.88	12.07	99.68	10.01
Filler.4	1.00	0.81	0.46	99.03	12.09	99.83	10.02
Filler.5	1.00	0.83	0.47	99.13	12.10	99.96	10.03
Filler	5.00	4.04	2.28	495.29	60.45	499.33	50.12
Checker	1.00	11.39	6.44	87.79	10.71	99.18	9.96

- **Name.** Nombre del recurso.
- **NonUse Cost.** Costo atribuible al ocio del recurso.
- **% NonUse Cost.** Porcentaje del costo de ocio del recurso sobre la suma de los costos de ocio de los recursos.
- **Usage Cost.** Costo por usar el recurso.
- **% Usage Cost.** Porcentaje del costo de utilización del recurso sobre la suma de los costos de utilización de los recursos.
- **Total Cost.** Costo de utilización más costo de ocio.
- **% Total Cost.** Porcentaje del costo total del recurso sobre la suma de los costos totales de los recursos.

Resource States By Percentage

Resource States for deaerco						
Name	Scheduled Time (MIN)	% In Use	% Travel To Use	% Travel To Park	% Idle	% Down
Filler.1	600.00	98.60	0.60	0.00	0.80	0.00
Filler.2	600.00	98.60	0.60	0.00	0.80	0.00
Filler.3	600.00	98.57	0.62	0.00	0.80	0.00
Filler.4	600.00	98.63	0.57	0.00	0.81	0.00
Filler.5	600.00	98.60	0.57	0.00	0.83	0.00
Filler	3000.00	98.60	0.59	0.00	0.81	0.00
Checker	600.00	85.14	3.47	0.00	11.39	0.00

- **Name.** Nombre del recurso.
- **Scheduled Time.** Total de tiempo que el recurso fue programado para estar disponible.
- **% In Use.** Porcentaje de tiempo que el recurso fue utilizado.
- **% Travel To Use.** Porcentaje de tiempo que el recurso fue utilizado en movimientos entre localizaciones.
- **% Travel To Park.** Porcentaje de tiempo que el recurso estuvo viajando a su nodo base.
- **% Idle.** Porcentaje de ocio del recurso.
- **% Down.** Porcentaje que el recurso estuvo no disponible a causa de paros no programados.

Resources

Resources for deaerco								
Name	Units	Scheduled Time (MIN)	Number Times Used	Avg Time Per Usage (MIN)	Avg Time Travel To Use (MIN)	Avg Time Travel To Park (MIN)	% Blocked In Travel	% Utilization
Filler.1	1.00	600.00	33.00	17.93	0.11	0.00	0.00	99.20
Filler.2	1.00	600.00	33.00	17.93	0.11	0.00	0.00	99.20
Filler.3	1.00	600.00	34.00	17.40	0.11	0.00	0.00	99.20
Filler.4	1.00	600.00	31.00	19.09	0.11	0.00	0.00	99.19
Filler.5	1.00	600.00	31.00	19.08	0.11	0.00	0.00	99.17
Filler	5.00	3000.00	162.00	18.26	0.11	0.00	0.00	99.19
Chec...	1.00	600.00	134.00	3.81	0.16	0.00	0.00	88.61

- **Name.** Nombre del recurso.
- **Units.** Número de recursos.
- **Scheduled Time.** Tiempo programado para utilizar el recurso.
- **Number of Times Used.** Número de ocasiones que se utilizó el recurso.
- **Avg Time Per Usage.** Tiempo promedio de utilización del recurso.
- **Avg Time Travel To Use.** Tiempo promedio por viaje del recurso.
- **Avg Time Travel To Park.** Tiempo promedio para dirigirse al nodo base.
- **% Blocked In Travel.** Porcentaje de tiempo que el recurso estuvo bloqueado al final del viaje.
- **% Utilization.** Porcentaje de utilización del recurso.

Variables

Variables for deaerco							
Name	Total Changes	Avg Time Per Change...	Minimum Value	Maximum Value	Current Value	Avg Value	
Inventory Aisle 1	69.00	8.56	2891.00	3600.00	2891.00	3261.05	
Inventory Aisle 2	86.00	6.88	2678.00	3600.00	2678.00	3138.78	
Inventory Aisle 3	108.00	5.50	2384.00	3600.00	2384.00	2984.03	
Inventory Aisle 4	70.00	8.46	2883.00	3600.00	2883.00	3234.83	
Inventory Aisle 5	107.00	5.58	2457.00	3600.00	2457.00	3019.44	
Inventory Aisle 6	60.00	9.97	2974.00	3600.00	2992.00	3254.74	
Inventory Aisle 7	117.00	5.12	2307.00	3600.00	2307.00	2930.76	

- **Name.** Nombre de la variable.
- **Total Changes.** Total de cambios de valor de la variable.
- **Average time Per Change.** El tiempo promedio entre cambios de valor de la variable.
- **Minimum Value.** Valor mínimo de la variable en el transcurso de la simulación.
- **Maximum Value.** Valor máximo de la variable en el transcurso de la simulación.
- **Current Value.** Valor de la variable al finalizar la simulación.
- **Avg Value.** Valor promedio de la variable a través del tiempo.



Anexo 3

Reportes estadísticos en ProModel

En ProModel los reportes en formato Viewer 4.0 contienen los resultados numéricos de la simulación, los cuales se presentaron en tablas con los siguientes formatos.

El reporte contiene la información separada en las siguientes hojas:

General

Name	Ejemplo 4.1
Run Date	01/11/2011 04:53:06 p.m.
Avg. Warmup Duration (Hr)	0
Avg. Simulation Duration (Hr)	1000

En esta sección se encuentran los datos generales del modelo, como nombre, fecha de ejecución, tiempo de calentamiento y tiempo de simulación.

Entity Activity

Name	Total Exits	Current Quantity In System	Average Time In System (Min)	Average Time In Move Logic (Min)	Average Time Waiting (Min)	Average Time In Operation (Min)	Average Time Blocked (Min)
Cliente	10,789.33	1.67	15.95	0.00	0.00	15.20	0.75
Contador	60,001.00	0.00	0.00	0.00	0.00	0.00	0.00

- **Name.** Nombre de la entidad.
- **Total Exits.** Número de entidades que abandonaron el sistema.
- **Current Quantity In System.** Número de entidades que permanecen en el sistema al finalizar la simulación.
- **Avg Time In System.** Tiempo promedio que las entidades permanecieron en el sistema simulado.
- **Avg Time In Move Logic.** Tiempo promedio que las entidades permanecieron viajando entre las localizaciones.
- **Avg Time Waiting.** Tiempo promedio que las entidades permanecieron esperando por un recurso o por otra entidad.
- **Avg Time In Operation.** Tiempo promedio que las entidades permanecieron en operación o en una banda transportadora.
- **Avg Time Blocked.** Tiempo promedio que las entidades permanecieron esperando por una localización desocupada.

Entities Costing

Scenario	Name	Explicit Exits	Total Cost	% Total Cost
Baseline	Bearing	0.00	0.00	0.00
Baseline	Blank	0.00	0.00	0.00
Baseline	Cog	391.00	6,433.84	72.06
Baseline	Pallet	93.00	0.00	0.00
Baseline	Reject	167.00	2,494.48	27.94

- **Name.** Nombre de la entidad.
- **Explicit Exits.** Número de entidades que abandonaron el sistema.
- **Total Cost.** Costo total de las entidades.
- **% Total Cost.** Porcentaje del costo de las entidades correspondiente a la entidad activa.

Entity States By Percentage

Name	% In Move Logic	% Waiting	% In Operation	% Blocked
Ciente	0.00	0.00	95.31	4.69
Contador	0.00	0.00	0.00	0.00

- **Name.** Nombre de la entidad.
- **% In Move Logic.** Porcentaje del tiempo que las entidades permanecieron viajando entre localizaciones.
- **% Waiting.** Porcentaje del tiempo que las entidades permanecieron esperando por un recurso o por otra entidad.
- **% In Operation.** Porcentaje del tiempo promedio que las entidades permanecieron en operación o en una banda transportadora.
- **% Blocked.** Porcentaje del tiempo promedio que las entidades permanecieron esperando por una localización desocupada.

Failed Arrival

Name	Location	Total Failed
Cliente	Fila	1,697.67
Contador	Oficina	0.00

- **Name.** Nombre de la entidad.
- **Location name.** Localización donde ocurre la llegada de la entidad.
- **Failed Arrivals.** Número de entidades que no entraron a la localización por falta de capacidad.

Locations Costing

Scenario	Name	Operation Cost	% Operation Cost	Resource Cost	% Resource Cost	Total Cost	% Total Cost
Baseline	Aisle 1	0.00	0.00	142.80	4.13	142.80	4.13
Baseline	Aisle 2	0.00	0.00	167.02	4.84	167.02	4.84
Baseline	Aisle 3	0.00	0.00	215.49	6.24	215.49	6.24
Baseline	Aisle 4	0.00	0.00	145.17	4.20	145.17	4.20
Baseline	Boxing Table One	0.00	0.00	382.38	11.07	382.38	11.07
Baseline	Boxing Table Three	0.00	0.00	391.88	11.34	391.88	11.34
Baseline	Boxing Table Two	0.00	0.00	383.14	11.09	383.14	11.09

- **Name.** Nombre de la localización.
- **Operation Cost.** Costo del tiempo de operación.
- **% Operation Cost.** Porcentaje del costo de la localización sobre la suma de los costos de operación.
- **Resource Cost.** Costo por el uso de un recurso operando dentro de la localización.
- **% Resource Cost.** Porcentaje del costo de recurso en la localización sobre la suma de los costos del resto de los recursos.
- **Total Cost.** Costo de operación más costo del recurso.
- **% Total Cost.** Porcentaje del costo total de la localización sobre la suma de los costos totales de las localizaciones.

Location States By Percentage (Multiple Capacity)

Location States (Multi Cap) □ ×						
Scenario	Name	Scheduled Time (Hr)	% Empty	% Part Occupied	% Full	% Down
Baseline	Aisle 1	59.62	77.92	22.00	0.09	0.00
Baseline	Aisle 2	59.62	74.79	25.00	0.21	0.00
Baseline	Aisle 3	59.62	68.53	31.13	0.34	0.00
Baseline	Aisle 4	59.62	77.39	22.52	0.09	0.00
Baseline	Counting Location	59.62	89.85	10.15	0.00	0.00
Baseline	Order Table	59.62	0.00	98.71	1.29	0.00

- **Name.** Nombre de la localización.
- **Scheduled Time.** Porcentaje de tiempo en operación.
- **% Empty.** Porcentaje de tiempo que la localización estuvo vacía.
- **% Partially Occupied.** Porcentaje de tiempo que la localización estuvo parcialmente llena.
- **% Full.** Porcentaje de tiempo que la localización estuvo llena.
- **% Down.** Porcentaje de tiempo que la localización estuvo no disponible por paros no programados.

Location States By Percentage (Single Capacity/Tank)

Location States (Single Cap) (Avg. Reps) □ ×							
Name	Scheduled Time (Hr)	% Operation	% Setup	% Idle	% Waiting	% Blocked	% Down
Fila	1,000.00	0.00	0.00	86.54	0.00	13.46	0.00
Oficina	1,000.00	0.00	0.00	100.00	0.00	0.00	0.00
Servidor	4,000.00	68.35	0.00	31.65	0.00	0.00	0.00
Servidor.1	1,000.00	80.92	0.00	19.08	0.00	0.00	0.00
Servidor.2	1,000.00	74.21	0.00	25.79	0.00	0.00	0.00
Servidor.3	1,000.00	65.08	0.00	34.92	0.00	0.00	0.00
Servidor.4	1,000.00	53.19	0.00	46.81	0.00	0.00	0.00

- **Name.** Nombre de la localización.
- **Scheduled Time.** Tiempo total programado.
- **% Operation.** Porcentaje de tiempo que la localización estuvo en operación.

- **% Setup.** Porcentaje de tiempo que la localización estuvo en preparación.
- **% Idle.** Porcentaje de tiempo que la localización estuvo ociosa por falta de entidades.
- **% Waiting.** Porcentaje de tiempo esperando por un recurso o por otra entidad para agruparse, unirse, etc.
- **% Blocked.** Porcentaje de tiempo que las entidades permanecieron bloqueadas en la localización.
- **% Down.** Porcentaje de tiempo por paros no programados.

Locations

Name	Scheduled Time (Hr)	Capacity	Total Entries	Average Time Per Entry (Min)	Average Contents	Maximum Contents	Current Contents	% Utilization
Fila	1,000.00	1.00	10,791.00	0.75	0.13	1.00	0.00	13.45
Oficina	1,000.00	1.00	60,001.00	0.00	0.00	1.00	0.00	0.00
Servidor	4,000.00	4.00	10,791.00	15.20	0.68	4.00	1.67	68.35
Servidor.1	1,000.00	1.00	3,197.33	15.18	0.81	1.00	0.67	80.92
Servidor.2	1,000.00	1.00	2,921.00	15.24	0.74	1.00	0.67	74.21
Servidor.3	1,000.00	1.00	2,570.67	15.19	0.65	1.00	0.33	65.09
Servidor.4	1,000.00	1.00	2,102.00	15.18	0.53	1.00	0.00	53.19

- **Name.** Nombre de la localización.
- **Scheduled Time.** Tiempo total programado de la localización.
- **Capacity.** Capacidad de la localización.
- **Total Entries.** Total de entidades que entraron a la localización.
- **Avg Time Per Entry.** Tiempo promedio de permanencia en la localización.
- **Avg Contents.** Número promedio de entidades en la localización.
- **Maximum Contents.** Número máximo de entidades en la localización en el transcurso de la simulación.
- **Current Contents.** Número de entidades en la localización al final de la simulación.
- **% Utilization.** Porcentaje de utilización.

Logs

Scenario	Name	Number Observations	Minimum Value	Maximum Value	Average Value
Baseline	Tiempo	391.00	276.53	6,260.60	3,286.73

- **Name.** Identificador del registro.
- **Numbers Observations.** Número de entidades que ejecutaron la instrucción LOG durante la simulación.

- **Minimum Value.** El mínimo valor de tiempo registrado por la instrucción LOG.
- **Maximum Value.** El máximo valor de tiempo registrado por la instrucción LOG.
- **Avg Value.** El valor promedio de tiempo registrado por la instrucción LOG.

Node Entries

Node Entries				
Scenario	Name	Node	Total Entries	Blocked Entries
Baseline	CellNet	N1	558.00	0.00
Baseline	CellNet	N2	582.00	0.00
Baseline	CellNet	N3	861.00	0.00
Baseline	CellNet	N4	838.00	0.00
Baseline	CellNet	N5	559.00	0.00
Baseline	CellNet	N6	1,418.00	0.00
Baseline	CellNet	N7	1,376.00	0.00

- **Name.** Nombre de la trayectoria.
- **Node.** Número del nodo.
- **Total Entries.** Total de entradas de un recurso a un nodo.
- **Blocked Entries.** Número de veces que el nodo estuvo bloqueado por otro recurso.

Resources Costing

Resource Costs								
Scenario	Name	Units	NonUse Cost	% NonUse Cost	Usage Cost	% Usage Cost	Total Cost	% Total Cost
Baseline	Boxer	3.00	743.28	46.03	687.62	15.82	1,430.91	24.00
Baseline	Boxer.1	1.00	219.08	13.57	257.89	5.93	476.97	8.00
Baseline	Boxer.2	1.00	256.38	15.88	220.59	5.07	476.97	8.00
Baseline	Boxer.3	1.00	267.82	16.59	209.15	4.81	476.97	8.00
Baseline	Checker	1.00	54.23	3.36	541.97	12.47	596.19	10.00
Baseline	Filler	5.00	4.04	0.25	2,976.57	68.48	2,980.61	50.00
Baseline	Receiving	2.00	813.17	50.36	140.77	3.24	953.94	16.00

- **Name.** Nombre del recurso.
- **Units.** Número de recursos.
- **NonUse Cost.** Costo atribuible al ocio del recurso.
- **% NonUse Cost.** Porcentaje del costo de ocio del recurso sobre la suma de los costos de ocio de los recursos.
- **Usage Cost.** Costo por usar el recurso.
- **% Usage Cost.** Porcentaje del costo de utilización del recurso sobre la suma de los costos de utilización de los recursos.

- **Total Cost.** Costo de utilización más costo de ocio.
- **% Total Cost.** Porcentaje del costo total del recurso sobre la suma de los costos totales de los recursos.

Resource States By Percentage

Resource States							
Scenario	Name	Scheduled Time (Hr)	% In Use	% Travel To Use	% Travel To Park	% Idle	% Down
Baseline	CellOp	100.00	73.62	4.39	0.00	5.32	16.67

- **Name.** Nombre del recurso.
- **Scheduled Time.** Total de tiempo que el recurso fue programado para estar disponible.
- **% In Use.** Porcentaje de tiempo que el recurso fue utilizado.
- **% Travel To Use.** Porcentaje de tiempo que el recurso fue utilizado en movimientos entre localizaciones.
- **% Travel To Park.** Porcentaje de tiempo que el recurso estuvo viajando a su nodo base.
- **% Idle.** Porcentaje de ocio del recurso.
- **% Down.** Porcentaje que el recurso estuvo no disponible a causa de paros no programados.

Resources

Resource Summary										
Scenario	Name	Units	Scheduled Time (Hr)	Work Time (Min)	Number Times Used	Average Time Per Usage (Min)	Average Time Travel To Use (Min)	Average Time Travel To Park (Min)	% Blocked In Travel	% Utilization
Baseline	Boxer	3.00	178.86	5,157.18	810.00	6.19	0.18	0.00	0.00	48.06
Baseline	Checker	1.00	59.62	3,251.91	813.00	3.85	0.15	0.00	0.00	90.90
Baseline	Filler	5.00	298.11	17,862.11	965.00	18.40	0.11	0.00	0.00	99.86
Baseline	Receiving	2.00	119.24	1,055.74	18.00	58.56	0.09	0.00	0.00	14.76
Baseline	Receiving.1	1.00	59.62	496.37	7.00	70.74	0.17	0.00	0.00	13.88
Baseline	Receiving.2	1.00	59.62	559.37	11.00	50.80	0.05	0.00	0.00	15.64

- **Name.** Nombre del recurso.
- **Units.** Número de recursos.
- **Scheduled Time.** Tiempo programado para utilizar el recurso.
- **Work Time.** Tiempo de trabajo efectivo del recurso
- **Number of Times Used.** Número de ocasiones que se utilizó el recurso.
- **Avg Time Per Usage.** Tiempo promedio de utilización del recurso.
- **Avg Time Travel To Use.** Tiempo promedio por viaje del recurso.
- **Avg Time Travel To Park.** Tiempo promedio para dirigirse al nodo base.

- **% Blocked In Travel.** Porcentaje de tiempo que el recurso estuvo bloqueado al final del viaje.
- **% Utilization.** Porcentaje de utilización del recurso.

Variables

Scenario	Name	Total Changes	Average Time Per Change (Min)	Minimum Value	Maximum Value	Current Value	Average Value
Baseline	Average Time In System	404.00	8.84	0.00	2,297.00	2,292.00	1,154.74
Baseline	Inventory Aisle 1	409.00	8.74	-880.00	3,600.00	-880.00	1,387.03
Baseline	Inventory Aisle 2	481.00	7.43	-1,977.00	3,600.00	-1,977.00	815.22
Baseline	Inventory Aisle 3	623.00	5.73	-3,457.00	3,600.00	-3,457.00	134.24
Baseline	Inventory Aisle 4	417.00	8.56	-1,177.00	3,600.00	-1,177.00	1,247.98
Baseline	Order Filled	1,114.00	3.21	0.00	1,114.00	1,114.00	563.42

- **Name.** Nombre de la variable.
- **Total Changes.** Total de cambios de valor de la variable.
- **Average time Per Change.** El tiempo promedio entre cambios de valor de la variable.
- **Minimum Value.** Valor mínimo de la variable en el transcurso de la simulación.
- **Maximum Value.** Valor máximo de la variable en el transcurso de la simulación.
- **Current Value.** Valor de la variable al finalizar la simulación.
- **Average Value.** Valor promedio de la variable a través del tiempo.

Scoreboard

Scenario	Name	Total Exits	Average Time In System (Min)	Average Time In Operation (Min)	Average Cost
Baseline	Bearing	392.00	1,225.60	0.00	0.00
Baseline	Blank	0.00	0.00	0.00	0.00
Baseline	Cog	391.00	90.73	55.70	16.45
Baseline	Pallet	93.00	128.85	1.00	0.00
Baseline	Reject	167.00	89.04	54.14	14.94

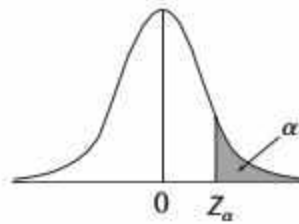
- **Name.** Nombre de la entidad.
- **Total Exits.** Total de entidades procesadas.
- **Average Time In System.** Tiempo de permanencia promedio de las entidades en el sistema.
- **Average Time In Operation.** Tiempo promedio que las entidades estuvieron siendo procesadas.
- **Average Cost.** Costo promedio por entidad.



Anexo 4

Distribuciones de probabilidad

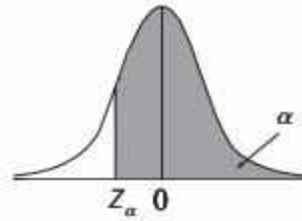
Probabilidades acumuladas para la distribución Normal Estándar



Z_α	0	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	Z_α
0.00	0.5000	0.5040	0.5080	0.5120	0.5160	0.5199	0.5239	0.5279	0.5319	0.5359	0.00
0.10	0.5398	0.5438	0.5478	0.5517	0.5557	0.5596	0.5636	0.5675	0.5714	0.5753	0.10
0.20	0.5793	0.5832	0.5871	0.5910	0.5948	0.5987	0.6026	0.6064	0.6103	0.6141	0.20
0.30	0.6179	0.6217	0.6255	0.6293	0.6331	0.6368	0.6406	0.6443	0.6480	0.6517	0.30
0.40	0.6554	0.6591	0.6628	0.6664	0.6700	0.6736	0.6772	0.6808	0.6844	0.6879	0.40
0.50	0.6915	0.6950	0.6985	0.7019	0.7054	0.7088	0.7123	0.7157	0.7190	0.7224	0.50
0.60	0.7257	0.7291	0.7324	0.7357	0.7389	0.7422	0.7454	0.7486	0.7517	0.7549	0.60
0.70	0.7580	0.7611	0.7642	0.7673	0.7704	0.7734	0.7764	0.7794	0.7823	0.7852	0.70
0.80	0.7881	0.7910	0.7939	0.7967	0.7995	0.8023	0.8051	0.8078	0.8106	0.8133	0.80
0.90	0.8159	0.8186	0.8212	0.8238	0.8264	0.8289	0.8315	0.8340	0.8365	0.8389	0.90
1.00	0.8413	0.8438	0.8461	0.8485	0.8508	0.8531	0.8554	0.8577	0.8599	0.8621	1.00
1.10	0.8643	0.8665	0.8686	0.8708	0.8729	0.8749	0.8770	0.8790	0.8810	0.8830	1.10
1.20	0.8849	0.8869	0.8888	0.8907	0.8925	0.8944	0.8962	0.8980	0.8997	0.9015	1.20
1.30	0.9032	0.9049	0.9066	0.9082	0.9099	0.9115	0.9131	0.9147	0.9162	0.9177	1.30
1.40	0.9192	0.9207	0.9222	0.9236	0.9251	0.9265	0.9279	0.9292	0.9306	0.9319	1.40
1.50	0.9332	0.9345	0.9357	0.9370	0.9382	0.9394	0.9406	0.9419	0.9429	0.9441	1.50
1.60	0.9452	0.9463	0.9474	0.9484	0.9495	0.9505	0.9515	0.9525	0.9535	0.9545	1.60
1.70	0.9554	0.9564	0.9573	0.9582	0.9591	0.9599	0.9608	0.9616	0.9625	0.9633	1.70
1.80	0.9641	0.9649	0.9656	0.9664	0.9671	0.9678	0.9686	0.9693	0.9699	0.9706	1.80
1.90	0.9713	0.9719	0.9726	0.9732	0.9738	0.9744	0.9750	0.9756	0.9761	0.9767	1.90
2.00	0.9772	0.9778	0.9783	0.9788	0.9793	0.9798	0.9803	0.9808	0.9812	0.9817	2.00
2.10	0.9821	0.9826	0.9830	0.9834	0.9838	0.9842	0.9846	0.9850	0.9854	0.9857	2.10
2.20	0.9861	0.9864	0.9868	0.9871	0.9875	0.9878	0.9881	0.9884	0.9887	0.9890	2.20
2.30	0.9893	0.9896	0.9898	0.9901	0.9904	0.9906	0.9909	0.9911	0.9913	0.9916	2.30
2.40	0.9918	0.9920	0.9922	0.9925	0.9927	0.9929	0.9931	0.9932	0.9934	0.9936	2.40
2.50	0.9938	0.9940	0.9941	0.9943	0.9945	0.9946	0.9948	0.9949	0.9951	0.9952	2.50
2.60	0.9953	0.9955	0.9956	0.9957	0.9959	0.9960	0.9961	0.9962	0.9963	0.9964	2.60
2.70	0.9965	0.9966	0.9967	0.9968	0.9969	0.9970	0.9971	0.9972	0.9973	0.9974	2.70
2.80	0.9974	0.9975	0.9976	0.9977	0.9977	0.9978	0.9979	0.9979	0.9980	0.9981	2.80
2.90	0.9981	0.9982	0.9982	0.9983	0.9984	0.9984	0.9985	0.9985	0.9986	0.9986	2.90
3.00	0.9987	0.9987	0.9987	0.9988	0.9988	0.9989	0.9989	0.9989	0.9990	0.9990	3.00
3.10	0.9990	0.9991	0.9991	0.9991	0.9992	0.9992	0.9992	0.9992	0.9993	0.9993	3.10
3.20	0.9993	0.9993	0.9994	0.9994	0.9994	0.9994	0.9994	0.9995	0.9995	0.9995	3.20
3.30	0.9995	0.9995	0.9995	0.9996	0.9996	0.9996	0.9996	0.9996	0.9996	0.9997	3.30
3.40	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9998	3.40
3.50	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	3.50
3.60	0.9998	0.9998	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	3.60
3.70	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	3.70
3.80	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	3.80
3.90	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	3.90
4.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	4.00

Fuente: Valores calculados con Excel.

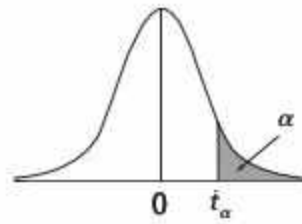
Probabilidades acumuladas para la distribución Normal Estándar



Z_α	-0.09	-0.08	-0.07	-0.06	-0.05	-0.04	-0.03	-0.02	-0.01	0	Z_α
-4.00	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	-4.00
-3.90	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	-3.90
-3.80	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	-3.80
-3.70	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	-3.70
-3.60	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9999	0.9998	0.9998	-3.60
-3.50	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	0.9998	-3.50
-3.40	0.9998	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	0.9997	-3.40
-3.30	0.9997	0.9996	0.9996	0.9996	0.9996	0.9996	0.9996	0.9995	0.9995	0.9995	-3.30
-3.20	0.9995	0.9995	0.9995	0.9994	0.9994	0.9994	0.9994	0.9994	0.9993	0.9993	-3.20
-3.10	0.9993	0.9993	0.9992	0.9992	0.9992	0.9992	0.9991	0.9991	0.9991	0.9990	-3.10
-3.00	0.9990	0.9993	0.9989	0.9989	0.9989	0.9988	0.9988	0.9987	0.9987	0.9997	-3.00
-2.90	0.9986	0.9986	0.9985	0.9985	0.9984	0.9984	0.9983	0.9982	0.9982	0.9981	-2.90
-2.80	0.9981	0.9980	0.9979	0.9979	0.9978	0.9977	0.9977	0.9976	0.9975	0.9974	-2.80
-2.70	0.9974	0.9973	0.9972	0.9971	0.9970	0.9969	0.9968	0.9967	0.9966	0.9965	-2.70
-2.60	0.9354	0.9963	0.9962	0.9951	0.9960	0.9959	0.9957	0.9956	0.9955	0.9953	-2.50
-2.50	0.9952	0.9951	0.9949	0.9948	0.9946	0.9945	0.9943	0.9941	0.9940	0.9938	-2.50
-2.40	0.9936	0.9934	0.9932	0.9931	0.9929	0.9927	0.9925	0.9922	0.9920	0.9918	-2.40
-2.30	0.9916	0.9913	0.9911	0.9909	0.9906	0.9904	0.9901	0.9898	0.9896	0.9893	-2.30
-2.20	0.9890	0.9887	0.9884	0.9881	0.9878	0.9875	0.9871	0.9858	0.9854	0.9851	-2.20
-2.10	0.9857	0.9854	0.9850	0.9846	0.9842	0.9838	0.9834	0.9830	0.9826	0.9821	-2.10
-2.00	0.9517	0.9812	0.9808	0.9803	0.9798	0.9793	0.9788	0.9783	0.9778	0.9772	-2.00
-1.90	0.9767	0.9761	0.9756	0.9750	0.9744	0.9738	0.9732	0.9726	0.9719	0.9713	-1.90
-1.80	0.9706	0.9599	0.9593	0.9686	0.9075	0.9571	0.9664	0.9656	0.9649	0.9541	-1.80
-1.70	0.9633	0.9625	0.9616	0.9608	0.9599	0.9591	0.9582	0.9573	0.9564	0.9554	-1.70
-1.60	0.9545	0.9535	0.9525	0.9515	0.9505	0.9495	0.9484	0.9474	0.9463	0.9452	-1.60
-1.50	0.9441	0.9429	0.9418	0.9406	0.9394	0.9382	0.9370	0.9357	0.9345	0.9332	-1.50
-1.40	0.9319	0.9300	0.9292	0.9279	0.9265	0.9251	0.9235	0.9222	0.9207	0.9192	-1.40
-1.30	0.9177	0.9162	0.9147	0.9131	0.9115	0.9099	0.9082	0.9066	0.9049	0.9032	-1.30
-1.20	0.9515	0.8997	0.8980	0.8962	0.8944	0.8925	0.8507	0.8888	0.8869	0.8849	-1.20
-1.10	0.8830	0.8810	0.8790	0.8770	0.8749	0.8729	0.8708	0.8686	0.8665	0.8643	-1.10
-1.00	0.8521	0.8599	0.8577	0.8554	0.8531	0.8508	0.8485	0.8461	0.8438	0.8413	-1.00
-0.90	0.8389	0.8365	0.8340	0.8315	0.8289	0.8264	0.8238	0.8212	0.8186	0.8159	-0.90
-0.80	0.8133	0.8106	0.8078	0.8051	0.8023	0.7995	0.7967	0.7939	0.7910	0.7881	-0.80
-0.70	0.7852	0.7823	0.7794	0.7764	0.7734	0.7704	0.7673	0.7642	0.7611	0.7580	-0.70
-0.60	0.7549	0.7517	0.7485	0.7454	0.7422	0.7389	0.7357	0.7324	0.7291	0.7257	-0.60
-0.50	0.7224	0.7190	0.7157	0.7123	0.7088	0.7054	0.7019	0.6985	0.6950	0.6915	-0.50
-0.40	0.6879	0.6844	0.6808	0.6772	0.6735	0.6700	0.6664	0.6628	0.6591	0.6554	-0.40
-0.30	0.6517	0.6480	0.6443	0.6406	0.6368	0.6331	0.6293	0.6255	0.6217	0.6179	-0.30
-0.20	0.6141	0.6103	0.6064	0.6026	0.5987	0.5948	0.5910	0.5871	0.5832	0.5793	-0.20
-0.10	0.5753	0.5714	0.5675	0.5636	0.5596	0.5557	0.5517	0.5478	0.5438	0.5398	-0.10
0.00	0.5359	0.5319	0.5279	0.5239	0.5199	0.5160	0.5120	0.5080	0.5040	0.5000	0.00

Fuente: Valores calculados con Excel.

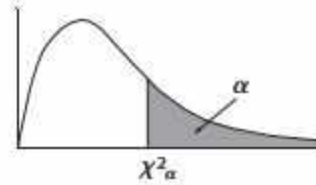
Valores críticos para la distribución t-student



v grados de libertad	t_α					
	$t_{0.10}$	$t_{0.05}$	$t_{0.025}$	$t_{0.01}$	$t_{0.005}$	$t_{0.001}$
1	3.078	6.314	12.706	31.821	63.657	318.309
2	1.886	2.920	4.303	6.965	9.925	22.327
3	1.638	2.353	3.182	4.541	5.841	10.215
4	1.533	2.132	2.776	3.747	4.604	7.173
5	1.476	2.015	2.571	3.365	4.032	5.893
6	1.440	1.943	2.447	3.143	3.707	5.208
7	1.415	1.895	2.365	2.998	3.499	4.785
8	1.397	1.860	2.306	2.896	3.355	4.501
9	1.383	1.833	2.262	2.821	3.250	4.297
10	1.372	1.812	2.228	2.764	3.169	4.144
11	1.363	1.796	2.201	2.718	3.106	4.025
12	1.356	1.782	2.179	2.681	3.055	3.930
13	1.350	1.771	2.160	2.650	3.012	3.852
14	1.345	1.761	2.145	2.624	2.977	3.787
15	1.341	1.753	2.131	2.602	2.947	3.733
16	1.337	1.746	2.120	2.583	2.921	3.686
17	1.333	1.740	2.110	2.567	2.898	3.646
18	1.330	1.734	2.101	2.552	2.878	3.610
19	1.328	1.729	2.093	2.539	2.861	3.579
20	1.325	1.725	2.086	2.528	2.845	3.552
21	1.323	1.721	2.080	2.518	2.831	3.527
22	1.321	1.717	2.074	2.508	2.819	3.505
23	1.319	1.714	2.069	2.500	2.807	3.485
24	1.318	1.711	2.064	2.492	2.797	3.467
25	1.316	1.708	2.060	2.485	2.787	3.450
26	1.315	1.706	2.056	2.479	2.779	3.435
27	1.314	1.703	2.052	2.473	2.771	3.421
28	1.313	1.701	2.048	2.467	2.763	3.408
29	1.311	1.699	2.045	2.462	2.756	3.396
30	1.310	1.697	2.042	2.457	2.750	3.385
40	1.303	1.684	2.021	2.423	2.704	3.307
50	1.299	1.676	2.009	2.403	2.678	3.261
60	1.296	1.671	2.000	2.390	2.660	3.232
70	1.294	1.667	1.994	2.381	2.648	3.211
80	1.292	1.664	1.990	2.374	2.639	3.195
90	1.291	1.662	1.987	2.368	2.632	3.183
100	1.290	1.660	1.984	2.364	2.626	3.174
∞	1.282	1.645	1.960	2.326	2.576	3.090

Fuente: Valores calculados con Excel.

Valores críticos para la distribución χ^2

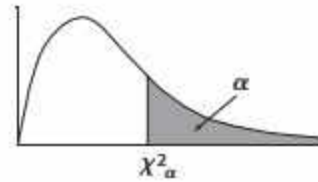


v grados de libertad	χ^2_{α}					
	$\chi^2_{0.10}$	$\chi^2_{0.05}$	$\chi^2_{0.025}$	$\chi^2_{0.01}$	$\chi^2_{0.005}$	$\chi^2_{0.001}$
1	2.706	3.841	5.024	6.635	7.879	10.828
2	4.605	5.991	7.378	9.210	10.597	13.816
3	6.251	7.815	9.348	11.345	12.838	16.266
4	7.779	9.488	11.143	13.277	14.860	18.467
5	9.236	11.070	12.833	15.086	16.750	20.515
6	10.645	12.592	14.449	16.812	18.548	22.458
7	12.017	14.067	16.013	18.475	20.278	24.322
8	13.362	15.507	17.535	20.090	21.955	26.124
9	14.684	16.919	19.023	21.666	23.589	27.877
10	15.987	18.307	20.483	23.209	25.188	29.588
11	17.275	19.675	21.920	24.725	26.757	31.264
12	18.549	21.026	23.337	26.217	28.300	32.909
13	19.812	22.362	24.736	27.688	29.819	34.528
14	21.064	23.685	26.119	29.141	31.319	36.123
15	22.307	24.996	27.488	30.578	32.801	37.697
16	23.542	26.296	28.845	32.000	34.267	39.252
17	24.769	27.587	30.191	33.409	35.718	40.790
18	25.989	28.869	31.526	34.805	37.156	42.312
19	27.204	30.144	32.852	36.191	38.582	43.820
20	28.412	31.410	34.170	37.566	39.997	45.315
21	29.615	32.671	35.479	38.932	41.401	46.797
22	30.813	33.924	36.781	40.289	42.796	48.268
23	32.007	35.172	38.076	41.638	44.181	49.728
24	33.196	36.415	39.364	42.980	45.559	51.179
25	34.382	37.652	40.646	44.314	46.928	52.620
26	35.563	38.885	41.923	45.642	48.290	54.052
27	36.741	40.113	43.195	46.963	49.645	55.476
28	37.916	41.337	44.461	48.278	50.993	56.892
29	39.087	42.557	45.722	49.588	52.336	58.301
30	40.256	43.773	46.979	50.892	53.672	59.703
40	51.805	55.758	59.342	63.691	66.766	73.402
50	63.167	67.505	71.420	76.154	79.490	86.661
60	74.397	79.082	83.298	88.379	91.952	99.607
70	85.527	90.531	95.023	100.425	104.215	112.317
80	96.578	101.879	106.629	112.329	116.321	124.839
90	107.565	113.145	118.136	124.116	128.299	137.208
100	118.498	124.342	129.561	135.807	140.169	149.449

Para $v > 100$ usar $\chi^2_{v,\alpha} = 0.5 \left(Z_{\alpha} + \sqrt{2v-1} \right)^2$

Fuente: Valores calculados con Excel.

Valores críticos para la distribución χ^2 (continuación)



v grados de libertad	χ^2_{α}					
	$\chi^2_{0.90}$	$\chi^2_{0.95}$	$\chi^2_{0.975}$	$\chi^2_{0.99}$	$\chi^2_{0.995}$	$\chi^2_{0.999}$
1	0.016	0.004	0.001	0.000	0.000	0.000
2	0.211	0.103	0.051	0.020	0.010	0.002
3	0.584	0.352	0.216	0.115	0.072	0.024
4	1.064	0.711	0.484	0.297	0.207	0.091
5	1.610	1.145	0.831	0.554	0.412	0.210
6	2.204	1.635	1.237	0.872	0.676	0.381
7	2.833	2.167	1.690	1.239	0.989	0.598
8	3.490	2.733	2.180	1.646	1.344	0.857
9	4.168	3.325	2.700	2.088	1.735	1.152
10	4.865	3.940	3.247	2.558	2.156	1.479
11	5.578	4.575	3.816	3.053	2.603	1.834
12	6.304	5.226	4.404	3.571	3.074	2.214
13	7.042	5.892	5.009	4.107	3.565	2.617
14	7.790	6.571	5.629	4.660	4.075	3.041
15	8.547	7.261	6.262	5.229	4.601	3.483
16	9.312	7.962	6.908	5.812	5.142	3.942
17	10.085	8.672	7.564	6.408	5.697	4.416
18	10.865	9.390	8.231	7.015	6.265	4.905
19	11.651	10.117	8.907	7.633	6.844	5.407
20	12.443	10.851	9.591	8.260	7.434	5.921
21	13.240	11.591	10.283	8.897	8.034	6.447
22	14.041	12.338	10.982	9.542	8.643	6.983
23	14.848	13.091	11.689	10.196	9.260	7.529
24	15.659	13.848	12.401	10.856	9.886	8.085
25	16.473	14.611	13.120	11.524	10.520	8.649
26	17.292	15.379	13.844	12.198	11.160	9.222
27	18.114	16.151	14.573	12.879	11.808	9.803
28	18.939	16.928	15.308	13.565	12.461	10.391
29	19.768	17.708	16.047	14.256	13.121	10.986
30	20.599	18.493	16.791	14.953	13.787	11.588
40	29.051	26.509	24.433	22.164	20.707	17.916
50	37.689	34.764	32.357	29.707	27.991	24.674
60	46.459	43.188	40.482	37.485	35.534	31.738
70	55.329	51.739	48.758	45.442	43.275	39.036
80	64.278	60.391	57.153	53.540	51.172	46.520
90	73.291	69.126	65.647	61.754	59.196	54.155
100	82.358	77.929	74.222	70.065	67.328	61.918

Para $v > 100$ usar $\chi^2_{v,\alpha} = 0.5 \left(Z_{\alpha} + \sqrt{2v-1} \right)^2$

Fuente: Valores calculados con Excel.

Valores críticos de la distribución F con $\alpha = 0.05$

		Grados de libertad para el denominador														
		1	2	3	4	5	6	8	9	10	15	20	30	50	100	
Grados de libertad para el numerador	x	1	161.45	18.51	10.13	7.71	6.61	5.99	5.32	5.12	4.96	4.54	4.35	4.17	4.03	3.94
	2	199.50	19.00	9.55	6.94	5.79	5.14	4.46	4.26	4.10	3.68	3.49	3.32	3.18	3.09	
	3	215.71	19.16	9.28	6.59	5.41	4.76	4.07	3.86	3.71	3.29	3.10	2.92	2.79	2.70	
	4	224.58	19.25	9.12	6.39	5.19	4.53	3.84	3.63	3.48	3.06	2.87	2.69	2.56	2.46	
	5	230.16	19.30	9.01	6.26	5.05	4.39	3.69	3.48	3.33	2.90	2.71	2.53	2.40	2.31	
	6	233.99	19.33	8.94	6.16	4.95	4.28	3.58	3.37	3.22	2.79	2.60	2.42	2.29	2.19	
	8	238.88	19.37	8.85	6.04	4.82	4.15	3.44	3.23	3.07	2.64	2.45	2.27	2.13	2.03	
	9	240.54	19.38	8.81	6.00	4.77	4.10	3.39	3.18	3.02	2.59	2.39	2.21	2.07	1.97	
	10	241.88	19.40	8.79	5.96	4.74	4.06	3.35	3.14	2.98	2.54	2.35	2.16	2.03	1.93	
	15	245.95	19.43	8.70	5.86	4.62	3.94	3.22	3.01	2.85	2.40	2.20	2.01	1.87	1.77	
	20	245.95	19.45	8.66	5.80	4.56	3.87	3.15	2.94	2.77	2.33	2.12	1.93	1.78	1.68	
	30	250.10	19.46	8.62	5.75	4.50	3.81	3.08	2.86	2.70	2.25	2.04	1.84	1.69	1.57	
	50	251.77	19.48	8.58	5.70	4.44	3.75	3.02	2.80	2.64	2.18	1.97	1.76	1.60	1.48	
	100	253.04	19.49	8.55	5.66	4.41	3.71	2.97	2.76	2.59	2.12	1.91	1.70	1.52	1.39	

Fuente: Valores calculados con Excel.

Valores críticos de la distribución F con $\alpha = 0.1$

		Grados de libertad para el denominador														
		1	2	3	4	5	6	8	9	10	15	20	30	50	100	
Grados de libertad para el numerador	x	1	39.86	8.53	5.54	4.54	4.06	3.78	3.46	3.36	3.29	3.07	2.97	2.88	2.81	2.76
	2	49.50	9.00	5.46	4.32	3.78	3.46	3.11	3.01	2.92	2.70	2.59	2.49	2.41	2.36	
	3	53.59	9.16	5.39	4.19	3.62	3.29	2.92	2.81	2.73	2.49	2.38	2.28	2.20	2.14	
	4	55.83	9.24	5.34	4.11	3.52	3.18	2.81	2.69	2.61	2.36	2.25	2.14	2.06	2.00	
	5	57.24	9.29	5.31	4.05	3.45	3.11	2.73	2.61	2.52	2.27	2.16	2.05	1.97	1.91	
	6	58.20	9.33	5.28	4.01	3.40	3.05	2.67	2.55	2.46	2.21	2.09	1.98	1.90	1.83	
	8	59.44	9.37	5.25	3.95	3.34	2.98	2.59	2.47	2.38	2.12	2.00	1.88	1.80	1.73	
	9	59.86	9.38	5.24	3.94	3.32	2.96	2.56	2.44	2.35	2.09	1.96	1.85	1.76	1.69	
	10	60.19	9.39	5.23	3.92	3.30	2.94	2.54	2.42	2.32	2.06	1.94	1.82	1.73	1.66	
	15	61.22	9.42	5.20	3.87	3.24	2.87	2.46	2.34	2.24	1.97	1.84	1.72	1.63	1.56	
	20	61.74	9.44	5.18	3.84	3.21	2.84	2.42	2.30	2.20	1.92	1.79	1.67	1.57	1.49	
	30	62.26	9.46	5.17	3.82	3.17	2.80	2.38	2.25	2.16	1.87	1.74	1.61	1.50	1.42	
	50	62.69	9.47	5.15	3.80	3.15	2.77	2.35	2.22	2.12	1.83	1.69	1.55	1.44	1.35	
	100	63.01	9.48	5.14	3.78	3.13	2.75	2.32	2.19	2.09	1.79	1.65	1.51	1.39	1.29	

Fuente: Valores calculados con Excel.

Valores críticos de la prueba de Kolmogorov-Smirnov

ν grados de libertad	$D_{\alpha=0.1}$	$D_{\alpha=0.05}$	$D_{\alpha=0.01}$
1	0.950	0.975	0.995
2	0.776	0.842	0.929
3	0.642	0.708	0.828
4	0.564	0.624	0.733
5	0.510	0.565	0.669
6	0.470	0.521	0.618
7	0.438	0.486	0.577
8	0.411	0.457	0.543
9	0.388	0.432	0.514
10	0.368	0.410	0.490
11	0.352	0.391	0.468
12	0.338	0.375	0.450
13	0.325	0.361	0.433
14	0.314	0.349	0.418
15	0.304	0.338	0.404
16	0.295	0.328	0.392
17	0.286	0.318	0.381
18	0.278	0.309	0.371
19	0.272	0.301	0.363
20	0.264	0.294	0.356
25	0.250	0.270	0.320
30	0.220	0.240	0.290
35	0.210	0.230	0.270
40	0.193	0.215	0.258
45	0.182	0.203	0.243
Para valores mayores a 35	$\frac{1.22}{\sqrt{n}}$	$\frac{1.36}{\sqrt{n}}$	$\frac{1.63}{\sqrt{n}}$

Fuente: Massey, F. J. (1951). "The Kolmogorov-Smirnov Test for Goodness of Fit", The Journal of the American Statistical Association (número 46), pp 70.

Estadísticos de prueba y valores críticos para la prueba de Anderson-Darling.

Distribución	Estadístico de prueba ajustado	Valores críticos α			
		0.1	.05	.025	.001
Parámetros conocidos $n \geq 5$	A_n^2	1.933	2.492	3.070	3.857
Normal	$A_n^2 \left(1 + \frac{4}{n} - \frac{25}{n^2}\right)$	0.632	0.751	0.870	1.029
Exponencial	$A_n^2 \left(1 + \frac{3}{5n}\right)$	1.070	1.326	1.587	1.943
Weibull	$A_n^2 \left(1 + \frac{1}{5\sqrt{n}}\right)$	0.637	0.757	0.877	1.038
Log-logística	$A_n^2 \left(1 + \frac{1}{4\sqrt{n}}\right)$	0.563	0.660	0.769	0.906

Fuente: Law, A.M. y Kelton, W.D. (2000), "Simulation Modeling and Analysis", (3th. ed). pp. 368, McGraw Hill.

Valores de la función Gamma

x	$\Gamma(x)$	x	$\Gamma(x)$	x	$\Gamma(x)$	x	$\Gamma(x)$
1.00	1.0000000	1.25	0.9064025	1.50	0.8862269	1.75	0.9190625
1.01	0.9943259	1.26	0.9043971	1.51	0.8865917	1.76	0.9213749
1.02	0.9888442	1.27	0.9025031	1.52	0.8870388	1.77	0.9237631
1.03	0.9835500	1.28	0.9007185	1.53	0.8875676	1.78	0.9262273
1.04	0.9784382	1.29	0.8990416	1.54	0.8881777	1.79	0.9287675
1.05	0.9735043	1.30	0.8974707	1.55	0.8888683	1.80	0.9313838
1.06	0.9687436	1.31	0.8960042	1.56	0.8896392	1.81	0.9340763
1.07	0.9641520	1.32	0.8946405	1.57	0.8904897	1.82	0.9368451
1.08	0.9597253	1.33	0.8933781	1.58	0.8914196	1.83	0.9396904
1.09	0.9554595	1.34	0.8922155	1.59	0.8924282	1.84	0.9426124
1.10	0.9513508	1.35	0.8911514	1.60	0.8935153	1.85	0.9456112
1.11	0.9473955	1.36	0.8901845	1.61	0.8946806	1.86	0.9486870
1.12	0.9435902	1.37	0.8893135	1.62	0.8959237	1.87	0.9518402
1.13	0.9399314	1.38	0.8885371	1.63	0.8972442	1.88	0.9550709
1.14	0.9364161	1.39	0.8878543	1.64	0.8986420	1.89	0.9583793
1.15	0.9330409	1.40	0.8872638	1.65	0.9001168	1.90	0.9617658
1.16	0.9298031	1.41	0.8867647	1.66	0.9016684	1.91	0.9652307
1.17	0.9266996	1.42	0.8863558	1.67	0.9032965	1.92	0.9687743
1.18	0.9237278	1.43	0.8860362	1.68	0.9050010	1.93	0.9723969
1.19	0.9208850	1.44	0.8858051	1.69	0.9067818	1.94	0.9760989
1.20	0.9181687	1.45	0.8856614	1.70	0.9086387	1.95	0.9798807
1.21	0.9155765	1.46	0.8856043	1.71	0.9105717	1.96	0.9837425
1.22	0.9131059	1.47	0.8856331	1.72	0.9125806	1.97	0.9876850
1.23	0.9107549	1.48	0.8857470	1.73	0.9146654	1.98	0.9917084
1.24	0.9085211	1.49	0.8859451	1.74	0.9168260	1.99	0.9958133
						2.00	1.0000000

Valores calculados a partir de la ecuación de aproximación de Lanczos

Para valores de x no tabulados, usar la relación $\Gamma(x + 1) = x \Gamma(x)$

Para valores naturales de x, usar $\Gamma(x) = (x - 1)!$

Fuente: A Precision Approximation of the Gamma Function; J SIAM Numer. Anal. Ser. B, Vol1; pp 68-96; (1964).

Probabilidades acumuladas de la distribución Poisson

x	Media										
	0.01	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
0	0.9900	0.9512	0.9048	0.8187	0.7408	0.6703	0.6065	0.5488	0.4966	0.4493	0.4066
1	1.0000	0.9988	0.9953	0.9825	0.9631	0.9384	0.9098	0.8781	0.8442	0.8088	0.7725
2		1.0000	0.9998	0.9989	0.9964	0.9921	0.9856	0.9769	0.9659	0.9526	0.9371
3			1.0000	0.9999	0.9997	0.9992	0.9982	0.9966	0.9942	0.9909	0.9865
4				1.0000	1.0000	0.9999	0.9998	0.9996	0.9992	0.9986	0.9977
5						1.0000	1.0000	1.0000	0.9999	0.9998	0.9997
6									1.0000	1.0000	1.0000

x	Media										
	1	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	1.9	2
0	0.3679	0.3329	0.3012	0.2725	0.2466	0.2231	0.2019	0.1827	0.1653	0.1496	0.1353
1	0.7358	0.6990	0.6626	0.6268	0.5918	0.5578	0.5249	0.4932	0.4628	0.4337	0.4060
2	0.9197	0.9004	0.8795	0.8571	0.8335	0.8088	0.7834	0.7572	0.7306	0.7037	0.6767
3	0.9810	0.9743	0.9662	0.9569	0.9463	0.9344	0.9212	0.9068	0.8913	0.8747	0.8571
4	0.9963	0.9946	0.9923	0.9893	0.9857	0.9814	0.9763	0.9704	0.9636	0.9559	0.9473
5	0.9994	0.9990	0.9985	0.9978	0.9968	0.9955	0.9940	0.9920	0.9896	0.9868	0.9834
6	0.9999	0.9999	0.9997	0.9996	0.9994	0.9991	0.9987	0.9981	0.9974	0.9966	0.9955
7	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	0.999	0.999	0.999
8									1.000	1.000	1.000

x	Media										
	2.5	3	3.5	4	4.5	5	5.5	6	6.5	7	7.5
0	0.0821	0.0498	0.0302	0.0183	0.0111	0.0067	0.0041	0.0025	0.0015	0.0009	0.0006
1	0.2873	0.1991	0.1359	0.0916	0.0611	0.0404	0.0266	0.0174	0.0113	0.0073	0.0047
2	0.5438	0.4234	0.3208	0.2381	0.1736	0.1247	0.0884	0.0620	0.0430	0.0296	0.0203
3	0.7576	0.6472	0.5366	0.4335	0.3423	0.2650	0.2017	0.1512	0.1118	0.0818	0.0591
4	0.8912	0.8153	0.7254	0.6288	0.5321	0.4405	0.3575	0.2851	0.2237	0.1730	0.1321
5	0.9580	0.9161	0.8576	0.7851	0.7029	0.6160	0.5289	0.4457	0.3690	0.3007	0.2414
6	0.9858	0.9665	0.9347	0.8893	0.8311	0.7622	0.6860	0.6063	0.5265	0.4497	0.3782
7	0.996	0.988	0.973	0.949	0.913	0.867	0.809	0.744	0.673	0.599	0.525
8	0.999	0.996	0.990	0.979	0.960	0.932	0.894	0.847	0.792	0.729	0.662
9	1.000	0.999	0.997	0.992	0.983	0.968	0.946	0.916	0.877	0.830	0.776
10		1.000	0.999	0.997	0.993	0.986	0.975	0.957	0.933	0.901	0.862
11			1.000	0.999	0.998	0.995	0.989	0.980	0.966	0.947	0.921
12				1.000	0.999	0.998	0.996	0.991	0.984	0.973	0.957
13					1.000	0.999	0.998	0.996	0.993	0.987	0.978
14						1.000	0.999	0.999	0.997	0.994	0.990
15							1.000	0.999	0.999	0.998	0.995
16								1.000	1.000	0.999	0.998
17										1.000	0.999
18											1.000

Fuente: Valores calculados con Excel.

Probabilidades acumuladas de la distribución Poisson (continuación)

x	Media										
	8	9	10	11	12	13	14	15	16	18	20
0	0.0003	0.0001	0.0000	0.0000	0.0000						
1	0.0030	0.0012	0.0005	0.0002	0.0001	0.0000	0.0000				
2	0.0138	0.0062	0.0028	0.0012	0.0005	0.0002	0.0001	0.0000	0.0000		
3	0.0424	0.0212	0.0103	0.0049	0.0023	0.0011	0.0005	0.0002	0.0001	0.0000	
4	0.0996	0.0550	0.0293	0.0151	0.0076	0.0037	0.0018	0.0009	0.0004	0.0001	0.0000
5	0.1912	0.1157	0.0671	0.0375	0.0203	0.0107	0.0055	0.0028	0.0014	0.0003	0.0001
6	0.3134	0.2068	0.1301	0.0786	0.0458	0.0259	0.0142	0.0076	0.0040	0.0010	0.0003
7	0.453	0.324	0.220	0.143	0.090	0.054	0.032	0.018	0.010	0.003	0.001
8	0.5925	0.4557	0.3328	0.2320	0.1550	0.0998	0.0621	0.0374	0.0220	0.0071	0.0021
9	0.7166	0.5874	0.4579	0.3405	0.2424	0.1658	0.1094	0.0699	0.0433	0.0154	0.0050
10	0.8159	0.7060	0.5830	0.4599	0.3472	0.2517	0.1757	0.1185	0.0774	0.0304	0.0108
11	0.8881	0.8030	0.6968	0.5793	0.4616	0.3532	0.2600	0.1848	0.1270	0.0549	0.0214
12	0.9362	0.8758	0.7916	0.6887	0.5760	0.4631	0.3585	0.2676	0.1931	0.0917	0.0390
13	0.9658	0.9261	0.8645	0.7813	0.6815	0.5730	0.4644	0.3632	0.2745	0.1426	0.0661
14	0.9827	0.9585	0.9165	0.8540	0.7720	0.6751	0.5704	0.4657	0.3675	0.2081	0.1049
15	0.9918	0.9780	0.9513	0.9074	0.8444	0.7636	0.6694	0.5681	0.4667	0.2867	0.1565
16	0.9963	0.9889	0.9730	0.9441	0.8987	0.8355	0.7559	0.6641	0.5660	0.3751	0.2211
17	0.9984	0.9947	0.9857	0.9678	0.9370	0.8905	0.8272	0.7489	0.6593	0.4686	0.2970
18	0.9993	0.9976	0.9928	0.9823	0.9626	0.9302	0.8826	0.8195	0.7423	0.5622	0.3814
19	0.9997	0.9989	0.9965	0.9907	0.9787	0.9573	0.9235	0.8752	0.8122	0.6509	0.4703
20	0.9999	0.9996	0.9984	0.9953	0.9884	0.9750	0.9521	0.9170	0.8682	0.7307	0.5591
21	1.0000	0.9998	0.9993	0.9977	0.9939	0.9859	0.9712	0.9469	0.9108	0.7991	0.6437
22		0.9999	0.9997	0.9990	0.9970	0.9924	0.9833	0.9673	0.9418	0.8551	0.7206
23		1.0000	0.9999	0.9995	0.9985	0.9960	0.9907	0.9805	0.9633	0.8989	0.7875
24			1.0000	0.9998	0.9993	0.9980	0.9950	0.9888	0.9777	0.9317	0.8432
25				0.9999	0.9997	0.9990	0.9974	0.9938	0.9869	0.9554	0.8878
26				1.0000	0.9999	0.9995	0.9987	0.9967	0.9925	0.9718	0.9221
27					0.9999	0.9998	0.9994	0.9983	0.9959	0.9827	0.9475
28					1.0000	0.9999	0.9997	0.9991	0.9978	0.9897	0.9657
29						1.0000	0.9999	0.9996	0.9989	0.9941	0.9782
30							0.9999	0.9998	0.9994	0.9967	0.9865
31							1.0000	0.9999	0.9997	0.9982	0.9919
32								1.0000	0.9999	0.9990	0.9953
33									0.9999	0.9995	0.9973
34									1.0000	0.9998	0.9985
35										0.9999	0.9992
36										0.9999	0.9996
37										1.0000	0.9998
38											0.9999
39											0.9999
40											1.0000

Fuente: Valores calculados con Excel.



Índice

A

Algoritmo

- congruencial aditivo, 30
- congruencial multiplicativo, 29
- de cuadrados medios, 24
- de multiplicador constante, 26
- de productos medios, 25
- lineal, 27

Algoritmos congruenciales no lineales, 31

- cuadrático, 31
- de Blum, Blum y Shub, 32

Análisis de sensibilidad, 15

Arribos cíclicos, 186

Atributo, 6

C

Corrida, tamaño insuficiente de la, 10

D

Distribución de conjunto de datos, determinación, 62

Distribuciones de probabilidad (Anexo 1), 296-309, 331

- de Erlang, 296
- de Weibull, 304
- distribuciones continuas, 296
- exponencial, 297
- gamma, 298
- log.logística, 229
- lognormal, 300
- normal, 301
- triangular, 302
- uniforme, 303

Distribuciones de probabilidad (Anexo 4), 331-342

- estadísticos de prueba y valores críticos
 - para la prueba de Anderson-Darling, 339
- probabilidades acumuladas para la distribución Normal Estándar, 333
- valores críticos de la prueba de Kolmorov-Smirnov, 338
- valores de la función gamma, 340
- Distribuciones discretas, 305
 - binomial, 306
 - de Bernoulli, 305
 - de Poisson, 308
 - geométrica, 307
 - uniforme discreta, 309
- E**
- Ensamblajes, acumulación y agrupamiento de piezas, 225
- Entidad, 4, 7
- Erlang, distribución de, 129
- Estado,
 - estable, 8, 9
 - transitorio, 8
- Evento(s), 4, 8
 - actuales, 4
 - futuros, 4
- F**
- Función de densidad de probabilidad, 61
- G**
- Generación de variables aleatorias,
 - distribución binomial, 88
 - distribución de Bernoulli, 82
 - distribución de Erlang, 86
 - distribución exponencial, 81
 - distribución normal, 87
 - distribución triangular, 89
 - distribución uniforme, 80
 - método de composición, 89
 - método de convolución, 86
 - método de la transformada inversa, 78
 - método de transformación directa, 92
- I**
- Instrucciones de control, 252
- Instrucciones de ProModel, 201-294
 - funciones probabilísticas, biblioteca, 202
- L**
- Localizaciones, 6
- M**
- Método de la transformada inversa, 78
- Modelo de un proceso de ensamble e inspección, 126
 - estado de la barra, 127
 - gráfica de estabilización, 128
 - longitud de la barra 1, 127
- Modelo de un sistema de inventarios, 130
 - demanda, 131
 - entrega de material, 131
 - inventario al final del día, 132
 - registros históricos, 147
 - residuo o módulo (MO), 131
- Modelo(s) de simulación, 2, 121
 - continuos, 2
 - de una línea de espera con un servidor, 122
 - determinísticos, 3
 - dinámicos, 3
 - discretos, 3
 - estáticos, 3
 - generación del, 12
 - probabilísticos (estocásticos) 3
 - validación del, 13
 - verificación del, 13
 - verificación y validación del, 114
- Modelado de un sistema, 175
 - definición de entidades, 179
 - definición de localizaciones, 176
 - definición de llegadas, 179
 - esquematación inicial del modelo, 175
 - uso de la instrucción DISPLAY, 181
 - uso de la opción View Text, 179
- N**
- Números aleatorios
 - entre 0 y 1, media de, 32

- independencia de, 34
- varianza de, 33
- Números pseudoaleatorios, 21-58
 - entre 0 y 1, 32
 - generación de, 22-24
 - propiedades, 30
 - prueba Chi-cuadrada, 38
 - prueba de corridas arriba y abajo, 41
 - prueba de huecos, 49
 - prueba de Kolmogorov-Smirnov, 65
 - prueba de medias, 35
 - prueba de series, 47
 - prueba de varianza, 36
 - prueba póker, 44
 - pruebas de independencia, 41
 - pruebas de uniformidad, 38
 - pruebas estadísticas para, 34
- O**
- Optimización con Simrunner, 257
- P**
- Pallet, 233
- Paros en los equipos, 212
- Principios básicos de la simulación, 1-20
 - definición, 4
 - introducción, 2
- ProModel, 152
 - construcción de un modelo, 153
 - editor de turnos, 153
 - editor gráfico, 152
 - elementos básicos, 152
 - estructura de programación en, 153
 - inclusión de gráficos, 183
 - introducción al uso de, 152
 - mejoramiento visual del modelo, 166
 - modelo M/M de líneas de espera, 153
 - referencias y ayuda, 153
 - resultados, 152
 - Stat::Fit, 152
- R**
- Recursos, 6, 8, 207
 - estáticos, 207
 - dinámicos, 207
- instrucción GET, 208
- instrucción FREE, 208
- utilización de turnos, 216
- Reglas de ruteo, 220
- Reloj de la simulación, 6, 8
 - absoluto, 6
 - relativo, 6
- Reportes estadísticos en promodel (Anexo 2), 311-319, 322
 - entities costing, 313
 - entity activity, 312
 - entity states by percentage, 313
 - failed arrival, 314
 - general, 312
 - locations costing, 314
 - logs, 316
 - resources states by percentage, 328
 - Scoreboard, 329
 - variables, 327
- Reportes estadísticos en promodel (Anexo 3), 322-329
- S**
- Selección de lenguajes de simulación, 134
 - arrastrar y colocar, 134
 - plantillas, 135
- Simulación con ProModel, 151-199
- Simulación
 - con ProModel, 151
 - de eventos discretos, 3, 4
 - de variables aleatorias, 113
 - definición del tiempo de, 182
 - definición, 2
 - modelos, 2, 121
 - réplica (corrida), 8
 - ventas y desventajas, 9
- Simulación de variables aleatorias, 113-149
- Simulaciones terminales, 114
 - intervalos de confianza, 114
- Simulaciones no terminales (o de estado estable), 117
 - longitud de las réplicas, 117
- Sistema, 7
 - estado del, 4, 7

Stat::Fit, 74

T

Transporte entre estaciones, 241

 montacargas, 245

 ruta, 245

V

Valor crítico, 63

Valor estadístico de prueba, 63

Valores críticos para la distribución t-student,
 332

Variables aleatorias, 6, 8, 11, 59-112

 continuas, 61

 definición, 60

 fin de la inspección, 122

 generación de, 78

 inicio de la inspección, 122

 prueba de Anderson-Darling, 67

 prueba de Chi-cuadrada, 63

 prueba de Kolmogorov-Smirnov, 65

 tiempo entre llegadas, 122

 tipos de, 60

Simulación

y análisis de sistemas con

ProModel[®]

2^a edición



Esta edición de Simulación y análisis de sistemas con ProModel presenta los conceptos de la modelación de los procesos estocásticos, el análisis estadístico de la información y su relación con la simulación estocástica discreta, utilizando como herramienta ProModel, un programa muy poderoso para el análisis y fácil de usar.

Entre las mejoras más significativas de este texto se encuentran las siguientes:

- Nuevos conceptos de programación, específicamente aquellos que tienen que ver con arribos cíclicos.
- El uso de la programación de paros por turnos.
- El manejo de materiales usando grúas viajeras.
- Ejemplos de instrucciones de control del tipo IF-THEN-ELSE, WHILE-DO.

El manejo de la herramienta SimRunner que permite la optimización de los modelos desarrollados con ProModel.

Asimismo, se han agregado casos integradores y una gran cantidad de ejercicios para fortalecer el aprendizaje.

Finalmente, se incluye un apéndice donde se describen los resultados obtenidos en los reportes con la nueva versión Output Viewer 4.0 de ProModel.

CD-ROOM con ProModel

El libro incluye un CD-ROM con la versión estudiantil más reciente de ProModel, que cuenta con todas las funcionalidades de la versión profesional.

Para mayor información visite el sitio Web de este libro en:
www.pearsonespañol.com/garciadunna

Visítenos en
www.pearsonespañol.com

ISBN 978-607-32-1511-4



9 786073 215114