Optimiza 12

LIBRO 4

Líneas de espera y Simulación

Optimiza12
Libro 4

Líneas de Espera y Simulación

Ing. Alejandro Roberti
Ing. Gustavo Chijani – Ing. Verónica Esaín — Ing. Esteban Gidekel
2019

.

Diseño de portada: **aero@optimiza.org**

Versión digital disponible (e-book) **http://optimiza.org**

ÍNDICE

CAPÍTULO 16 — Modelos de líneas de espera. Colas.
Características de un sistema de colas 3
Población de clientes 5
Proceso de llegadas 5
Proceso de llegada determinístico 5
Proceso de llegada probabilístico 5
Proceso de colas 7
Proceso de servicio 8
Modelos de colas 9
Indicadores de rendimiento en sistemas de colas 10
Sistemas M/M//1: un canal, una línea con llegada exponencial y proceso de servicio exponencial 12
Obtención de los indicadores con WinQSB 14
Cambios en el número de clientes que demandan servicio 17
Sistemas M/M/c: canal múltiple, una línea con llegada exponencial y proceso de servicio exponencial. 18
Interpretación de indicadores 20
Análisis de costos 22
Reconocimiento del modelo 23
Costos asociados 24
Sistemas con población finita (M/M/c//K) 26
Sistemas con capacidad de espera limitada (M/M/c/L) 28
Otros modelos 32
Ejemplos: 32

Apéndice al Capítulo 16
Algunas funciones de distribución de probabilidad en hoja de cálculo aplicables al tema de colas 35
Uso de la función DISTR.EXP.N 35
Sintaxis 36
Uso de la función POISSON.DISTR 36
Sintaxis 36

Capítulo 17. — Simulación

Simulación 40
Sistemas caóticos.. 42
Efecto Mariposa.. 43
Simulación en computadora.. 44
Simular un dado.. 47
Como simular la "raspadita"... 49
Simulando "la raspadita" en hoja de cálculo 49
Uso de Visual Basic para construir macros....................................... 52
Segundo ejemplo.. 59
Nota sobre las funciones en hoja de cálculo.................................... 66
Metodología de la simulación 67
Clasificación del sistema.. 68
Identificación de componentes generales de un sistema de simulación.... 69
Generación de números aleatorios... 71
Generación de números aleatorios con distribución uniforme............ 71
Generación de una serie continua distinta de la uniforme.................. 72
Simulando colas con WinQSB.. 62

Capítulo 18 — Estudios de Casos: a) Inventarios. b) proyección de Rentabilidad

Caso a) Inventarios 82
Política de inventario.. 82
Condiciones de simulación.. 82
Día 1 84
Día 2 85
Día 3 85
Días subsiguientes 86
Solución alternativa... 86
Caso b) Rentabilidad proyectada.. 90
Desarrollo y búsqueda de datos... 91
Generación de valores aleatorios en cada una de las variables............ 92
Hoja 1, datos... 93
Hoja 2 93

El Ruido de un Trueno.. 101

Presentación

El cuarto libro de la serie “***Optimiza 12***” cubre la teoría de colas o filas de espera y los principios básicos indispensables para la construcción de simuladores, no solo aplicados a líneas de espera sino a otros aspectos prácticos de la ingeniería.

Ambos temas se abordan con la perspectiva de su inserción general en Investigación Operativa aprovechando sus métodos y herramientas, por lo cual, los objetivos que se tuvieron en cuenta en el momento del diseño e inclusión en la obra fueron brindados por la experiencia de los autores a lo largo de varios años de trabajo con estudiantes de Ingenierías, ya sea en Alimentos, Industrial e Informática.

Esos objetivos contemplan la posibilidad de otorgar al futuro profesional herramientas básicas en estos temas, pero siempre con una mirada que le permita comprender que existen medios informáticos accesibles y relativamente sencillos que le ayudarán a agilizar cualquier gestión en las materias. Se propone, incluso, el diseño propio o la adecuación personal de esas herramientas. Creemos que este enfoque no solo es novedoso, sino que conlleva un potencial de inmediata aplicación profesional en ingeniería.

Si bien este texto es de aplicación general en ingenierías, el tema particular de simulación es demasiado superficial para los estudiantes de informática o de computación. Para ellos, entonces, existe un libro específico donde la teoría y práctica se encuentran adecuadamente profundizadas.

Septiembre, 2019

CAPÍTULO 16. MODELOS DE LÍNEAS DE ESPERA. COLAS

Los modelos de líneas de espera son capaces de describir las relaciones que se establecen durante la prestación de un ***servicio*** o ***proceso*** al que acceden ***clientes***. Los ***clientes*** son las entidades que deben recibir ese servicio o proceso. El servicio está instalado en una ***estación de trabajo***, o sitio específico donde hay un espacio (real o virtual) de atención (una boletería, un cajero, un equipo, etc.) al que se denomina ***canal de atención***. Ese servicio es prestado por un ***servidor***. Cuando el cliente es atendido abandona el sistema.

Para comenzar, veremos algunos ejemplos de sistemas cliente-servidor:

- Tomemos por caso un recinto donde funciona un banco: los clientes llegan al local, hacen fila, que puede ser una o más de una, frente a una ventanilla, o más de una. Puede haber una fila para dos ventanillas o cada ventanilla puede tener su propia fila. Cada ventanilla es un ***canal de atención*** y el cajero es el ***servidor***.

- En otros bancos, los clientes esperan sentados en cualquier orden, pero cada uno de ellos tiene una clave por la cual los llama el servidor en un orden preestablecido, no necesariamente el de llegada, sino que, además, por tipo de trámite o tipo de cliente.

- En algunos procesos productivos que se realizan en líneas en paralelo, ellas convergen a un solo lugar, donde las partes deben esperar para terminar su incorporación al producto final, como, por ejemplo, una estación de ensamble final de equipos o procesos de embalaje de varios componentes en un solo estuche.

- También tenemos las colas en las cajas de los supermercados, donde los clientes, luego de concluir con sus compras, eligen un canal de atención y esperan ser atendidos para finalmente salir del sistema o supermercado. En algunas cadenas ese sistema se reemplazó por el de cola única y múltiples cajas.

- El Administrador de Impresión de un sistema operativo es un caso de atención por colas de espera de los clientes (documentos a imprimir por orden de llegada o con ciertas convenciones de prioridad en algunos casos). Este Administrador irá conduciendo a los clientes cuando el servidor esté disponible (servidor "*on line*", servidor desocupado, etc.) y en el orden en que deben ser atendido. En este caso, los clientes son las peticiones de impresión que realizan ciertos programas o aplicaciones y el servidor es la impresora.

- Otro ejemplo: una red de computadoras donde una cantidad de ellas acceden a los recursos o a la administración que realiza una de ellas que cumple el rol de servidor, mediante peticiones de servicio que son atendidas por orden de llegada.

- Una planta fabril con máquinas herramienta que operan todas simultáneamente pero que requieren atención técnica aleatoria. El personal técnico las atiende por orden de llegada de los pedidos.

- Una etiquetadora de botellas PET que sirve a una línea de envasado que llenó esas botellas con aceite de oliva. Las botellas hacen cola esperando por su etiqueta.

- Una cooperativa eléctrica del interior de la provincia que dispone de dos móviles para atender solicitudes de reparaciones que efectúan los clientes y en el orden en que éstos llaman.

Sea cual fuere el caso, estos sistemas requieren un conocimiento mínimo de su estructura y funcionalidad, ya que es muy alta la probabilidad de que sea necesario optimizar su funcionamiento y administrarlo correctamente. Por ejemplo, puede ser necesario saber si hay que agregar una nueva estación de trabajo o aún quitarla. Para lo cual debemos respondernos preguntas como estas:

¿Cuánto tiempo tiene que esperar un cliente promedio para ser atendido?
¿Cuánto tiempo demora la atención de un cliente?
¿Se puede predecir ese tiempo?
¿Cuántos clientes hay — en promedio — en la cola?
¿Cuál es el máximo que puede haber?
¿Es posible que todos puedan esperar?
¿Cuántos servidores o ventanillas brindarán un servicio aceptable?
¿Qué es un servicio aceptable en el contexto?
¿Conviene que los clientes hagan una sola fila — modelo banco —? ¿o es preferible que sean varias filas — modelo supermercado —?
¿Cuál es el espacio necesario para que los clientes puedan esperar? [1]

Características de un sistema de colas

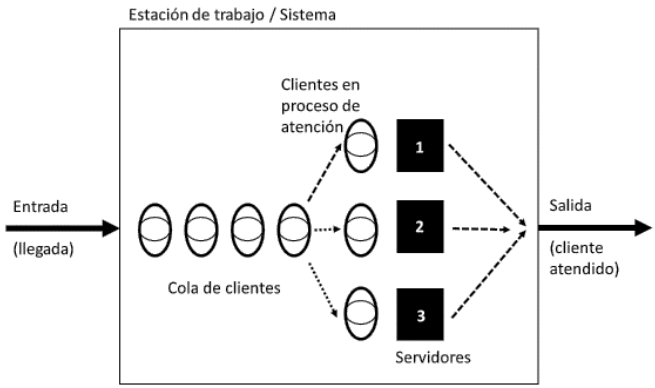
Los sistemas de colas se caracterizan por tener ciertos componentes definidos:

a. Los ***clientes***, que son los demandantes del servicio y, en el lapso en que esperan dicho servicio, conformarán la cola. Se denomina ***población*** al total de clientes que, potencialmente, pueden demandar el servicio ofrecido.
b. Una ***estación de trabajo***, que es el recinto físico o virtual donde se brinda el servicio demandado por los clientes. Es, en sí misma, un sistema.
c. Un ***proceso de llegada***, que es la forma, frecuencia, orden, etc. en que llegan o pueden llegar los clientes para ser servidos
d. Un ***proceso de cola***, que es:
    1) la manera en que esperan los clientes para ser atendidos y
    2) la ***disciplina o política de la cola*** que es la forma en que el cliente es elegido para ser atendido

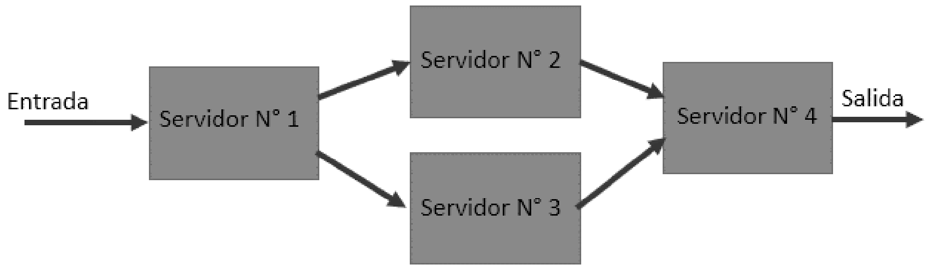
[1] Este espacio puede ser físico (sala de espera, hall de banco, playón de materiales y productos) o en otro soporte (buffer de memoria, capacidad de retención de líneas telefónicas rotativas, etc.)

e. Uno o más ***servidores***, que son los entes que realizan efectivamente el ser vicio que buscan los clientes
f. Un ***proceso de salida*** que es la forma, tiempo, orden, etc. en que se atienden o pueden ser atendidos los clientes (son servidos) luego de lo cual, cada uno de esos clientes, podrá
   1) abandonar el sistema inmediatamente después de ser atendido (cola de un paso) o
   2) pasar a otra cola (red de colas).

El siguiente esquema nos muestra un sistema de una fila, tres servidores y de un paso (luego de la atención, el cliente abandona el sistema)



En la figura de abajo vemos un diagrama de un sistema hipotético de cuatro servidores, donde el acceso al servidor 2 o 3 se realiza luego de que el cliente fue servido por el servidor 1. Al servidor 4 llegan clientes atendidos por los 2 y 3. Recién después de la atención por parte de 4, el cliente abandona el sistema.



Población de clientes

Es necesario conocer la ***población de clientes*** (o cantidad total de clientes que, en algún momento, pueden demandarnos el servicio). En muchos casos, es solamente una idea general del tamaño potencial de la base de clientes.

Un banco o un supermercado poseen una base grande de clientes, es una ***población infinita***. Este término se presta a confusiones, por lo que vamos a aclararlo: el número de clientes, como tal, es finito, pero la posibilidad de demanda de servicios no lo es: una vez satisfecho el servicio requerido por un cliente en particular, éste no se excluye del sistema y está en condiciones de volver a demandarlo. Como esto ocurre potencialmente con todos y cada uno de los clientes, hace que esa población finita se convierta en infinita. Un conjunto de 20 máquinas que operan 16 hs diarias pero que cada una de ellas tiene una probabilidad de atascarse en un tiempo medio de 1 hora de funcionamiento, independientemente de si se atascó antes o no, es una población infinita. Al contrario, si se instala una oficina de empadronamiento general en una ciudad de 3 millones de habitantes, cada "cliente", una vez empadronado, no volverá a demandar el servicio. Esta es una población finita.

Proceso de llegadas

Como en otros modelos ya estudiados, definimos estos procesos en función de lo que conocemos o estimamos de cuándo y cómo un cliente va a demandar el servicio que se ofrece. Así tendremos llegadas (clientes que solicitan servicio) ***determinísticas*** o ***estocásticas***.

Proceso de llegada determinístico

Este tipo de procesos se da cuando dos clientes sucesivos llegan en intervalos de tiempo idénticos y conocidos, por ejemplo, una etiquetadora a la que los clientes (botellas) llegan en intervalos iguales de tiempo (ciclo de tiempo)

Proceso de llegada probabilístico

En estos casos, el tiempo entre llegadas sucesivas es variable y se ajusta a una distribución de probabilidad. La determinación de la distribución de llegadas real es dificultosa, por lo cual se adopta una distribución de probabilidades conocida que se adapte razonablemente bien al tipo de eventos. Se utiliza, generalmente, la distribución exponencial. Así, la función de densidad para las llegadas de clientes, que se denomina $\boldsymbol{\lambda}$, (número promedio de llegadas por unidad de tiempo) está dada por

$$f(t) = \left(\frac{1}{\lambda}\right) e^{-\lambda T}$$

Para determinado tiempo *T*, se usa la función de densidad para calcular la probabilidad de que el siguiente cliente llegue dentro de las *T* unidades de tiempo a partir de la llegada del anterior:

$$P(tiempo\ entre\ llegadas \leq T) = 1 - e^{-\lambda T}$$

Veamos un **ejemplo** de este tipo de procesos de llegadas en un local bancario**:**

Los clientes llegan al banco a una tasa promedio diaria de 20 clientes cada hora, λ = 20 clientes/h. Si en este instante ingresa un cliente, ¿Cuál es la probabilidad de que llegue un cliente nuevo en los próximos 10 minutos?

El tiempo entre llegadas máximo, ***T***, es de 10 minutos (***T*** = 1/6 h), por tanto, calcularemos:

$$P(tiempo\ entre\ llegadas \leq 1/6\ h) = 1 - e^{-20\times(1/6)} = 0{,}964$$

Si lo hacemos en hoja de cálculo, emplearemos una de las siguientes funciones, según la versión que usemos, en base a la sintaxis general:

=***DISTR.EXP [.N](valor; lambda; acumulación*)**

que, llevada a la hoja, nos queda una de estas:

=DISTR.EXP(0,16667;20;VERDADERO) o bien =DISTR.EXP(0,16667;20;1)
=DISTR.EXP.N(0,16667;20;VERDADERO) o bien =DISTR.EXP.N(0,16667;20;1)

Como lo que buscamos es la probabilidad acumulada, debemos utilizar **VERDADERO** o “1” en el campo acumulación. (Ver Apéndice del Capítulo 16)

Otra manera de abordar este caso es refiriéndonos al número de clientes que llegan. Por ejemplo, queremos averiguar cuál es la probabilidad de que dos clientes lleguen dentro de los diez minutos siguientes. En este caso, al ser una variable discreta (*k* = 2 clientes), la distribución de probabilidad para el número de llegadas será una distribución de *Poisson*:

$$P(T,k) = \frac{e^{-\lambda T}(\lambda T)^k}{k!}$$

Siguiendo con el ejemplo anterior, para **λ** = 20 clientes/h y ***T*** = 1/6 h. La probabilidad de que lleguen dos clientes (***k*** = 2) en los próximos 10 minutos es:

$$P\left(T \leq \frac{1}{6}; k = 2\right) = \frac{e^{-20/6}(20/6)^2}{2 \times 1} = 0{,}20$$

En hoja de cálculo, la expresión de la función será, según la versión:

=DISTR.POISSON(A1;A2;ACU) (LibreOffice y OpenOffice)
=POISSON(A1;A2;ACU) (LibreOffice, OpenOffice, Excel)
=POISSON.DIST(A1;A2;ACU) (Excel)

en cualquiera de ellas, A1 representa el número de eventos (en este ejemplo, ***k*** = 2), y A2 el producto (λT = 20/6) En ACU se debe poner FALSO o "0" para indicar que no estamos buscando la probabilidad acumulada. (Ver Apéndice del Capítulo 16)

Si buscáramos la probabilidad acumulada la expresión que debemos utilizar es:

$$P\left(T \leq 1/6; k = 2\right) = \sum_{0}^{k} \frac{e^{-\lambda T}(\lambda T)^k}{k!}$$

o en Excel incluir el valor lógico "Verdadero", en Open/Libre Office el valor "1" en la función correspondiente de las presentadas más arriba.

Proceso de colas

Parte del proceso de colas tiene que ver con el tipo de política de espera: en algunos casos el cliente puede elegir entre una u otra fila pues hay más de una, en otros, no puede elegir y se incorporará a una fila para ser atendido por uno de varios servidores (ver la primera figura de este capítulo).

El número de clientes que está en la fila de espera (el largo de la cola) tiene importancia: en un banco el espacio se puede considerar infinito: no importa cuantos clientes estén esperando, siempre hay lugar para ellos, aunque sea en la vereda. En telefonía, un sistema de 0800 con líneas rotativas tiene capacidad finita de mantener llamadas en espera, lo que es notado por el que llama porque escucha una grabación diciendo que espere, si es que hay lugar en el "recinto" de espera o recibe tono de ocupado si no lo hay.

También forma parte del proceso la ***política de la cola***, que es el orden de atención respecto al orden de llegada. En este caso, hay varias posibilidades:

1. Primero en llegar, primero en salir (**FIFO** por sus siglas en inglés*, First Input-First Output*). Es, por ejemplo, lo que ocurre en un cajero del supermercado.
2. Último en llegar, primero en salir (**LIFO** por las siglas *Last Input-First Output*), es el caso de un depósito de bolsas de harina apiladas, donde las bolsas son los clientes que esperan una encima de la otra.
3. Selección aleatoria de servicio (**RSS** por sus siglas en inglés *Random service selection* o **SIRO** por *Service in random orden*). Se puede ejemplificar con un conjunto de secuencias de video (clientes) que esperan ser editadas en un disco rígido. Accederán al servidor en orden aleatorio.
4. Por prioridades. Caso de consultorio odontológico con una fila de pacientes (clientes) en espera de consulta ordinaria que se puede alterar por la demanda por urgencias con dolor o hemorragia.

Proceso de servicio

El proceso de servicio es la descripción de la manera — o política — en que son atendidos los clientes. Para describirlo disponemos de distintos enfoques:

Por las estaciones:
- Un supermercado con varias cajas es un sistema de ***cola múltiple***. Los servidores pueden ser idénticos (cada caja funciona a la misma velocidad) o no.
- Alternativamente un sistema de ***cola única*** (no confundir con servidor único, ya que se puede dar el caso de que una sola cola abastece a varios servidores que presten el mismo servicio). Suele ser el caso más común en ventanillas bancarias.

Por el número de clientes atendido en forma simultánea:
- La cola del supermercado, o de un banco, son sistemas donde se atiende a un cliente por vez.
- La cola para esperar el ómnibus es un sistema de atención por lotes.

Por la prioridad en la atención:
- El ingreso al servidor puede alterar la política de espera (FIFO o LIFO) ante la llegada de clientes con prioridades.

En cualquier caso, necesitaremos conocer el tiempo que demanda la atención del cliente. Este tiempo puede ser determinístico o probabilístico. En los probabilísticos es difícil conocer la distribución de probabilidad real. Generalmente se utiliza una *distribución exponencial*, cuya función de densidad depende de un parámetro (μ) y es:

$$s(t) = \left(\frac{1}{\mu}\right) e^{-\mu t}$$

donde μ es el número promedio de clientes atendido por unidad de tiempo y μ^{-1} será el tiempo medio de atención de un cliente.

Todo lo que hemos visto para el proceso de llegadas (en cuanto a funciones y métodos), se aplica en forma similar a estos procesos de servicio, solamente reemplazando λ por μ.

Modelos de colas

La simbología utilizada para diferenciar los diversos modelos utilizados es simplemente descriptiva de los parámetros más importantes. Se basa en un código con cinco símbolos: **X/Y/c/L/K**.

X: llegada/Y: servicio/c: canales/L: espacio/K: población

Sím-bolo	Valor	
Proceso de llegada		
X	**D**	Tiempo entre llegadas determinístico
	M	Tiempo entre llegadas probabilísticos con distribución exponencial
	G	Tiempo entre llegadas probabilísticos con otra distribución diferente a la exponencial (gral)
Proceso de Servicio		
Y	**D**	Tiempo de servicio determinístico
	M	Tiempo de servicio probabilísticos con distribución exponencial
	G	Tiempo de servicio probabilísticos con otra distribución diferente a la exponencial (general)
Proceso de Colas		
c	**1 a n**	Número de canales o servidores paralelos existentes (todos son idénticos en su velocidad)
Espacio de espera finito y/o población de clientes finita		
L	**1 a n**	Número total de clientes que pueden estar en el sistema en cualquier momento (haciendo cola más número de servidores)
K	**1 a n**	Número máximo de clientes de la población

Así, si decimos, por ejemplo, que un sistema es **M/M/2** significa que: ***tiene tiempos de llegadas probabilísticos con distribución exponencial, el tiempo de atención a un cliente también es probabilístico con distribución exponencial y tiene 2 servidores en paralelo, capacidad de espera no acotada (infinita) y sirve a una población infinita de clientes***.

Si agregáramos los demás elementos, por ejemplo, K=1 y L=10 sería **M/M/2/10/1,**

si **K=∞** y **L=∞** sería **M/M/2**,

si **K=∞** y **L=10** sería **M/M/2/10**

si **K=100** y **L=∞** sería **M/M/2//100**

si las llegadas fueran con distribución normal, las salidas con distribución exponencial, con 7 servidores capacidad de espera infinita y población infinita, sería **G/M/7**

Indicadores de rendimiento en sistemas de colas

Solamente haremos referencia a sistemas de una cola y en estado estacionario: No interesa la evolución de la cola del supermercado desde que llega el primer cliente de la mañana hasta una hora después ni desde que se cerraron las puertas de ingreso hasta que se despachó el último cliente, ya que esos son estados de transición.

Los siguientes son los indicadores más utilizados:

1. **Cola no explosiva:** se obtiene multiplicando el número total de servidores por la tasa de servicio ($c\mu \geq \lambda$) Si se cumple la desigualdad, define que el estado estacionario es posible. Si, en cambio, los arribos son mayores que las salidas se dice que el sistema es ***explosivo***.
2. **Tiempo promedio de espera, *Wq***: es el tiempo promedio que el cliente que llega tiene que esperar para ingresar al canal de atención.
3. **Tiempo promedio en el sistema, *W***: es el tiempo promedio que el cliente debe esperar desde que ingresa hasta que es servido, termina su atención y sale del sistema.
4. **Longitud media de la cola, *Lq***: es la cantidad promedio de clientes que están esperando en la cola para ser atendidos.
5. **Promedio de clientes en el sistema, *L*** es la cantidad promedio de clientes en la cola más los que están siendo atendidos.
6. **Probabilidad de bloqueo, *Pw***: es la probabilidad de que un cliente que llegue tenga que esperar a ser atendido. Es equivalente a ***Pb*: Probabilidad de que el servidor esté ocupado**

7. **Probabilidad de que el (los) servidor(es) estén libres *P0***: es la probabilidad de que el cliente que llega no deba esperar
8. **Fracción de tiempo de uso del servidor, *U***: es la probabilidad de que en un tiempo dado el servidor esté ocupado.
9. **Distribución de probabilidad de estado, *Pn***: es la probabilidad de que existan *n* clientes en el sistema, siendo *n* = 0, 1, 2, ... (obviamente, *P0* es un valor de *Pn*)
10. **Probabilidad de denegación de servicio, *Pd***: probabilidad de no atención por saturación de espacio finito de espera.

Todas estas definiciones deben conducir a relaciones vinculadas a los procesos de llegada, salida y servicio. Inclusive si no se conocen las distribuciones de probabilidades asociadas, simplemente conociendo λ y μ (promedios de llegadas por unidad de tiempo y de clientes atendidos por unidad de tiempo, respectivamente)[2].

Así, en sistemas con *K* y *L* infinitos, el tiempo total que un cliente permanece en el sistema será

Tiempo promedio en el sistema = tiempo promedio de espera + tiempo promedio de servicio

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

(si el promedio de atención — μ — es, por ejemplo, 4 clientes/h, el tiempo de atención será μ^{-1} = ¼ h/cliente)

También podemos calcular el número de clientes que hay en el sistema, en promedio,

Número promedio de clientes en el sistema =
número promedio de llegadas por unidad de tiempo x
tiempo promedio en el sistema

$$L = \lambda W$$

(Si para un cliente que acaba de llegar se espera que esté en el sistema ½ hora, en esa media hora llegarán otros clientes a una tasa, por ejemplo, λ = 6/hora. Cuando el cliente

---

[2] El método tradicional de evaluación de un sistema de colas comienza con un simple conteo de clientes que arriban por unidad de tiempo que puede ser realizado *in situ*.

se va, deja detrás un promedio de ½ x 6 = 3 clientes nuevos. En promedio, existen 3 clientes en el sistema en cualquier momento).

Y el número promedio de clientes en la cola

Número promedio de clientes en la cola =
número promedio de llegadas por unidad de tiempo x
tiempo promedio en la cola

$$Lq = \lambda \, Wq$$

Supongamos un sistema con λ = 6 clientes/h y μ = 4 clientes/h, con un número promedio de clientes en la cola, Lq = 2. Tendremos:

Wq =	Lq / λ =	2 / 6 =	1 / 3	Tiempo promedio en la cola, 1/3 h
W =	Wq + μ^{-1} =	1 / 3 + 1 / 4 =	7 / 12	Tiempo promedio en el sistema, 7/12 h
L =	λ W =	6 x 7 / 12 =	3,5	Número de clientes en el sistema, 3,5

Sistemas M/M//1: un canal, una línea con llegada exponencial y proceso de servicio exponencial

Las especificaciones para la aplicación del modelo M/M/1 son las siguientes:

- Población infinita
- Llegada con distribución de Poisson a tasa promedio λ
- Una sola cola, con capacidad infinita y política FIFO
- Proceso de servicio de un solo servidor con distribución exponencial con promedio μ clientes/u tiempo

Lo primero será comprobar si es posible el estado estacionario, que el sistema no sea explosivo. Para eso verificamos que la tasa de atención sea superior a la de arribos: $\mu > \lambda$.

En el siguiente ejemplo, encontraremos los indicadores de rendimiento del sistema.

Banco Simplex ha instalado un sistema de cinco cajeros automáticos en un centro comercial de 5 plantas que funciona 24 horas al día. Se desea comprender el

funcionamiento del conjunto, para lo cual nos piden analizar un solo cajero en las horas de mayor movimiento. Al comenzar nuestra tarea observamos durante varios días los movimientos del servicio y podemos estimar que a ese cajero arriba un promedio de sesenta clientes por hora, λ = 60 y que es capaz de atender en promedio μ = 66 clientes por hora. Como 66 > 60 podemos hacer un análisis de estado estacionario, por eso calcularemos los indicadores de rendimiento:

a. Factor de tráfico, o intensidad de tráfico, $\tau = \lambda / \mu$ = 60/66 = 0,9091

Como lo que analizamos es el número de clientes que llegan por cada cliente que es atendido, el factor de tráfico es, en realidad la *probabilidad de que esté ocupado el servidor*.

b. Probabilidad de que no haya clientes en el sistema (sistema no ocupado) (P0)

P0 = 1 – τ = 1 – 0,9091 = 0,0909

(el 9% de las veces el cliente encuentra el cajero vacío, el 91% de las veces está ocupado)

c. Largo promedio de la fila

$$L_q = \frac{\tau^2}{1-\tau} = \frac{0,9091^2}{1-0,9091} = 9,0909$$

El cajero tendrá nueve clientes haciendo cola, sin contar el que está siendo atendido.

d. Tiempo promedio de espera en la cola

$$W_q = \frac{L_q}{\lambda} = \frac{9,0909}{60} = 0,1515$$

el tiempo promedio para cada cliente que llega es de aproximadamente 9 minutos antes de poder pasar al cajero.

e. Tiempo de espera en el sistema

$$W = W_q + \frac{1}{\mu} = 0,1515 + \frac{1}{66} = 0,1667$$

en promedio, cada cliente invierte 10 minutos en el trámite: hacer la cola y ser atendido

f. Promedio de clientes en el sistema: los 9 que hacen cola más el que está siendo atendido

$L = \lambda W$ = 60 x 0,1667 = 10

g. Probabilidad de que un cliente tenga que esperar

$P_w = 1 - P_0 = \tau$ = 0,9091.

.

h. Probabilidad de que haya n clientes cuando un cliente llegue al sistema

$P_n = \tau^n P_0$

Podemos construir la siguiente tabla

n	P_n
0	**0.0909**
1	**0.0826**
2	**0.0751**
3	**0.0683**
...	**...**

Así, para saber, por ejemplo, cuál es la probabilidad de que no haya más de tres clientes en el sistema será:

$\Sigma P_i = P_0 + P_1 + P_2 + P_3$ = 0,0909 + 0,0826 + 0,0751 + 0,0683 = 0,3169

i. Utilización del cajero

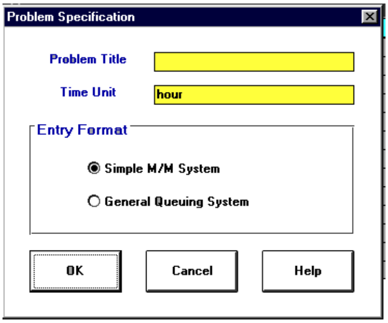
U = τ = 0,90901

El 91% del tiempo el cajero está en uso.

Obtención de los indicadores con WinQSB

Si ahora, con este mismo problema, utilizamos el software WinQSB, podremos, más adelante, introducir algunas modificaciones analíticas que nos permiten no solo comprender mejor el modelo sino también utilizar sus potencialidades de análisis de sensibilidad.

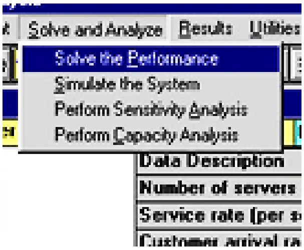
WinQSB presenta dos módulos relacionados con Colas: uno se denomina ***Análisis de Colas*** y el otro ***Simulación de Colas***. Veremos el primero, ***Análisis de Colas***, (*Queueing Analysis*):



Seleccionamos el módulo de colas "*Queueing Analysis*", y especificamos el modelo (en este caso, "sistema simple M/M") y las unidades de tiempo (recordemos que, como en todos los módulos de WinQSB, las unidades de tiempo sólo son un texto no operable que sirve como referencia a quien usa el programa). Luego procedemos a ingresar los datos. Los valores que no conocemos, simplemente no se completan. Las letras "M" en *Capacidad de la cola* y *Población de clientes,* indican *infinito*.

Data Description	ENTRY
Number of servers	1
Service rate (per server per hour)	66
Customer arrival rate (per hour)	60
Queue capacity (maximum waiting space)	M
Customer population	M
Busy server cost per hour	
Idle server cost per hour	
Customer waiting cost per hour	
Customer being served cost per hour	
Cost of customer being balked	
Unit queue capacity cost	

Luego procedemos a encontrar los rendimientos mediante el menú correspondiente y el informe que vemos en las figuras siguientes:



Performance Summary for QA Problem

06-06-2002	Performance Measure	Result
1	System: M/M/1	From Formula
2	Customer arrival rate (lambda) per hour =	60,0000
3	Service rate per server (mu) per hour =	66,0000
4	Overall system effective arrival rate per hour =	60,0000
5	Overall system effective service rate per hour =	60,0000
6	Overall system utilization =	90,9091 %
7	Average number of customers in the system (L) =	10,0000
8	Average number of customers in the queue (Lq) =	9,0909
9	Average number of customers in the queue for a busy system (Lb) =	10,0000
10	Average time customer spends in the system (W) =	0,1667 hours
11	Average time customer spends in the queue (Wq) =	0,1515 hours
12	Average time customer spends in the queue for a busy system (Wb) =	0,1667 hours
13	The probability that all servers are idle (Po) =	9,0909 %
14	The probability an arriving customer waits (Pw) or system is busy (Pb) =	90,9091 %
15	Average number of customers being balked per hour =	0
16	Total cost of busy server per hour =	$0

También podremos acceder a un informe de Distribución de probabilidades de n, mediante el menú "Resultados":

06-06-2002 14:41:55 n	Estimated Probability of n Customers in the System	Cumulative Probability
0	0,0909	0,0909
1	0,0826	0,1736
2	0,0751	0,2487
3	0,0683	0,3170
4	0,0621	0,3791
5	0,0564	0,4355
6	0,0513	0,4868
7	0,0467	0,5335
8	0,0424	0,5759
9	0,0386	0,6145
10	0,0350	0,6495
11	0,0319	0,6814
12	0,0290	0,7103
13	0,0263	0,7367
14	0,0239	0,7606
15	0,0218	0,7824
16	0,0198	0,8022
17	0,0180	0,8201
18	0,0164	0,8365
19	0,0149	0,8514
20	0,0135	0,8649
21	0,0123	0,8772
22	0,0112	0,8883
23	0,0102	0,8985
24	0,0092	0,9077
25	0,0084	0,9161

Cambios en el número de clientes que demandan servicio

Supongamos ahora que hemos detectado un aumento en la llegada promedio de clientes. En lugar de recibir un promedio de 60 por hora, llegan 70. Esto implica que deberíamos cambiar el equipo del cajero a un modelo que sea capaz de atender más de 70 clientes por hora, en lugar del que tenemos que atiente 66, para evitar que el sistema sea explosivo. Consultamos con el proveedor y vemos que hay uno de 73 clientes por hora. Supongamos que este aumento de demanda se originó por que al lado del cajero instalaron un cine. Creemos, además, que si la cola es muy larga se puede generar confusión, por eso tenemos que evaluar que probabilidad hay de que la cola sea de 16 o más personas (más una persona usando el cajero). La figura nos muestra la pantalla de carga con los datos del problema modificado con el nuevo modelo de cajero.

Data Description	EN1
Number of servers	1
Service rate (per server per hour)	73
Customer arrival rate (per hour)	70
Queue capacity (maximum waiting space)	M
Customer population	M
Busy server cost per hour	
Idle server cost per hour	
Customer waiting cost per hour	
Customer being served cost per hour	
Cost of customer being balked	
Unit queue capacity cost	

Obtendremos, entonces, los siguientes indicadores del problema modificado con $\lambda = 70$ y $\mu = 73$. Ahora hay 22 clientes en la cola y deben esperar 20 minutos (0,33 h).

rformance Summary for QA Problem

06-06-2002	Performance Measure	Result
1	System: M/M/1	From Formula
2	Customer arrival rate (lambda) per hour =	70,0000
3	Service rate per server (mu) per hour =	73,0000
4	Overall system effective arrival rate per hour =	70,0000
5	Overall system effective service rate per hour =	70,0000
6	Overall system utilization =	95,8904 %
7	Average number of customers in the system (L) =	23,3333
8	Average number of customers in the queue (Lq) =	22,3744
9	Average number of customers in the queue for a busy system (Lb) =	23,3333
10	Average time customer spends in the system (W) =	0,3333 hours
11	Average time customer spends in the queue (Wq) =	0,3196 hours
12	Average time customer spends in the queue for a busy system (Wb) =	0,3333 hours
13	The probability that all servers are idle (Po) =	4,1096 %
14	The probability an arriving customer waits (Pw) or system is busy (Pb) =	95,8904 %
15	Average number of customers being balked per hour =	0
16	Total cost of busy server per hour =	$0

Es evidente que, con estos indicadores, tanto los niveles de espera como los de cantidad de clientes haciendo cola son muy altos, por lo cual tendremos que pensar en otra alternativa, por ejemplo, poner dos cajeros, dos servidores. Para eso tendremos que cambiar el modelo M/M/1 que estábamos utilizando, que aplica a un solo servidor, por otro que pueda incorporar más servidores.

Sistemas M/M/c: canal múltiple, una línea con llegada exponencial y proceso de servicio exponencial.

Las especificaciones generales del sistema M/M/c son:

1. Población de clientes infinita
2. Proceso de llegadas que sigue una distribución de Poisson con tasa λ clientes/unidad de tiempo
3. Una cola sola de capacidad infinita, política FIFO.
4. Servicio compuesto por c servidores idénticos, cada uno de los cuales atiende con distribución exponencial a μ clientes/unidad de tiempo.
5. La tasa total de servicio promedio debe ser mayor que la tasa de arribos para que el sistema esté en estado estacionario, $c\,\mu > \lambda$

Vamos a utilizar el mismo ejemplo anterior, pero ahora suponiendo que, en vez de un solo cajero tendremos dos. Recordamos que buscamos tener una capacidad de atención superior a 70 cl/h. El catálogo nos daba que el modelo superior tenía una capacidad de 73 cl/h y habíamos determinado que con esa capacidad tendríamos colas numerosas. Como vemos que hay un modelo de 40 cl/h, vamos a probar con dos de ese modelo:

c = 2 servidores

λ = 70 clientes/hora

μ = 40 clientes/hora

1. Cálculo del factor de tráfico

$\boldsymbol{\tau = \lambda / \mu} = 1.75$

2. Probabilidad de que no se encuentre ningún cliente en el sistema (P_0)

$$P_0 = \frac{1}{\left(\sum_{n=0}^{c-1} \frac{\tau^n}{n!}\right) + \left(\frac{\tau^c}{c!}\right)\left(\frac{c}{c-\tau}\right)}$$

$$donde$$

$$\left(\sum_{n=0}^{c-1} \frac{\tau^n}{n!}\right) = \frac{\tau^0}{0!} + \frac{\tau^1}{1!} + \ldots + \frac{\tau^{c-1}}{(c-1)!}$$

por lo tanto, P_0 = 0,0667
(7% del tiempo la estación de cajeros estará vacía)

3. Número promedio de clientes en la cola (*Lq*)

$$L_q = \frac{\tau^{c+1}}{(c-1)!} \frac{1}{(c-\tau)^2} P_0 = 5{,}7167$$

4. Tiempo promedio de espera en la cola (*Wq*)
   Wq = *Lq* / λ = 0,081667

5. Tiempo promedio de espera en el sistema. (*W*)
   W = *Wq* + (1/μ) = 0,10667

6. Número de clientes promedio en el sistema (*L*)
   L = λ *W* = 7,4667

7. Probabilidad de que un cliente que llega tenga que esperar (*Pw*)

$$P_w = \frac{1}{c!} \tau^c \frac{c}{c-\tau} P_0 =$$

$$= \frac{1}{2!} \times 1{,}75^2 \times \frac{2}{2-1{,}75} \times 0{,}0667 = 0{,}81667$$

el 82% de las veces un cliente tendrá que esperar.

8. Probabilidad de que *n* clientes estén en el sistema (*Pn*)
   a) si $n \le c$, se aplica:

$$P_n = \frac{\tau^n}{n!} P_0$$

y obtenemos:

n	P_n
0	0,06667
1	0,11667
2	0,10210

b) si $n > c$, se aplica

$$P_n = \frac{\tau^n}{c!.c^{n-c}} P_0$$

y se obtiene

n	P_n
3	**0,08932**
4	**0,07816**
...	**...**

Ejemplo, ¿cuál es la probabilidad de que un solo cajero esté funcionando (o que un cajero esté vacío) ?, es lo mismo que preguntar si hay menos de dos clientes en el sistema: se contesta sumando

***P0 +P1* = 0,06667 + 0,11667 = 0,18334**

9. Grado de uso del sistema (*U*)

$$U = 1 - \left[P_0 + \left(\frac{c-1}{c} \right) P_1 + \left(\frac{c-2}{c} \right) P_2 + \ldots + \left(\frac{1}{c} \right) P_{c-1} \right] =$$

$$U = 1 - \left[P_0 + \left(\frac{1}{2} \right) P_1 \right] = 1 - \left[0{,}06667 + (0{,}5 \times 0{,}11667) \right] = 0{,}875$$

Interpretación de indicadores

Vamos a analizar los indicadores obtenidos aplicando WinQSB con los datos cargados como se muestra en la figura siguiente:

Data Description	ENTRY
Number of servers	2
Service rate (per server per hour)	40
Customer arrival rate (per hour)	70
Queue capacity (maximum waiting space)	M
Customer population	M
Busy server cost per hour	
Idle server cost per hour	
Customer waiting cost per hour	
Customer being served cost per hour	
Cost of customer being balked	
Unit queue capacity cost	

Los indicadores que obtendremos son los que vemos en la figura de abajo, recordando que las primeras tres líneas son el enunciado del problema (llegan 70 clientes por hora y la velocidad de atención de las dos cajas es de 40 clientes por hora).

06-06-2002	Performance Measure	Result
1	System: M/M/2	From Formula
2	Customer arrival rate (lambda) per hour =	70,0000
3	Service rate per server (mu) per hour =	40,0000
4	Overall system effective arrival rate per hour =	70,0000
5	Overall system effective service rate per hour =	70,0000
6	Overall system utilization =	87,5000 %
7	Average number of customers in the system (L) =	7,4667
8	Average number of customers in the queue (Lq) =	5,7167
9	Average number of customers in the queue for a busy system (Lb) =	7,0000
10	Average time customer spends in the system (W) =	0,1067 hours
11	Average time customer spends in the queue (Wq) =	0,0817 hours
12	Average time customer spends in the queue for a busy system (Wb) =	0,1000 hours
13	The probability that all servers are idle (Po) =	6,6667 %
14	The probability an arriving customer waits (Pw) or system is busy (Pb) =	81,6667 %
15	Average number of customers being balked per hour =	0
16	Total cost of busy server per hour =	$0

Podríamos preguntarnos qué probabilidad hay de que la cola sea demasiado larga. Digamos que en algún momento haya más de 17 clientes en el sistema. Para resolver esta cuestión deberemos hacer un análisis de probabilidad (*Probability Summary*), en el menú *Resultados* (*Results*), y obtendremos una tabla como la siguiente:

06-06-2002 14:55:21 n	Estimated Probability of n Customers in the System	Cumulative Probability
0	0,0667	0,0667
1	0,1167	0,1833
2	0,1021	0,2854
3	0,0893	0,3747
4	0,0782	0,4529
5	0,0684	0,5213
6	0,0598	0,5811
7	0,0524	0,6335
8	0,0458	0,6793
9	0,0401	0,7194
10	0,0351	0,7545
11	0,0307	0,7852
12	0,0269	0,8120
13	0,0235	0,8355
14	0,0206	0,8561
15	0,0180	0,8741
16	0,0157	0,8898
17	0,0138	0,9036
18	0,0121	0,9156
19	0,0105	0,9262
20	0,0092	0,9354
21	0,0081	0,9435
22	0,0071	0,9505

Si sumamos la probabilidad simple de cada valor de *n* por encima de 17, [P(18)=0,0121 + P(19) + … + P(∞) =0,0105 + …], obtendremos la probabilidad de que se encuentren más de 17 clientes. Eso, en la práctica, lo podremos hacer restando de 1 la probabilidad acumulada hasta 17:

$$1 - P(17) = 1 - 0{,}9036 = 0{,}096 = 9{,}6\%$$

Análisis de costos

Hasta ahora, en el ejemplo anterior, la decisión de agregar o no un cajero se basaba en consideraciones de servicio (tiempo de esperas y largos de la fila) que fueran aceptables. Agregaremos ahora la posibilidad de tomar decisiones a partir de análisis de los costos asociados.

Como ejemplo, supongamos que en una planta de envasado tenemos un número de máquinas ensachetadoras automáticas que se atascan con frecuencia y que son reparadas en el mismo orden en que reclaman el servicio (se sigue una política FIFO). Estas reparaciones las realizan una dotación de siete técnicos que van a las máquinas demandantes.

La frecuencia de roturas de equipos es tal que vemos que, en promedio, las máquinas atascadas son de 10 a 12 en cualquier momento, lo que nos da una tasa de roturas de 25 máquinas/h. Se reparan en tiempos del orden de los 15 minutos. Se sabe que aumentando el personal de reparaciones aumentaría la producción de la planta por disminución del número de máquinas detenidas, pero no se sabe a cuántas personas más habría que contratar.

Reconocimiento del modelo

El sistema se puede describir con un modelo de colas, siendo las máquinas atascadas los ***clientes*** y el total de las máquinas de la planta su ***población***. Como hay un gran número de máquinas y que se atascan independientemente de que previamente se hayan reparado o no vamos a suponer que la población es ***infinita***. Hay siete servidores idénticos que operan bajo la política **FIFO**. El modelo es el de una sola fila de clientes esperando a ser atendidos por el primer servidor disponible y lo parametrizamos de la siguiente manera:

1. La aparición de máquinas atascadas puede ser aproximada a un modelo de Poisson con una tasa promedio de 25 clientes/hora
2. Cada máquina es reparada en una cantidad aleatoria de tiempo aproximada a una distribución exponencial con un promedio de servicio de 15 minutos, lo que da una tasa de 4 clientes/hora para la atención.

Por tanto, en base a estos datos y supuestos, definimos con que modelo describiremos el caso. Será un sistema

M/M/7
$\boldsymbol{\lambda}$ = 25 clientes/hora
$\boldsymbol{\mu}$ = 4 clientes/hora
población infinita

Los indicadores de rendimiento que obtendremos con WinQSB son los siguientes:

06-08-2002	Performance Measure	Result
1	System: M/M/7	From Formula
2	Customer arrival rate (lambda) per hour =	25,0000
3	Service rate per server (mu) per hour =	4,0000
4	Overall system effective arrival rate per hour =	25,0000
5	Overall system effective service rate per hour =	25,0000
6	Overall system utilization =	89,2857 %
7	Average number of customers in the system (L) =	12,0973
8	Average number of customers in the queue (Lq) =	5,8473
9	Average number of customers in the queue for a busy system (Lb) =	8,3333
10	Average time customer spends in the system (W) =	0,4839 hours
11	Average time customer spends in the queue (Wq) =	0,2339 hours
12	Average time customer spends in the queue for a busy system (Wb) =	0,3333 hours
13	The probability that all servers are idle (Po) =	0,1017 %
14	The probability an arriving customer waits (Pw) or system is busy (Pb) =	70,1674 %

Vemos que hay 12 máquinas en espera y que la cada máquina no funciona por 0,48 hora (aprox. media hora). Podríamos ahora buscar sucesivos resultados para diferentes cantidades de servidores, digamos, usar WinQSB varias veces, agregando un técnico por vez, desde 8 hasta 11, y construir, con los informes de cada caso, una tabla como la que sigue.

NÚMERO DE SERVIDORES	7	8	9	10	11
Overall system utilization (%)	89.2857	78.1250	69.4444	62.5000	56.8182
Average number of customers in the queue (Lq)	5.8473	1.4936	0.5363	0.2094	0.0830
Average number of customers in the system (L)	12.0973	7.7436	6.7863	6.4594	6.3330
Probabilidad de espera (Pw)(Pb)	0.7017	0.4182	0.2360	0.1257	0.0630
Average time in the queue (Wq)	0.2339	0.0597	0.0215	0.0084	0.0033
Average time in the system (W)	0.4839	0.3097	0.2715	0.2584	0.2533

Evidentemente, a medida que agregamos servidores disminuyen los tiempos de espera y el número de máquinas fuera de servicio (con 11 servidores se logra algo así como 6 a 7 máquinas detenidas).

Costos asociados

Para trabajar en un análisis de costos vamos a identificar dos costos básicos: el costo de los servidores (costo de personal) y lo que cuesta que una máquina esté detenida esperando el servicio.

Costo total de personal/u.de tiempo = Costo/hora de cada servidor x número de servidores

Costo total de espera = Costo de cada máquina fuera de servicio x promedio de máquinas fuera de servicio

de esta manera podremos calcular un costo global del servicio, que será el que deberíamos optimizar:

Costo total = costo total de personal/u. de tiempo + costo total de espera

Para calcular estos costos, asignaremos, por ejemplo, un costo – una pérdida – de $100 por hora/máquina detenida. ¿Cómo hubiésemos llegado a este valor hipotético en la práctica? Por ejemplo, podría haber sido calculando el nivel de producción de la máquina en actividad y el valor horario de los bienes que ella produce. Al estar detenida ese valor no producido es el "costo" — pérdida marginal — del equipo detenido. Estos valores son difíciles de calcular, pueden incluir costos directos por ventas no realizadas[3]

[3] Un ejemplo de asignación de costo a la hora máquina detenida: tomar la producción horaria de esa máquina si funcionara y establecer el valor de la producción no realizada como costo de pérdida. Si la máquina produjera 1000 sachets por hora, una hora detenida significaría el valor de 1000 sachets.

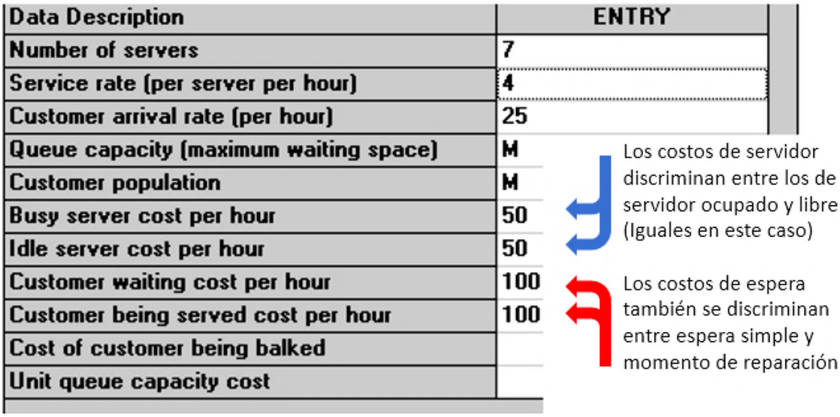
e indirectos como insatisfacción del cliente o cesión de mercado por falta de stock o multas por no cumplimiento de compromisos, etc.

Asignamos también, por ejemplo, a la mano de obra un costo de $50 por hora/hombre, y ahora podremos valorizar cada alternativa obtenidas en la tabla anterior.

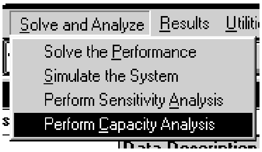
Servidores	Clientes en el sistema	Cálculo del costo/hora	Costo/hora
7	12,0973	50 x 7 + 100 x 12,0973	1559,73
8	7,7436	50 x 8 + 100 x 7,7436	1174,36
9	6,7863	50 x 9 + 100 x 6,7863	1128,23
10	6,4594	50 x 10 + 100 x 6,4594	1145,94
11	6,3330	50 x 11 + 100 x 6,3330	1183,30

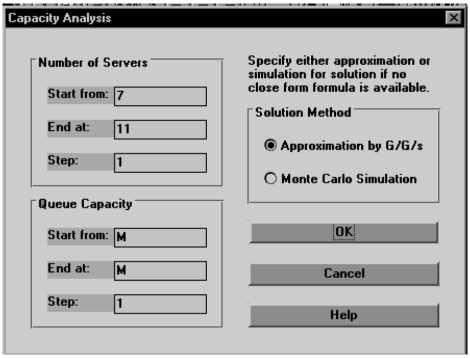
Como podemos observar, el punto de mínimo costo es utilizar 9 servidores (costo óptimo).

Podemos encontrar un resultado similar, sin pasos intermedios, utilizando WinQSB, según el procedimiento que ilustramos con las siguientes capturas de pantallas:



A la pantalla de carga de datos le agregamos los costos de espera y los de cada servidor, pero manteniendo el número original de servidores, siete. Observamos que el costo de cada servidor está dividido en ocupado y libre, pues puede ocurrir que sean diferentes. Los costos de espera también pueden ser diferentes entre espera simple y en atención (por ejemplo, en atención puede haber desperdicio de insumos).





En el menú *Solve and Analyze* seleccionamos *Análisis de capacidad*. En el cuadro de parámetros del análisis pondremos que comenzaremos con siete servidores (*start from*) y finalizaremos en once servidores con un paso de un servidor por vez. Vemos que podríamos variar la capacidad de la cola, aunque no lo haremos en este caso.

Number of Server	Queue Capacity	Total Cost	Busy Server Cost	Idle Server Cost	Waiting Customer Cost	Served Customer Cost
7	M	$1559,7280	312,5000	37,5000	584,7282	625,0001
8	M	$1174,3640	312,5000	87,5000	149,3638	625,0001
9	M	$1128,6260	312,5000	137,5000	53,6263	624,9999
10	M	$1145,9420	312,5000	187,5000	20,9422	625,0000
11	M	$1183,2950	312,5000	237,5000	8,2950	625,0000

El informe que obtenemos, figura de arriba, nos muestra el valor de cada uno de los sistemas en el rango que va de siete a once servidores. Solo nos resta seleccionar el óptimo, que en este caso es de nueve servidores.

Sistemas con población finita (M/M/c//K)

La característica principal de estos modelos es que, al ser una población limitada, es de esperar que la tasa de arribos de clientes disminuya en el tiempo. Si se supone que hay 100 máquinas ensachetadoras, y que, luego de reparadas, no se atascan más, es evidente que, a razón de 12 máquinas/h en reparación, el remanente se va achicando a medida que transcurren las horas.

Los extremos de variación de las tasas de arribo serán entonces desde un máximo (que tendremos cuando el sistema se inicia y aún no fue atendido ningún cliente), hasta cero (lo que ocurre cuando todos los clientes pasaron por el sistema, y por tanto no puede llegar ninguno más). El proceso de llegadas se tomará para cada cliente en particular: hay que determinar la tasa en que una máquina requiere ser atendida.

Supongamos que la frecuencia con que una máquina requiere atención es de una vez cada cuatro horas, se trabaja como si fuera una tasa. Por tanto, la tasa de arribos anterior (λ = 25/h) ahora la referimos a cada una de las máquinas por lo que tendremos

$$\lambda' = \frac{\lambda}{K} = \frac{25 \ {solicitudes}/{hora}}{100 \ máquinas} = 0{,}25 \ {solicitudes}/{h \ máquina}$$

Junto con los datos anteriores, que son:

μ = 4 máquinas/hora
costo por servidor = $50/hora servidor
costo de pérdida de producción = $100/hora máquina

ingresamos a WinQSB:

Data Description	ENTRY
Number of servers	7
Service rate (per server per hour)	4
Customer arrival rate (per hour)	.25
Queue capacity (maximum waiting space)	M
Customer population	100
Busy server cost per hour	50
Idle server cost per hour	50
Customer waiting cost per hour	100
Customer being served cost per hour	100
Cost of customer being balked	
Unit queue capacity cost	

Vemos que a los datos ya existentes hemos agregado: la población finita (100) (sombreado en la figura) y la tasa de arribos que ahora representa la tasa de requerimientos de servicio de cada máquina (*Customer Arrival rate*), (0,25, también sombreado). Con eso obtendremos el reporte siguiente, en el que vemos que el número de máquinas en espera pasó de 12,0973 a 7,5885.

06-23-2003	Performance Measure	Result
1	System: M/M/7//100	From Formula
2	Customer arrival rate (lambda) per hour =	0,2500
3	Service rate per server (mu) per hour =	4,0000
4	Overall system effective arrival rate per hour =	23,1029
5	Overall system effective service rate per hour =	23,1029
6	Overall system utilization =	82,5102 %
7	Average number of customers in the system (L) =	7,5885
8	Average number of customers in the queue (Lq) =	1,8128
9	Average number of customers in the queue for a busy system (Lb) =	3,4505
10	Average time customer spends in the system (W) =	0,3285 hours
11	Average time customer spends in the queue (Wq) =	0,0785 hours
12	Average time customer spends in the queue for a busy system (Wb) =	0,1494 hours
13	The probability that all servers are idle (Po) =	0,1766 %
14	The probability an arriving customer waits (Pw) or system is busy (Pb) =	52,5382 %
15	Average number of customers being balked per hour =	0
16	Total cost of busy server per hour =	$288,7859
17	Total cost of idle server per hour =	$61,2141
18	Total cost of customer waiting per hour =	$181,2831
19	Total cost of customer being served per hour =	$577,5717
20	Total cost of customer being balked per hour =	$0
21	Total queue space cost per hour =	$0
22	Total system cost per hour =	$1108,8550

Si ahora hacemos nuevamente la búsqueda de óptimo, pero con los nuevos datos, obtendremos que es de 8 servidores a un costo de 1051,45, como vemos en la figura que sigue:

Capacity Analysis for ensMMck

06-10-2002 15:14:45	Number of Server	Queue Capacity	Total Cost	Busy Server Cost	Idle Server Cost	Waiting Customer Cost	Served Customer Cost
1	7	100	$1108,8520	288,7856	61,2144	181,2806	577,5712
2	8	100	$1051,4500	292,1418	107,8582	67,1668	584,2835
3	9	100	$1062,5090	293,3587	156,6413	25,7920	586,7174
4	10	100	$1097,5090	293,8274	206,1726	9,8538	587,6548
5	11	100	$1141,6770	294,0099	255,9901	3,6569	588,0198

Sistemas con capacidad de espera limitada (M/M/c/L)

Para sistemas de este tipo agregaremos dos nuevos indicadores de rendimiento, que son:

- La probabilidad de que a un cliente que llega le sea denegado el servicio porque el recinto de espera está completo y, por tanto, es rechazado. ***Pd***
- El costo asociado con la pérdida de un cliente no atendido, que se agrega al costo por servidor y al costo por espera

Para conocer más sobre este modelo, seguiremos el siguiente ejemplo:
Una empresa planea lanzar una oferta de venta masiva telefónica por cinco días para lo cual debe decidir qué tipo de sistema debe alquilar a la compañía telefónica. Esta ofrece sistemas de líneas rotativas de 15 y de 20 líneas con opciones de espera de 5 y de 10 llamadas o sin opciones de espera para cada una:

MODELO	LÍNEAS	LLAMADAS EN ESPERA	ALQUILER ($/DIA)
150	15	0	150
155	15	5	180
1510	15	10	225
20	20	0	220
25	20	5	264
210	20	10	330

En sistemas telefónicos se demuestra que usar la distribución de Poisson para la tasa de llegadas es razonablemente una buena aproximación. Supondremos los siguientes datos:

Llegadas, λ = 150 llamadas por hora
Servicio por línea telefónica μ = 12 llamadas por hora.
Se sabe que la compra promedio por llamada (incluyendo los que desisten) es de $50
Cuando al cliente le da ocupado, se estima que el 80% llama de nuevo, el costo de perder un cliente será:
Cd = costo de la llamada perdida x probabilidad de perder al cliente = $ 50 x 0,20 = $ 10.

Lo que haremos ahora es utilizar WinQSB y cargar los datos del problema para una central modelo “Modelo 150”, es decir, de 15 líneas sin capacidad de espera.

Servidores: 15

Capacidad de espera: 0
Tasa de servicio: 12 cl/h
Tasa de arribo: 150 cl/h
Población: Inf

Costos:
- del servidor ocupado: 1,25 $/h
- del servidor libre: Igual
--de espera 0
--de atención 0
--de cliente rechazado 10 $/cl

Data Description	ENTRY
Number of servers	15
Service rate (per server per hour)	12
Customer arrival rate (per hour)	150
Queue capacity (maximum waiting space)	0
Customer population	M
Busy server cost per hour	1.25
Idle server cost per hour	1.25
Customer waiting cost per hour	0
Customer being served cost per hour	0
Cost of customer being balked	10
Unit queue capacity cost	

Los renglones "*Busy (idle) server cost per hour*" *(costo por servidor, ocupado y libre*) se calculan así:

Ya hemos estimado el costo por pérdida de un cliente. Nos falta estimar los costos del servicio (*Busy (idle) server cost*). Para ello usamos la siguiente expresión, la que hemos construido teniendo en cuenta que, en este caso, cada servidor es una línea telefónica que es atendida 8 h diarias:

$$C_s = \frac{150\$ / dia}{8h / dia \times 15 lineas} = 1{,}25\$ / h.linea$$

El costo total de los servidores será 1,25 $/hora.linea x 15 líneas = 18,75 $/hora.

El costo de espera, en este caso, suponemos que no existe, por lo tanto, el costo por hora de clientes en el sistema (que es el producto del costo de espera por el número de clientes que esperan), no existe tampoco. ***Cw*** = 0

06-10-2002	Performance Measure	Result
1	System: M/M/15/15	From Formula
2	Customer arrival rate (lambda) per hour =	150,0000
3	Service rate per server (mu) per hour =	12,0000
4	Overall system effective arrival rate per hour =	134,9266
5	Overall system effective service rate per hour =	134,9266
6	Overall system utilization =	74,9593 %
7	Average number of customers in the system (L) =	11,2439
8	Average number of customers in the queue (Lq) =	0
9	Average number of customers in the queue for a busy system (Lb) =	0
10	Average time customer spends in the system (W) =	0,0833 hours
11	Average time customer spends in the queue (Wq) =	0 hour
12	Average time customer spends in the queue for a busy system (Wb) =	0 hour
13	The probability that all servers are idle (Po) =	0,0005 %
14	The probability an arriving customer waits (Pw) or system is busy (Pb) =	10,0489 %
15	Average number of customers being balked per hour =	15,0734
16	Total cost of busy server per hour =	$14,0549
17	Total cost of idle server per hour =	$4,6951
18	Total cost of customer waiting per hour =	$0
19	Total cost of customer being served per hour =	$0
20	Total cost of customer being balked per hour =	$150,7336
21	Total queue space cost per hour =	$0
22	Total system cost per hour =	$169,4836

En el renglón 14 de la figura anterior, aparece ***Pb***. Como es un sistema de capacidad de espera nula, esta es la probabilidad de negación de servicio. El costo total por denegación del servicio será el producto del costo de negación de un cliente ***Cd*** por el número de llegadas λ por la probabilidad de denegación del servicio ***Pd*** :

costo total por denegación = ***Cd*** $\boldsymbol{\lambda}$ ***Pd*** = 10 x 150 x 0,1005 = 150,75 $/h

Así el costo total por hora para un sistema de 15 líneas sin capacidad de espera será:

Costo total = Costo de los servidores + costo de la espera + costo de la negación del servicio

Costo de los servidores = Costo de un servidor x número de servidores
Costo de la espera = Costo de un cliente en espera x largo de la cola
Costo de la negación del servicio = Costo de negación del clente x tasa de llegada x probabilidad de denegación

Costo total = ***Cs. c*** + ***Cw L*** + ***Cd*** $\boldsymbol{\lambda}$ ***Pd***
Costo total = 1,25 x 15 + 11,2439 x 0 + 10 x 150 x 0,1005 = 169,50 $/h

Como tenemos una oferta de seis sistemas telefónicos diferentes, realizaremos un análisis similar para cada uno de ellos y tabularemos los indicadores obtenidos, como en la siguiente tabla.

	Modelo					
	150	**155**	**1510**	**20**	**25**	**210**
Líneas	15	15	15	20	20	20
Espera	0	5	10	0	5	10
C_s	1,2500	1,5000	1,8750	1,3750	1,6500	2,0625
C_w	0,0000	0,0000	0,0000	0,0000	0,0000	0,0000
C_d	10,0000	10,0000	10,0000	10,0000	10,0000	10,0000
L	11,2350	12,7220	13,5650	12,3310	12,5270	12,5550
p_d	0,1005	0,0311	0,0114	0,0135	0,0013	0,0001
Costo total ($/h)	169,5000	69,08	45,2600	47,7800	34,9000	41,4300

`De ella podemos inferir que deberíamos seleccionar el modelo 25: 20 líneas con una capacidad de espera de 5 clientes, ya que presenta el menor costo operativo.

Otros modelos

Por supuesto que no hemos agotado la revisión de todos los modelos posibles. Por ejemplo, el modelo M/D/c es aquel en que el tiempo de servicio es Determinístico, en el cual cada cliente requiere el mismo tiempo de servicio — por ejemplo, una línea en la que hay una envasadora-formadora —.

En cambio, en un sistema M/G/c tendremos una distribución de probabilidad de servicio distinta de la exponencial o abarca aquellos casos en que la distribución de probabilidad no es conocida. En estos casos la distribución se denomina General (G). Estas posibilidades se extienden a las llegadas y ambos procesos pueden tener cualquier combinación. Por ejemplo, D/G/c o G/M/c.

En caso de que usemos la distribución General, deberemos conocer, además de la tasa de arribos promedio, λ, la cantidad media de tiempo de servicio y la desviación estándar del tiempo de servicio. Si es cero, entonces el tiempo es determinístico. (M/D/c)

Ejemplos:

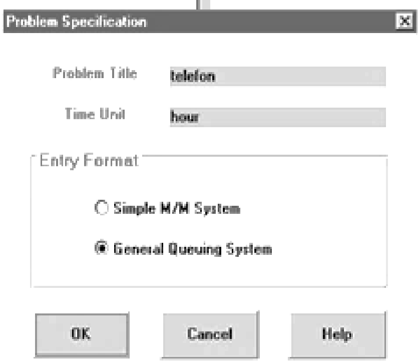
Supongamos un sistema de una cola con tres servidores, cuyos clientes llegan con distribución de Poisson de λ = 46 clientes/hora. Un estudio del tiempo de servicio arroja que cada servidor necesita un promedio de 5 minutos (0,08333 h) para atender un cliente con una desviación estándar de 2 minutos (0,0333 h). Esto indica que cada servidor puede procesar un promedio de μ = 12 clientes /hora.

En este caso, al ingresar los datos lo haremos con los siguientes parámetros:

1. Formato de entrada: *General Queuing System*
2. En el renglón *Service Time distr (in hours),* (distribución de tiempo de servicio), buscaremos la distribución *General*, (haciendo doble clic sobre la palabra *exponential*) lo que habilitará dos renglones, uno para la media y el otro para desviación estándar.
3. Las unidades de los valores a cargar deben ser coherentes: la media será 1/12 (horas), la desviación estándar será 0,03333 horas y λ será 1/46.
4. El renglón *Interarrival Time distr*. Debe ser *Exponencial* (generalmente aparece por defecto), con el parámetro de locación***, a***, igualado a cero y el parámetro de escala, la media, ***b* = 1/λ**
5. Por último, cuando se pide la resolución debe optarse por el método general y **no** por la simulación Monte Carlo. (que discutiremos en el siguiente capítulo).

Observe las diferencias en la carga y resultados:

Data Description	ENTRY
Number of servers	4
Service time distribution (in hour)	General/Arbitrary
Mean (u)	.0833
Standard deviation (s>0)	.0333
(Not used)	
Service pressure coefficient	
Interarrival time distribution (in hour)	Exponential
Location parameter (a)	0
Scale parameter (b>0) (b=mean if a=0)	0.02173913
(Not used)	
Arrival discourage coefficient	
Batch (bulk) size distribution	Constant
Constant value	1
(Not used)	
(Not used)	
Queue capacity (maximum waiting space)	M
Customer population	M
Busy server cost per hour	
Idle server cost per hour	
Customer waiting cost per hour	
Customer being served cost per hour	
Cost of customer being balked	
Unit queue capacity cost	



06-10-2002	Performance Measure	Result
1	System: M/G/4	From Approximation
2	Customer arrival rate (lambda) per hour =	46,0000
3	Service rate per server (mu) per hour =	12,0048
4	Overall system effective arrival rate per hour =	46,0000
5	Overall system effective service rate per hour =	46,0000
6	Overall system utilization =	95,7950 %
7	Average number of customers in the system (L) =	15,8321
8	Average number of customers in the queue (Lq) =	12,0003
9	Average number of customers in the queue for a busy system (Lb) =	13,2109
10	Average time customer spends in the system (W) =	0,3442 hours
11	Average time customer spends in the queue (Wq) =	0,2609 hours
12	Average time customer spends in the queue for a busy system (Wb) =	0,2872 hours
13	The probability that all servers are idle (Po) =	0,4252 %
14	The probability an arriving customer waits (Pw) or system is busy (Pb) =	90,8363 %
15	Average number of customers being balked per hour =	0
16	Total cost of busy server per hour =	$0
17	Total cost of idle server per hour =	$0
18	Total cost of customer waiting per hour =	$0
19	Total cost of customer being served per hour =	$0
20	Total cost of customer being balked per hour =	$0
21	Total queue space cost per hour =	$0
22	Total system cost per hour =	$0

Debemos hacer, en este punto, dos comentarios:

1) estos sistemas sólo son aplicables a una sola fila, con política FIFO y el cliente ya atendido lo abandona. Otros sistemas, no contemplados acá, pueden comprender múltiples filas, atención en lotes, llegada en lotes, atención con discriminación, sistemas en los que el cliente al llegar y ver que la cola es grande decide retirarse, etc. Generalmente se abordan por simulación (ver capítulo siguiente).
2) En el capítulo de Simulación desarrollaremos un modelo de colas con velocidades de atención distintas para cada servidor y distribuciones de probabilidad diferentes para cada cola.

Apéndice al Capítulo 16

Algunas funciones de distribución de probabilidad en hoja de cálculo aplicables al tema de colas

A partir del año 2010 Excel ha modificado la sintaxis de algunas funciones. Consulte en la página de ayuda de Excel sobre dichos cambios. La sintaxis en CALC es muy similar, pero en ella, no todas las herramientas de análisis estadísticos están disponibles. Consulte los foros para averiguar sobre dichas limitaciones en caso de ser necesario. En este apéndice se hacen referencias a las dos sintaxis en la medida de lo posible. Pueden existir diferencias mínimas, además, en la sintaxis que emplea Calc, pero las ayudas de cualquiera de todos estos programas son suficientes para sortear las dificultades que pudieran presentarse.

Uso de la función DISTR.EXP.N

Devuelve la probabilidad de una variable aleatoria continua siguiendo una distribución exponencial. Se usa para calcular el tiempo entre dos sucesos, tal como el tiempo que tarda una máquina de cajero automático en entregar dinero.

La función DISTR.EXP.N puede usarse para determinar la probabilidad de que el proceso demore un determinado lapso como máximo.

Sintaxis

DISTR.EXP(X; λ; *acum*)

DISTR.EXP.N(X, λ; *acum*)

X es el valor de la función.

λ es el valor del parámetro.

acum es un valor lógico que indica qué forma de la función exponencial debe proporcionarse. Si el argumento *acum* es VERDADERO, DISTR.EXP.N devuelve la función de distribución acumulada; si es FALSO, devuelve la función de densidad de la probabilidad.

La ecuación para la función de densidad de la probabilidad es:

$$f(x;\lambda) = \lambda e^{-\lambda x}$$

La ecuación para la función de distribución acumulada es:

$$F(x;\lambda) = 1 - e^{-\lambda x}$$

Uso de la función POISSON.DISTR

Devuelve la probabilidad de una variable aleatoria siguiendo la distribución de Poisson. Una de las aplicaciones comunes de la distribución de Poisson es la predicción del número de sucesos en un determinado período de tiempo, como, por ejemplo, el número probable de automóviles que llegan a una zona de peaje en el intervalo de un minuto.

La distribución de Poisson se caracteriza por

1. la media es pequeña respecto al número total posible de casos (condición de *suceso raro*)
2. la incidencia de un suceso es independiente de incidencias previas (condición de *aleatoridad*)

La función se construye con una serie

$$\frac{1}{e^{\lambda}};\frac{\mu}{1!\,e^{\lambda}};\frac{\mu^2}{2!\,e^{\lambda}};\frac{\mu^3}{3!\,e^{\lambda}};\ldots;\frac{\mu^r}{r!\,e^{\lambda}}$$

donde cada término es la frecuencia relativa esperada del suceso raro ***x***,

para ***x*** = 0, 1, … , ***r***.

y la media λ se considera igual a la media μ y a la varianza σ^2

Sintaxis:

POISSON(*k*; λT; acumulado)

POISSON.DIST(*k*; λT; acumulado)

k es el número de sucesos.

λT es el valor numérico esperado. (Es el producto λT en este texto, pero hay que tener cuidado porque en las ayudas puede aparecer simplemente como λ)

Acumulado es un valor lógico que determina la forma de la distribución de probabilidad devuelta. Si el argumento acumulado es VERDADERO, POISSON devuelve la probabilidad de que un suceso aleatorio ocurra un número de veces comprendido entre 0 y *k* inclusive; si el argumento acumulado es FALSO, la función devuelve la probabilidad de que un suceso ocurra exactamente *k* veces.

POISSON se calcula como:

Si el argumento *acumulado* = FALSO:	**Si el argumento *acumulado* = VERDADERO:**
$POISSON = \frac{e^{-\lambda T}(\lambda T)^{k}}{k!}$	$POISSON = \sum_{k=0}^{x} \frac{e^{-\lambda T}(\lambda T)^{k}}{k!}$

CAPÍTULO 17. SIMULACIÓN

Puede ocurrir que necesitemos describir una realidad muy compleja mediante un modelo matemático y que nos encontremos que – paradójicamente – el modelo resulte tan complejo como esa realidad o que, por ejemplo, sea exageradamente complicado, tanto en sí mismo como en los resultados que arrojaría. Esto sería muy poco práctico.

Es en estos casos — y otros parecidos — cuando se recurre a los métodos de simulación. Los simuladores describen la realidad, o una parte de ella, utilizando elementos que emulen los objetos o sistemas en estudio y su comportamiento. Los simuladores pueden ser físicos, computacionales, matemáticos, etc. y siempre constituyen un sistema en si mismo que representa solo algunas de las características del sistema estudiado.

Veamos un caso simple, por ejemplo, queremos usar un modelo del tipo G/G/ para describir una estación de peaje con tres vías, una de las cuales puede estar preparada para pago exacto u otros medios de pago (tarjeta, por ejemplo). Significa que hay un servidor que trabaja a diferente velocidad que los otros dos y que los clientes pueden elegirlo o no, aunque también pueden elegir alguno de los otros en función del largo de la cola o de otros factores, como comodidad o gusto. Por último, podremos agregar que hay determinadas franjas horarias con mayor afluencia de clientes en general o de determinado tipo (camiones, por ejemplo) que cambia la velocidad de atención. En este caso, lo que vimos en el capítulo anterior, solo podría aplicarse a cada cola, pero no a determinar la forma y velocidad o tasa de llegada de clientes a esa cola (o porqué llega a esa y no a otra) y, por lo tanto, sería imposible describir el sistema en conjunto.

Pero podemos diseñar un modelo que imite ese escenario real y someterlo a cambios del estado de la naturaleza a fin de observar comportamientos resultantes. Por ejemplo, si bien el modelo de colas se puede utilizar para describir el funcionamiento de cada ventanilla del peaje, con un modelo de simulación se pueden incluir vehículos que demoren más que otros, o que opten por una u otra cola usando criterios de elección basados en afinidad o cantidad de clientes en espera o si tienen tarjeta de pago.

Se aplican modelos de simulación en casos como la administración de una clínica médica, para comprender el impacto de agregar más personal de enfermería y de apoyo, o en el diseño de procesos productivos, para identificar cuellos de botella en cada línea. También pude usarse en la administración de un aeropuerto, para ver la influencia de cambios de tarifas o de horarios o de factores climáticos sobre el congestionamiento de tránsito, y en muchas otras aplicaciones.

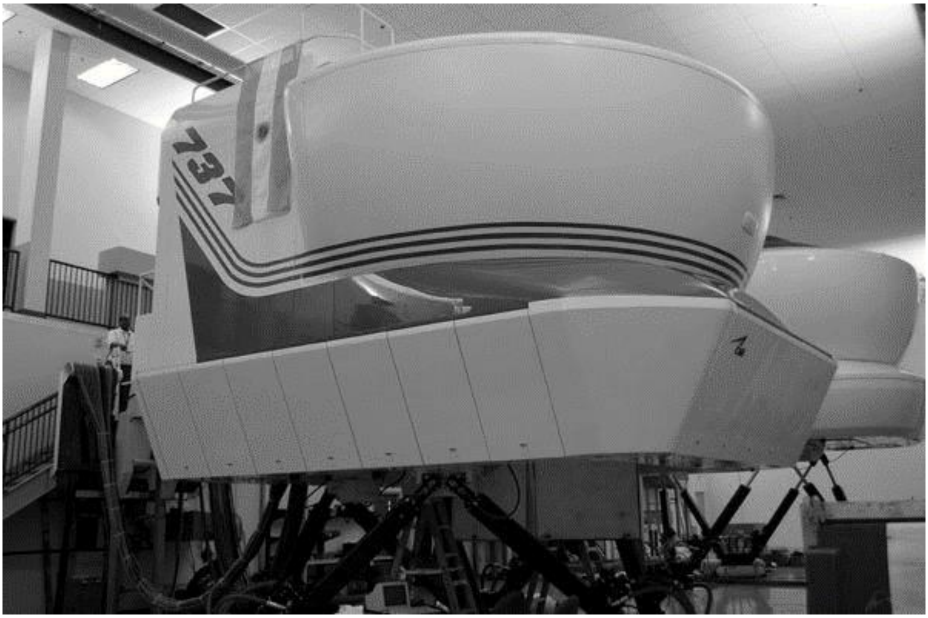
Simulación

Claude.E. Shannon[4] definió ***simulación*** de esta manera: "*La simulación es el proceso de diseñar un modelo de un sistema real y llevar a cabo experiencias con él, con la finalidad de comprender el comportamiento de ese sistema o evaluar nuevas estrategias — dentro de los límites impuestos por un cierto criterio o un conjunto de ellos — para el funcionamiento del sistema*".

[4] Claude Elwood Shannon (1916 — 2001) matemático, ingeniero eléctrico y criptógrafo americano considerado el padre de la teoría de la información. Fundó el campo de la teoría de la información en 1948. Es uno de los que sentó las bases de la teoría del diseño de la computadora digital en 1937. A los 21 años, en el MIT, demostró que las aplicaciones electrónicas de álgebra booleana podrían construir cualquier relación lógico-numérica. Contribuyó al campo del criptoanálisis en la Segunda Guerra Mundial, con trabajos sobre el descifrado de códigos y la seguridad en las telecomunicaciones.

Muchos "juegos de simulación" se basan en esta definición: están aquellos que, por ejemplo, permiten simular estrategias con diversos grados de complejidad o deportes extremos o juegos de supervivencia, los juegos de guerra en los colegios militares, o la teoría de juegos aplicada en decisiones que discutimos en otro libro de la serie Optimiza.

También podemos aplicar el concepto a equipamiento especializado que permite analizar comportamientos en situaciones de alto riesgo sin poner en peligro la vida o la propiedad, como los simuladores (entrenadores) de vuelo, de conducción, de maquinaria, de entrenamiento, etc.



También podemos encontrar los simuladores de realidades mucho más complejas, como los simuladores de desarrollos sociales (un ejemplo comercial y muy conocido es el programa "Sim-City", que se vendía como un juego para PC pero que es capaz de simular un desarrollo social cambiante ante hechos o decisiones determinadas)

Las imágenes de arriba ilustran un simulador para vehículos y un modelo nuevo de simulador para las series 700 y 800 de aviones *Boeing 737* en uso en Aerolíneas Argentinas).
La posibilidad de realizar este tipo de modelos ha dado origen a varios desarrollos conceptuales y a profundizaciones de la teoría de sistemas.

Desde el punto de vista de la estabilidad, los sistemas pueden ser clasificados en tres grupos:

- ***Sistemas estables***: aquellos que tienden, en el tiempo a una posición puntual o espacial (órbita): reconocen un atractor
- ***Sistemas inestables***: aquellos que no reconocen o escapan de los atractores
- ***Sistemas caóticos***: manifiesta los dos comportamientos y se refiere a subsistemas que son atraídos por un atractor, pero tienen objetos o se relacionan con otros sistemas que, simultáneamente, lo alejan de él.

En un **sistema estable** del que se conocen sus ecuaciones características, y con unas condiciones iniciales fijas, podremos conocer exactamente su evolución en el tiempo. Un ejemplo de sistema estable es el sistema solar modelizado en un instante cualquiera contemporáneo.

En cambio, una de las características de los **sistemas inestables** es su independencia de las condiciones iniciales o cuando pequeñas perturbaciones lo sacan del estado estacionario en que se encontraba.

En los **sistemas caóticos**, una mínima diferencia en esas condiciones hace que el sistema evolucione de manera totalmente distinta. La atmósfera terrestre, desde el punto de vista de su dinámica, es un ejemplo de sistema caótico.

Sistemas caóticos

En 1963, Lorenz[5], describió un modelo meteorológico basado en tres ecuaciones diferenciales bien definidas. Supuso que, conociendo las condiciones iniciales se podría predecir el clima. Sin embargo, no tuvo éxito, ya que al ser éste un sistema caótico, y

[5] Edward Norton Lorenz (1917 – 2008) matemático y meteorólogo estadounidense, trabajó sobre la rotación de los fluidos y en la dinámica atmosférica y predicción climatológica. Planteó por primera vez la teoría del caos, introdujo el concepto de atractores extraños el de *efecto mariposa*.

no poder conocer nunca con exactitud los parámetros que fijan las condiciones iniciales (en cualquier sistema de medición, por definición, siempre se comete un error, por pequeño que éste sea) hace que, aunque se conozca el modelo, éste se aleja de la realidad pasado un cierto tiempo. Por otra parte, el modelo atmosférico es teórico y puede no ser perfecto, y el determinismo, en el que se basa, es también teórico.

Así fue como Lorenz comenzó a definir lo que hoy es la teoría del caos: "Un sistema caótico es sensiblemente dependiente de cambios interiores de las condiciones iniciales (...) una propiedad esencial es que estados próximos entre sí terminan por divergir sin importar lo pequeñas que puedan ser las diferencias iniciales."

En cualquier sistema con una dependencia sensible, es imposible poder realizar predicciones perfectas, tal es el caso en las predicciones meteorológicas donde los resultados no son precisos y pueden variar de un momento a otro. Un sistema no estacionario es capaz de entrar y salir del caos.

Lorenz enunció que los atractores extraños pueden ser un conjunto de estados, y que su representación gráfica es un atractor.

"Un conjunto de estados de un sistema que se dan una y otra vez, o que son aproximadamente los mismos una y otra vez, cada vez más próximos entre sí pertenecen al conjunto de los atractores".

Estas consideraciones, que acá hemos simplificado mucho, podrían llevar a suponer que cuando se habla de Caos se habla de una teoría y a asumir que Caos es "ausencia de orden". Sin embargo, no necesariamente es una teoría, sino que puede interpretarse como un campo de investigación, que abarca diferentes líneas de pensamiento.

La idea inicial de la Teoría del Caos es simple: en determinados sistemas naturales, pequeños cambios en las condiciones iniciales conducen a enormes discrepancias en los resultados

Caos está entendido como cierto tipo de orden de características impredecibles, pero descriptibles en forma concreta y precisa. Es decir: un tipo de orden de movimiento impredecible.

Efecto Mariposa

Es un concepto que hace referencia a la sensibilidad de un sistema a las condiciones iniciales. Su nombre proviene supuestamente de un proverbio chino: "*el aleteo de las alas de una mariposa se puede sentir al otro lado del mundo*". Si bien se acepta que el término fue introducido por Lorenz hay quien lo relaciona, simultáneamente con el escritor Ray Bradbury[6] ("El ruido de un trueno") en la génesis del nombre.

La idea es que, dadas unas condiciones iniciales de un determinado sistema natural, la más mínima variación en ellas puede provocar que el sistema evolucione en formas totalmente diferentes.

Cuando Lorenz presenta su trabajo en la Academia de Ciencias de Nueva York, tratando de predecir el clima a través de modelos matemáticos que relacionaban variables, lograba predecir la meteorología del día siguiente. Al revisar los datos se dio cuenta que, haciendo pequeñísimos cambios en ellos, se lograban resultados absolutamente diferentes[7].

Consecuencia de este efecto es el abandono de la idea determinista de una naturaleza mecánica y un tratamiento dentro de la Teoría General de Sistemas de Bertalaffy[8]. La naturaleza no se asemeja a modelos previsibles y determinados, sino que existe un orden aparentemente aleatorio en los acontecimientos, que incapacita al hombre y su saber científico en predecir y controlar la realidad.

Simulación en computadora

En realidad, la utilización de simulaciones se puede aplicar a problemas determinísticos o determinados para tratar de encontrar un modo de comportamiento con resultados compatibles con la "realidad", ya que, muchas veces, los datos estadísticos, por si solos, o no son suficientes, o no alcanzan a explicarla.

---

[6] Ray Douglas Bradbury (1920 – 2012) Escritor estadounidense del género fantástico y ciencia ficción. Reconocido por su obra Crónicas marcianas (1950) y la novela distópica Fahrenheit 451 (1953).

[7] Este tema dio (y da) argumentos para varias películas, historias y especulaciones (la novela de Michael Crichton*, Jurassic Park* – llevada al cine – se basa en una interpretación de la teoría del caos, las películas "*Efecto Mariposa*", "*Torre de Babel*", "*Corre Lola, corre"* y otras, hasta un episodio de la serie *Los Simpson*, y algunos de la serie "*Black Mirrow*" tratan estos temas). En esta publicación se incluye el cuento completo de Bradbury mencionado, al que nosotros consideramos como pionero, ya que fue escrito unos años antes de que Lorenz comenzara a mencionar el caos y el efecto mariposa. (hay una versión digital del cuento en *www.optimiza.org*).

[8] Karl Ludwig von Bertalanffy (1901 – 1972) Biólogo y filósofo austríaco, reconocido fundamentalmente por su teoría general de sistemas. Consultar en *www.optimiza.org* otras publicaciones nuestras que desarrollan este tema.

Por ejemplo, en el típico caso del informe médico: “Lo suyo no es grave, ya que solamente se complica en un paciente cada 1000”, si bien tiene una base estadística correcta, al paciente no le da ninguna información concreta sobre lo que le realmente puede pasar a él individualmente, más allá que, por aplicaciones erróneas de la estadística en situaciones singulares, para el paciente en cuestión será solamente un proceso binario: le fue bien o mal[9].

Con el siguiente ejemplo, veremos ilustrado un simple caso de simulación, fácil y posible:

Lotería del Oeste lanza un juego consistente en una “*raspadita”.* Son tarjetas con tres filas. En cada fila hay dos casillas, una con un valor oculto de $ 1 y la otra con uno de $ 5. El jugador debe raspar solamente una casilla de cada renglón. Si obtiene los tres

[9] Hay un ejemplo muy conocido. Una muy popular marca de aspirinas, en Argentina, publicita su producto diciendo que el ingerir una dosis diaria “previene uno de cada tres infartos” en un claro error (¿) del publicista que confunde la conclusión de algún estudio (“se encontró que en los pacientes con este tratamiento se produjo un 33% menos de infartos, cuando se los comparó con los que recibieron un placebo”) con una selectiva eliminación de “uno cada tres” en los infartos individuales, lo cual no es cierto médicamente.

renglones con números iguales gana la cantidad indicada. ¿Cuál es la mínima cantidad que se debe cobrar por tarjeta para obtener una ganancia esperada?

Como es un problema simple, podemos aplicar la teoría de probabilidades. Pero lo que vamos a hacer es construir el modelo que permita obtener la respuesta y después comparar con la teoría. Buscamos saber la cantidad a cobrar por tarjeta, que debe ser al menos igual a las ganancias esperadas por los apostadores, que es lo que la Lotería debe pagarles. Primero, olvidemos la simulación y calculemos la ganancia esperada por cada tarjeta (para el apostador). La calculamos como la esperanza matemática:

Ganancia esperada =
\$ 1 x (probabilidad de ganar \$ 1) +
\$ 5 x (probabilidad de ganar \$ 5)

La probabilidad que tiene un apostador de ganar \$1 (o \$5) es la probabilidad de que la casilla raspada en las tres filas tenga 1 (o 5):

P(\$1) = P(fila1,\$1) x P(f2,\$1) x P(f3,\$1) = ½ x ½ x ½ = 1/8 ,
P(\$5) = P(fila1,\$5) x P(f2,\$5) x P(f3,\$5) = ½ x ½ x ½ = 1/8 ,

por tanto

Ganancia esperada = \$ 1 x (1/8) + \$ 5 x (1/8) = \$ 0,75

En conclusión, el valor 0,75 \$/tarjeta, es el que equilibrará gastos en premios con ingresos.

Es un resultado aceptable. Pero, como en el caso del médico, no dice nada sobre un emprendimiento concreto: quien compra una tarjeta podrá no ganar, o ganar \$ 1 o ganar \$ 5, no tiene significado una ganancia expresada como E(g) = 0,75.

Supongamos que se decide encarar esto como un negocio, con una tirada inicial de 1000 tarjetas. ¿cuánto debemos pagar en concepto de premios? Según acabamos de calcular, sabemos que tenemos una esperanza matemática de pagar 0,75 \$ por cada tarjeta que vendamos.

Pero, es un indicador estadístico. ¿Funcionará igual para 100 tarjetas? ¿para 1000? ¿tiene que ver con la cantidad? ¿con la "suerte"? Entonces decidimos simular que vamos a vender solamente 100 tarjetas y sacar conclusiones, nuevamente, aplicando estadística, pero con otro enfoque.

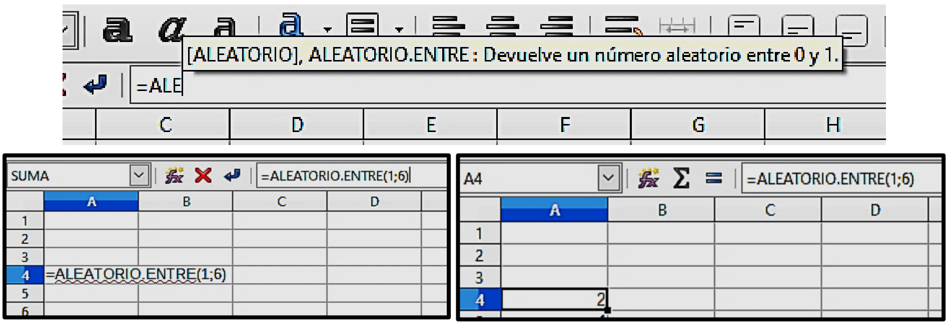
Para simular este problema podría elegir varias maneras. Veamos, antes, un simulador muy sencillo, más aún que el de “la raspadita”, al que volveremos.

Simular un dado

Queremos averiguar si, al arrojar un dado, va a salir un *as*. Como sabemos estadísticas podemos afirmar que, tirándolo repetidas veces, en uno de cada seis tiros vamos a obtener el *as*. Eso es formalmente correcto, pero, en la realidad, si estoy jugando, al tirar el dado una sola vez, podrá salir o no, y si me dan la posibilidad de tirar siete veces ¿va a salir en el primer tiro y en el séptimo? ¿Se va a encontrar un as en seis tiros? Si hago la prueba veré que, a veces, en el primer tiro sale y otras veces se tiran diez o doce veces el dado y no sale ninguno, o sale cinco veces. Estos resultados parecen erráticos, pero, sin embargo, no lo son, ya que tirando muchas más veces obtendré una proporción de salidas que, efectivamente, ***tiende*** a ser de 1 en 6.

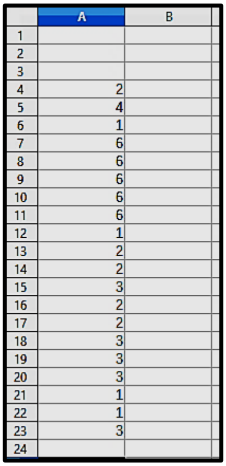
Sería interesante conocer, por un lado, lo que el modelo predice: *un as cada seis tiros*, basado en que, en muchos tiros se va a dar esa proporción, por otro lado, frente a un tirador determinado, en un número acotado de tiros, ¿Cuántos ases van a salir? ¿tiene “suerte”?

Vamos a construir un simulador que nos permita tirar un dado todas las veces que queramos y sacar conclusiones sobre lo que ocurre. Para eso vamos a usar una hoja de cálculo y la función “=ALEATORIO.ENTRE”:



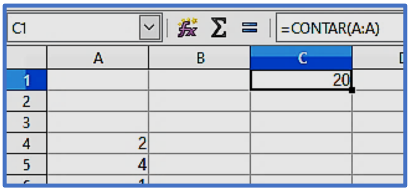
¡Hemos construido un simulador! se tira un dado y, obviamente, va a entregar un número entero entre uno y seis. ¿Cuándo? cada vez que se pulsa la tecla F9 (o se cambie el contenido de cualquier otra celda) se producirá un recálculo, equivale a una nueva tirada de dado.

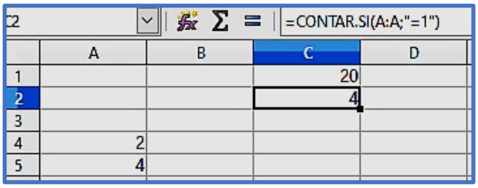
Como nos damos cuenta que para hacer un trabajo más completo queremos saber qué pasa si tiro el dado varias veces, sería tan tedioso usar este simulador como el dado real, luego de cada tiro debería anotar el resultado y llevar una "contabilidad" de los eventos. Así que el siguiente paso será crear una lista idéntica equivalente a un número determinado de "tiradas", pueden ser tres tiradas, diez o mil. Por ejemplo, veinte veces. Ver figura a la derecha.



El último paso será hacer la contabilidad y sacar conclusiones. Para ello usaremos las siguientes funciones, una en cada celda desde C1 a C3:

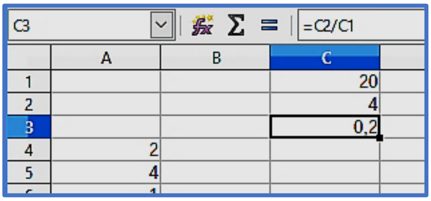
=CONTAR(A:A), que nos permite saber cuántas "veces" se tiró el dado, ya que cuenta como evento todo lo que se escribe en la columna "A"





=CONTAR.SI(A:A;"=1"), con lo que podemos contar solo si los dados que salieron son *ases*. Si miramos con atención la sintaxis notamos que el criterio de comparación se puede cambiar a cualquier otro valor. Podemos reemplazar "1" por otra "cara" del dado.

=C2/C1, que nos permite presentar la frecuencia de aparición del *as* en el ensayo. Deberíamos comparar este valor con el teórico (*P(1)* = 1/6 = 0,16666….) para poder extraer conclusiones (en el caso del ejemplo se aprecia marcada diferencia)



Finalmente podemos "embellecer" incorporando elementos para facilitar el uso del simulador que acabamos de construir:

	A	B	C	D
1		Tiradas:	20	
2		Ases:	4	
3		Promedio:	0,2000	
4	2			
5	4			
6	1			
7	6			
8	6			
9	6			

Notamos que, simplemente agregando renglones en la columna A, podemos ampliar la simulación de 20 a cientos, miles o decenas de miles de tiradas de dado, sin modificar el resto del "simulador".

Como simular la "raspadita"

Volviendo al caso de las tarjetas de la "raspadita", podremos intentar conocer lo que puede ocurrir fabricando algunos cartones y entregándolos, en carácter de prueba, a una cierta cantidad de apostadores. Esa sería una ***simulación física***, realizada imprimiendo, por ejemplo 200 tarjetas, dándoselas a algunas personas para que raspen y tabular los resultados. Usando los valores obtenidos para las ganancias se puede calcular un promedio y una desviación. Esa es una estimación de ganancia y la podemos usar para tomar decisiones. ¿Por qué se querría hacer esto? El modelo probabilístico no incluye cosas vagas e imposibles de definir como "la buena (mala) suerte del jugador" "la corazonada" "una buena (mala) racha" la posible dependencia entre resultados y cantidades y otras por el estilo que tratan de explicar los desvíos puntuales y en números acotados de ensayos que se apartan del modelo.

Otra manera de simular sería mediante ***simulación analógica***: por ejemplo, lanzando al aire monedas (P = ½): tres lanzamientos cara son $1, tres cruces son $5, los demás son $ 0. Este enfoque u otro similar, evita imprimir tarjetas, pero no la tediosa tarea de anotar los resultados uno por uno (y juntar las monedas del piso). La simulación por computadora es ideal para estos problemas, ya que se pueden arrojar, tabular los resultados y calcular lo necesario mediante una sola operación simple. (y no juntar las monedas del piso).

Simulando "la raspadita" en hoja de cálculo

Como ya vimos en los capítulos dedicados a Programación lineal, había varias posibilidades para trasladar esos problemas a una hoja de cálculo. Ocurre lo mismo en simulación, aunque los modelos son diferentes. Vamos a desarrollar un ejemplo, y, seguramente, el lector podrá encontrar otras alternativas para esta propuesta. Nuestra propuesta es:

1) Destinamos un reglón cualquiera (en este ejemplo, el renglón 7) para simular una tarjeta a “raspar”, que será la primera.
2) A partir de él, los siguientes renglones (filas) serán la segunda tarjeta, (el renglón 8); y así sucesivamente hasta el 107 que será la centésima tarjeta.
3) La columna A, la vamos a reservar para simular un evento aleatorio que representa la primera raspada. Así, en cada renglón, a partir del 7 y hasta el 106 tendremos un número aleatorio que representará la primera raspada, la de la primera fila de la supuesta tarjeta. Ese número aleatorio tendrá distribución uniforme entre 0 y 1.
4) Lo mismo haremos en las columnas B y C, en las que representaremos la segunda raspada en la segunda fila de la “tarjeta”, y la tercera, respectivamente.
5) Las columnas D, E y F, desde el renglón 7 al 106, serán destinadas a representar el “resultado” de la función aleatoria de las columnas A, B y C, respectivamente. Para ello, vamos a convertir la función aleatoria uniforme entre 0 y 1 en una función entera binaria, también uniforme, es decir en 1 o 5 con la misma probabilidad de ocurrencia.
6) La columna G será usada, desde el renglón 7 al 106, para verificar si la tarjeta es o no ganadora y, en caso de ser, cuánto ganó (1 o 5). Allí tendremos tres valores posibles: 0, 1 y 5. (No ganó, ganó 1 o ganó 5, respectivamente)

Así, podremos completar el renglón número 7 de la hoja de cálculo hasta la columna G. En las figuras siguientes se ve, primero, como se completa desde la columna A hasta la columna C:

	A	B	C	D
1				
2				
3				
4				
5				
6				
7	**=ALEATORIO()**	**=ALEATORIO()**	**=ALEATORIO()**	
8				

En la próxima figura vemos como se completa desde la columna D hasta la columna F. En la última figura se muestra la columna G. En las tres figuras se ven las funciones utilizadas, no los valores que realmente van a aparecer en cada celda.

			C	D	E	F	G
1							
2							
3							
4							
5							
6							
7			**=ALEATORIO()**	**=SI(A7<0,5;1;5)**	**=SI(B7<0,5;1;5)**	**=SI(C7<0,5;1;5)**	
8							

						F	G
1							
2							
3							
4							
5							
6							
7						**=SI(C7<0,5;1;5)**	**=SI(Y(D7=E7;E7=F7);SI(D7=1;1;5);0)**
8							

Así, tenemos toda la fila 7 con las siguientes expresiones:

CELDA	FORMULA	COMENTARIOS
A7	=ALEATORIO()	Calcula un nº aleatorio entre 0 y 1 en A, B y en C
B7	=ALEATORIO()	
C7	=ALEATORIO()	
D7	=SI(A7<0,5;1;5)	Si el nº en A7,(y en B7 y en C7) es menor que 0,5 escribe 1, si es mayor escribe 5
E7	=SI(B7<0,5;1;5)	
F7	=SI(C7<0,5;1;5)	
G7	=SI(Y(D7=E7;E7=F7);SI(D7=1;1;5);0)	Si los tres son iguales y si valen 1, escribe 1 si son iguales y valen 5, escribe 5 caso contrario (son distintos) escribe 0

Usando las facilidades de arrastre esta “hoja—simulador” podemos completar las 100 tarjetas en pocos segundos, ya que solo escribimos la fila 7 en las columnas A, B, C. D, E, F y G, y luego expandimos hasta el renglón deseado (habíamos propuesto el 106, pero el lector verá que puede ser cualquier otro) a fin de tener el número de renglones equivalentes a la cantidad de tarjetas que se simulan.

Solo nos queda realizar la contabilidad y un reporte estadístico con los resultados obtenidos, para ello, usaremos las primeras filas (renglones) de la columna I:

	...	H	I	J
1				
2			=CONTAR(A:A)	calcula el número de ensayos
3			=CONTAR.SI(G:G;”<>0)	cuenta cuantas ganadoras hubo
4			=SUMA(G:G)	calcula el monto total a pagar
5			=PROMEDIO(G:G)	calcula el premio medio de todas las ganadoras
6				

En este caso hemos habilitado todos los renglones de cada columna desde la A hasta la G, no solamente desde el 7 al 106 (100 tarjetas como nos habíamos propuesto), lo que nos permite, en el futuro, poder hacer simulaciones rápidamente agregando o quitando tarjetas (renglones) sin necesidad de hacer cambios en el sector de la hoja que hemos destinado a la contabilidad.

Una vez completada la hoja obtendremos “resultados” como el de la figura siguiente. Debemos tener presente que la función ALEATORIO() es volátil, esto quiere decir que se recalcula cada vez que el programa tiene que revisar la hoja activa y recalcular todas las celdas. Eso ocurre ante cualquier evento como, por ejemplo, cambiar un valor en una celda cualquiera, esté o no relacionada con otras, o cuando pulsamos la tecla F9.

	A	B	C	D	E	F	G	H	I
2								apostadores	208
3								ganaron	44
4								total ganado	148
5								promedio	0,71153846
6				A	B	C	R		
7	0,20630315	0,31459622	0,86076555	5	5	1	0		
8	0,78983073	0,13891297	0,47561373	1	5	5	0		
9	0,21372341	0,18892554	0,64909311	5	5	1	0		
10	0,95479392	0,24238902	0,71386591	1	5	1	0		
11	0,08840578	0,44453924	0,03125071	5	5	5	5		
12	0,12452714	0,19234289	0,21083278	5	5	5	5		
13	0,80003555	0,36045198	0,21020529	1	5	5	0		
14	0,63750984	0,19545429	0,76789179	1	5	1	0		
15	0,67656921	0,797495	0,40967969	1	1	5	0		
16	0,67559883	0,26738936	0,24226886	1	5	5	0		
17	0,5337724	0,39004039	0,79795781	1	5	1	0		
18	0,95501491	0,41273075	0,64006545	1	5	1	0		

Uso de Visual Basic para construir macros

Además de lo que hemos visto, existe la posibilidad de desarrollar una *macro* en ***Visual Basic*** (tanto para Office como para Calc). No es objeto de esta publicación ahondar en el lenguaje de programación *Visual Basic for Aplications* (VBA). Quien quiera utilizar el lenguaje puede intentar, en una hoja Excel ya abierta, usar la opción "*Herramientas/Macros/Editor de Visual Basic*" (opción disponible tanto en Excel como en las últimas versiones de Calc). Una vez abierto el editor seleccionaremos la opción "*nuevo módulo"* o, simplemente hacemos clic en el botón que recuerda la tecla Play de un grabador, con lo cual, al no haber macros solamente se abrirá el primer módulo disponible y se podrá comenzar a escribir código siguiendo la ayuda del programa y los ejemplos que siguen.

Generadores de números pseudo aleatorios

Todos los lenguajes de programación son capaces de generar números aleatorios[10] con distribución uniforme. A continuación, transcribimos un pequeño programa en VBA capaz de hacerlo:

```
Sub Raspadita()
'
' Raspadita Macro
'
       Rnd (-1)
       Randomize (2)
For i = 1 To 200
       a = Rnd
       b = Rnd
       c = Rnd
       If a > 0.499 Then a = 1 Else a = 5
       If b > 0.499 Then b = 1 Else b = 5
       If c > 0.499 Then c = 1 Else c = 5
       If a = b And b = c Then
              gano = gano + a
              aciertos = aciertos + 1
       End If
Next i
       Range("a3").Value = "Aciertos  : "
       Range("c3").Value = aciertos
       Range("a4").Value = "% de ac.  : "
       Range("c4").Value = aciertos * 100 / 200
       Range("a6").Value = "Ganancias : "
       Range("c6").Value = gano
       Range("a7").Value = "Promedio  : "
       Range("c7").Value = gano / 200
End Sub
```

[10] Este tema se profundiza un poco más más adelante, en este mismo capítulo.

Lo que vemos arriba es un conjunto de instrucciones escritas en el lenguaje Visual Basic. Cuando ejecutamos este programa (en la terminología informática se dice "corremos la macro") y el programa lee cada renglón, uno por uno, hace exactamente lo que la instrucción ordena. Si hay un apóstrofe en algún lugar del renglón ('), lo que se escribió a la derecha de él, es ignorado. Esto lo podemos ver en los renglones 2, 3 y 4, que los usamos para escribir el nombre que, para nuestro uso, le pusimos a la macro.

Cada una de las "sentencias" o instrucciones que vemos arriba tiene un significado preciso, que — brevemente — es el siguiente:

SUB Raspadita

Indica el inicio del programa (SUB) y el nombre que le dimos nosotros (Raspadita). Todo lo que sigue en este apartado escrito en negrita y mayúsculas son las palabras "*reservadas*" de VisualBasic, las cuales no pueden usarse para otra cosa más que para dar estas instrucciones precisas, ni pueden ser escritas con otra sintaxis.

RND(–1**)**

es una instrucción especial que se utiliza en VBA para obtener un número aleatorio con un argumento negativo cualquiera (en este caso –1). Este llamado, antes de "*Randomize*", provoca que cada vez que se corre el programa se genera la misma secuencia de números aleatorios. Esto es útil para comprobar que el programa funciona predeciblemente.

RANDOMIZE (2**)**

Prepara una secuencia de números aleatorios, para lo cual el operador debe proveer una *semilla* (número cualquiera entre – 32000 y 32000 aprox.). Si no lo hacemos, (se usamos solo RANDOMIZE(), sin el 2 dentro del paréntesis) el programa va a buscar como *semilla* una función llamada "TIMER" que devuelve un número generado por el reloj del sistema. En este caso se usa 2, para obtener valores que podamos comparar entre diferentes corrida o con nuestros compañeros.

FOR I=1 **TO** 200

Esta es una de las funciones *VisualBasic* más usadas, se denomina "ciclo FOR… NEXT" y sirve para realizar un ciclo (bucle) repetitivo de acciones mientras se mantenga determinada condición. En este caso, la letra "I" es el nombre de una variable que se nos ocurrió a nosotros[11], de esta manera, se dará a la

[11] El tema de variables es más extenso, por ahora diremos solamente lo siguiente: nosotros, como programadores de macros podemos definir variables, en este caso la letra "I", donde "I" es el nombre que le hemos dado a la variable. Ese nombre puede ser cualquier otro, por ejemplo "Mario", "X1" "A" "SumaTotal", con la salvedad que no se pueden usar como nombres palabras que VB ha reservado para otros fines (por

variable “I” un número inicial de 1, realizará las instrucciones que siguen hasta la instrucción NEXT. Allí incrementará su valor en una unidad y repetirá el mismo bloque hasta NEXT y seguirá así hasta que, cuando “I” alcance el valor de 200 en vez de repetir el bloque va a pasar al renglón siguiente a la instrucción NEXT (que en este caso es la que comienza con RANGE…)

a=**RND;** b=**RND;** c=**RND**

En estos tres renglones, (que se leen y ejecutan 200 veces a medida que avanza “I”) en cada uno de los ciclos, guardaremos en tres variables (llamadas a, b y c) el número aleatorio consecutivo obtenido de la secuencia preparada en el paso 1, borrando el número que había en esas variables remanente del ciclo anterior.

IF a > .499 **THEN** a=1

Otra de las funciones fundamentales es “IF…THEN…[ELSE][12]” que provocan lo que se denomina “bifurcaciones”. El programa ejecuta acciones que tienen que ver con una condición lógica. En este caso, comparamos el número aleatorio que hemos obtenido y guardado durante el paso anterior en la variable llamada “a”. Cada uno de los tres números obtenidos sigue una distribución de probabilidad uniforme y su valor está entre 0 y 1. Al comparar con 0,499 estamos averiguando si está en la mitad inferior o en la superior del rango. Como para cada número la distribución es uniforme, tendrá la misma probabilidad (0,5) de que sea inferior o superior a la mitad. Si el número es mayor suponemos que obtuvimos un “1” al raspar y guardamos ese valor, 1, en este caso, en la misma variable a, reemplazando el número aleatorio obtenido en el paso anterior y guardado en “a”.

ELSE a = 5

---

ejemplo, no se pueden usar “For”, ni “FOR”, ni “for”, ni “End”, etc.). En las variables se puede almacenar un número (hay ciertas reglas en cuanto a la longitud del mismo, o una palabra o frase (“String”). Cada vez que “escribimos” algo en una variable (por ejemplo, escribimos un “5” en la variable “AX”), lo que había antes en esa variable se borra, salvo que, al escribir decidamos hacer alguna operación con ese valor previo. Por ejemplo, si en AX estaba el número 3245, si ejecutamos AX=5, en AX quedará 5. Pero si ejecutamos AX = AX + 5, en AX quedará 3250

[12] El uso de corchetes en las guias de sintaxis indica que el término (en este caso ELSE), puede ser omitido si no es necesario que esté. Otro ejemplo: aunque nosotros no lo mostramos, FOR…NEXT tiene un término entre corchetes [STEP] que simboliza el “paso” de avance del contador. Por ejemplo, si la variable xw va a contar desde 0 a 100000 de 100 en 100, se escribe FOR xw=0 TO 100000 STEP 100. Si se omite, contará de 1 en 1. Si la variable j va a contar disminuyendo desde 10 a 1 se escribe FOR j=10 TO 1 STEP —1.

Si no es mayor supone que se obtuvo un “5”, y será este el valor que guardamos en “a”

Luego repetimos este paso con las otras dos variables, b y c.

NOTA: en estos últimos pasos hemos guardado, en tres posiciones “de memoria” (llamadas a, b y c) tres números que van, cada uno de ellos, entre 0 y 1, que inmediatamente los hemos convertido en tres números que pueden valer solo 1 o 5, cada uno de ellos

IF a=b **AND** b=c **THEN**

En este caso, utilizamos la función “IF…THEN” para algo un poco más complejo. La combinamos con una prueba lógica (AND) y podemos leerla de esta manera: “***Si*** a=b ***y*** b=c***, entonces…”.*** Con esto buscamos saber si los tres números guardados en a, b y c son iguales (tres unos o tres cincos). Esa será una tarjeta ganadora. Si no lo son, es “no—ganadora”. Ahora la función IF…THEN no termina en un renglón como las anteriores. La usamos como un “bloque”. Los renglones que siguen serán las instrucciones que hay que realizar si la condición resulta cierta. Estos renglones “terminan” cuando hay una instrucción “ELSE” (que acá no existe, pues no hay nada que hacer si es no—ganadora) y siempre una instrucción “END IF”, que acá si existe

gano = gano + a

En este punto, si la tarjeta es ganadora, en una nueva variable que hemos bautizado “gano”, agregamos al contenido que había antes el valor de cualquiera de las variables donde habíamos guardado el resultado de la raspadita, en este caso, la variable “a” (todas valen lo mismo, pues es tarjeta ganadora. En la primera “vuelta por el ciclo, “gano” tiene un cero, pues el programa al iniciarse pone todas variables en cero. Así sabremos el total acumulado a pagar por todas las tarjetas ganadoras

aciertos = aciertos + 1

En la misma condición que la variable anterior, en esta nueva variable llevamos una suma de cuantas tarjetas ganaron. La variable “aciertos” es incrementada en una unidad por tarjeta ganada. Así sabremos cuantas tarjetas ganaron.

NOTA: Es importante tener en cuenta que estos dos pasos, manipulando las variables “gano” y aciertos” se realizan **solamente** si las tres “raspaditas” (a, b y c) son iguales, o sea, ganaron 1 o 5.

END IF

Indica que hasta allí llegó el bloque de instrucciones “IF… THEN…[ELSE]”. Vemos que “IF…THEN…[ELSE]” se puede escribir en un solo renglón, sin necesidad de “END IF” (en el ejemplo, son los primeros tres renglones que comienzan con “IF…”) o en varios renglones, en cuyo caso hace falta señalar donde termina el bloque comenzado con “IF…”. En cualquier caso, la palabra ELSE puede estar (si hace falta) o puede omitirse, en caso contrario, como hicimos en este bloque.

NEXT i

Tanto si el apostador ganó como no, llegamos a esta instrucción y debemos realizar un nuevo ciclo hasta llegar a 200, o sea, “jugar” una nueva tarjeta.

Si ya se llegamos a 200, entonces seguimos con las instrucciones que vienen después de “NEXT…”. Este bloque nos permite hacer la presentación de un informe y la contabilidad final de los resultados.

Para eso vamos a “escribir” los resultados en celdas de la hoja de cálculo.

RANGE(“XX**”).VALUE=** nombre_de_la_variable_cuyo_contenido_queremos_mostrar

RANGE("a3**").VALUE =** "Aciertos : "

En la celda A3 escribimos la palabra “Aciertos:”

RANGE("c3**").VALUE =** aciertos

En la celda siguiente a la derecha de la anterior, escribimos el valor de la variable aciertos, que será el total de tarjetas ganadoras que contamos.

RANGE("a4**").VALUE =** "% de ac. : "

RANGE("c4**").VALUE =** aciertos * 100 / 200

Acá calculamos el porcentaje de tarjetas que ganaron respecto al total hemos “jugado” (200) y lo escribimos en la celda C4

RANGE("a6**").VALUE = "**Ganancias **: ".**

En la celda A6 escribimos la palabra “Ganancias”

RANGE("c6**").VALUE =** gano

Escribimos en C6 el importe total a abonar por todas las tarjetas ganadoras, que habíamos contabilizado en la variable “gano”.

RANGE("a7**").VALUE = "**Promedio **: "**

RANGE("c7**").VALUE =** gano / 200

Por último, acá calculamos el importe medio a pagar por tarjeta jugada, que es dato solicitado inicialmente y lo escribimos en C7.

END SUB

Indica el fin del programa.

La figura siguiente nos muestra una hoja con los resultados obtenidos a partir de usar RND(-1) y semilla[13] “2”.

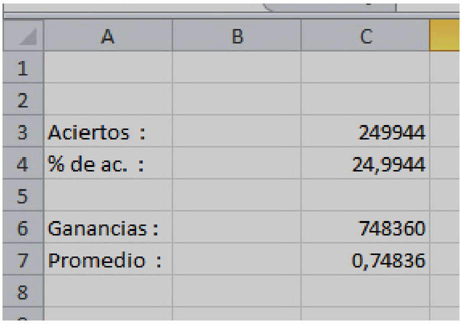
	A	B	C
1			
2			
3	Aciertos :		40
4	% de ac. :		20
5			
6	Ganancias :		120
7	Promedio :		0,6
8			

La ventaja de este programa es que se puede simular para, por ejemplo, 11200 intentos. Los resultados con la misma semilla serian:

	A	B	C
1			
2			
3	Aciertos :		2771
4	% de ac. :		24,7410714
5			
6	Ganancias :		8219
7	Promedio :		0,73383929
8			

O una de 1.000.000 tarjetas, con igual semilla:

[13] Se denomina “semilla” a cualquier número que debe ingresarse para que el algoritmo pueda generar una serie de números pseudoaleatorios. Más adelante volveremos al tema, pero es importante saber que a igualdad de semilla obtendremos una sucesión de números aleatorios con distribución uniforme que será idéntica cada vez que ejecutemos el programa con la misma semilla. En caso de que no necesitemos esa predicibilidad, muchos programadores proveen una semilla que, si bien no es aleatoria, es impredecible. Por ejemplo, leer la hora del sistema (en Basic es la función TIMER).



	A	B	C
1			
2			
3	Aciertos :		249944
4	% de ac. :		24,9944
5			
6	Ganancias :		748360
7	Promedio :		0,74836
8			

Como se puede observar, un simple programa permite hacer cualquier tipo de ensayo y, aún disponer de distintos resultados. Se puede, además, concluir que a medida que avanza el número de apostadores hay una tendencia a un 25% de aciertos y a un promedio de ganancia por tarjera de 0,75, que son los valores que se obtendrían por aplicación lisa y llana del cálculo estadístico.

Segundo ejemplo

En el primer ejemplo generábamos un acontecimiento a partir de que conocíamos la probabilidad de que ocurriera. En este segundo ejemplo veremos una importante variación en la forma de generar el acontecimiento.

Un "Video Club" se mantiene en el mercado porque se dedica exclusivamente a estrenos y a películas y series que no están en plataformas como *Netflix*. Trabaja en formato BD4K, BD y DVD y compra originales exclusivamente a un precio medio de USD$ 25 la copia y las alquila a USD$ 3 por día, luego de un mes las vende a coleccionistas (dejando un "*back up"* en el archivo) a un precio residual de USD$10 la copia. Las probabilidades de demanda diaria de las películas son:

COPIAS	PROBABILIDAD
0	0,15
1	0,25
2	0,45
3	0,10
4	0,05

¿Cuántas copias deben comprarse?

Si se llama ***n*** al número de copias adquiridas, se puede formalizar el problema así:

Ganancia esperada =
ingresos mensuales esperados + precio de venta de las copias — costo de las copias

Ganancia esperada =
ingresos diarios esperados x 30 días + $10 x n — $25 x n

a su vez, los ingresos diarios esperados serán (si compramos las 4 copias):

ingresos diarios esperados =
∑(ingreso por película) x P(número de películas demandadas)

ingresos diarios esperados =
\$0 x P(0 pedidos) + \$3 x P(1 pedido) + \$6 x P(2) + \$9 x P(3) + \$12 x P(4)

ingresos diarios esperados =
\$0 x 0,15 + \$3 x 0,25 + \$6 x 0,45 + \$9 x 0,10 + \$12 x 0,05 = \$ 4,95

con lo cual se tiene, para cuatro películas en todo el mes:

Ganancia esperada =
\$4,95 x 30 días + \$10 x n — \$25 x n = 148,5 +12 — 100 = 88,5:

Pero como esa ganancia es función de las copias que compramos, entonces, debemos calcular para cada estrategia de compra:

a) Ingresos diarios esperados

0 copias	\$0 x 0,15 + \$0 x 0,25 + \$0 x 0,45 + \$0 x 0,10 + \$0 x 0,05 = \$ 0,00
1 copias	\$0 x 0,15 + \$3 x 0,25 + \$3 x 0,45 + \$3 x 0,10 + \$3 x 0,05 = \$ 2,55
2 copias	\$0 x 0,15 + \$3 x 0,25 + \$6 x 0,45 + \$6 x 0,10 + \$6 x 0,05 = \$ 4,35
3 copias	\$0 x 0,15 + \$3 x 0,25 + \$6 x 0,45 + \$9 x 0,10 + \$9 x 0,05 = \$ 4,80
4 copias	\$0 x 0,15 + \$3 x 0,25 + \$6 x 0,45 + \$9 x 0,10 + \$12 x 0,05 = \$ 4,95

Generando una tabla con valores de número de copias n de 0 a 4:

b) Ganancia esperada

n	Ganancia esperada	
	(Ing.Diar.Esp) x 30 días + 10 x n — \$25 x n =	
0	0,00 x 30 + 10 x 0 — 25 x 0 =	0,00
1	2,55 x 30 + 10 x 1 — 25 x 1 =	61,50
2	4,35 x 30 + 10 x 2 — 25 x 2 =	100,50
3	4,80 x 30 + 10 x 3 — 25 x 3 =	99,00
4	4,95 x 30 + 10 x 4 — 25 x 4 =	88,50

Comprobamos que con 2 copias se obtiene un máximo de ganancias esperadas.

Si ahora queremos aplicar a este problema un modelo de simulación de manera que los eventos sean los pedidos de clientes para cada uno de los días del mes, tendremos que realizar los siguientes pasos:

a. Generar la demanda diaria de acuerdo con la distribución de probabilidad dada en el enunciado.
(Este es el cambio conceptual respecto al caso anterior: antes generábamos directamente el evento: "raspábamos" una tarjeta. Ahora, en vez de simular llegadas de clientes vamos a simular que probabilidad le corresponde a cada día)
b. Calcular el ingreso por alquiler basado en que se decidió disponer de **dos** copias de la película.
c. Calcular la ganancia total, de esta manera:

Ganancia esperada total =
ingresos totales por alquiler + ingresos por reventa — costo de compra =
= ingresos totales por alquiler + 2 x 10 — 2 x 25 =
= ingresos totales por alquiler — 30

para obtener los *ingresos totales por alquiler* generaremos una tabla de números aleatorios uniformes, transformándolos en una distribución acorde a la enunciada para el número de copias diarias:

```
a = RND

IF a < .15 THEN a = 0 : Goto final
IF a < .40 THEN a = 1
IF a < .85 THEN a = 2
IF a < .95 THEN a = 3
IF a < 1   THEN a = 4
final:
```

vemos que el intervalo uniforme de los números aleatorios fue dividido en intervalos proporcionales a la distribución enunciada: al generar un número aleatorio cualquiera, por ejemplo, 0,34342, obtendremos la probabilidad de la demanda diaria correspondiente, que en este caso es de una película, ya que el número generado cae entre 0,15 y 0,40.

En el problema anterior, la sucesión de números aleatorios uniformemente distribuida entre 0 y 1 se dividía en dos, como la distribución uniforme de una moneda: todos los números menores que 0,5 eran cara y los mayores eran seca. En ese caso el número aleatorio se utilizaba para simular los que ocurría raspando la tarjeta con una probabilidad del 50%. En este caso del videoclub, el número aleatorio se utiliza para establecer que probabilidad tiene un evento (que probabilidad hay de que el día simulado sea una jornada de x clientes) y proceder según esa probabilidad, considerando que llegaron la cantidad de clientes correspondiente.

Lo que hicimos fue generar un día. Ahora tenemos que llevarlo a los 30 días del mes y que funcione como un modelo de simulación, para eso haremos lo siguiente:

1. Generaremos el número aleatorio para establecer la cantidad de clientes de ese cada día a partir de que probabilidad le correspondió al día.
2. Calcularemos los ingresos diarios basados en las dos copias disponibles.

Lo que sigue es una propuesta de algoritmo simple que nos permite simular lo que ocurre con las ganancias si trabajamos sobre una existencia de DOS copias de las películas, escrito en VBA:

```
RANDOMIZE
FOR i = 1 TO 30
a = RND
IF a < .15 THEN a = 0: GOTO final
IF a < .4 THEN a = 1
IF a < .85 THEN a = 2
IF a < .95 THEN a = 3
IF a < 1 THEN a = 4
final:
IF a = 0 THEN diaria = 0
IF a = 1 THEN diaria = 3
IF a >= 2 THEN diaria = 6
acumula = acumula + diaria
NEXT i

Range(“B2”).Value = "Ingresos totales : " & acumula
Range(“B3”).Value = "Ganancia total   : "& acumula - 30
```

(Hemos elegido la celda B2 para presentar los ingresos totales y la B3 para la ganancia total. Por supuesto que el lector puede presentarlo de otra manera).

El código que hemos escrito se analiza y explica en la siguiente tabla:

Análisis del código	
`RANDOMIZE`	Puesta en marcha del GNA
`FOR i = 1 TO 30`	Establecemos un ciclo de 30 días de simulación (mes)
`a = RND`	Obtenemos un número aleatorio uniforme entre 0 y 1 que corresponde a ese día y lo guardamos en “a”
`IF a < .15 THEN a = 0: GOTO final`	Si “a” es menor que 0,15 significa que no vendrán clientes. Ponemos a=0 y vamos a la etiqueta “final”. Si esta condición no se dio, vamos al renglón siguiente.
`IF a < .4 THEN a = 1`	Si “a” es menor que 0,4 ponemos un 1 en “a”. No hace falta ir a “final” por que “1” es mayor a cualquier número con los que vamos a comparar en los renglones que siguen. Vamos al renglón siguiente tanto si cambiamos “a” como si no lo hicimos.
`IF a < .85 THEN a = 2`	Lo mismo que en el renglón anterior.
`IF a < .95 THEN a = 3`	Lo mismo que en el renglón anterior
`IF a < 1 THEN a = 4`	Lo mismo que en el reglón anterior
`final:`	Etiqueta. Llegamos acá porque “a” tiene ahora un valor entre 0 y 4

`IF a = 0 THEN diaria = 0`
`IF a = 1 THEN diaria = 3`
`IF a >= 2 THEN diaria = 6`
	Guardamos en “diaria” la contabilidad (número de clientes que alquilaron una copia ese día a $3 c/u)

`acumula = acumula + diaria`	Contamos el acumulado mensual de los alquileres, “acumula” vale 0 el primer día y a ese valor se le va sumando lo contabilizado día a día
`NEXT i`	Volvemos a “For… next” para hacer el día siguiente. Si ya hicimos 30, vamos al renglón siguiente.
`Range(“B2”).Value = "Ingresos totales : " & acumula`	Mostramos lo facturado por alquiles en todo el mes, que es el contenido de “acumula”
`Range(“B3”).Value = "Ganancia total : "& acumula - 30`	Mostramos la ganancia tota del mes, que es el contenido de “acumula” menos 30

Ejemplo de reporte obtenido:

Ingresos totales : 126
Ganancia total : 86

Con este método podremos hacer ensayos de manera tal que, cambiando en el código unos pocos elementos, nos permite registrar los resultados para 1, 2, 3 o 4 copias y luego construyendo una tabla final. Comenzamos modificando el código de arriba para que nos quede así:

```
Dim diaria(4)
Dim acumula(4)

Randomize
For i = 1 To 30
   a = Rnd
   If a < 0.15 Then a = 0: GoTo final
   If a < 0.4 Then a = 1
   If a < 0.85 Then a = 2
   If a < 0.95 Then a = 3
   If a < 1 Then a = 4
final:
   If a = 0 Then diaria(1) = 0: diaria(2) = 0: diaria(3) = 0: diaria(4) = 0
   If a = 1 Then diaria(1) = 3: diaria(2) = 0: diaria(3) = 0: diaria(4) = 0
   If a = 2 Then diaria(1) = 3: diaria(2) = 3: diaria(3) = 0: diaria(4) = 0
   If a = 3 Then diaria(1) = 3: diaria(2) = 3: diaria(3) = 3: diaria(4) = 0
   If a = 4 Then diaria(1) = 3: diaria(2) = 3: diaria(3) = 3: diaria(4) = 3
   For w = 1 To 4
      acumula(w) = acumula(w) + diaria(w)
   Next w
Next i

Cells(2, 2).Value = "EL PROBLEMA DEL VIDEO CLUB"
Range("B3").Value = "NumCopias"
Cells(3, 3).Value = "Ganancias"
Range("B4").Value = 0
Range("C4").Value = 0
For w = 1 To 4
   NumeroRenglon = w + 4
   Cells(NumeroRenglon, 2).Value = w
   For a = 1 To w
      sumatoria = sumatoria + acumula(a)
   Next a
   Cells(NumeroRenglon, 3).Value = sumatoria - 25 * w + 10 * w
   sumatoria = 0
Next w

End Sub
```

El análisis del código es:

Análisis del código	
Dim diaria(4)	Esta instrucción sirve para declarar, antes de ejecutar el programa, el nombre de una memoria. En estos casos, al tener un valor entre paréntesis (4), significa que con el mismo nombre (diaria) tendremos cuatro variables diferentes a las que se puede acceder con un índice numérico a cada una de ellas,[14].
Dim acumula(4)	Idem
Randomize	Igual explicación que en el problema anterior
For i = 1 To 30	
a = Rnd	
If a < 0.15 Then a = 0: GoTo final	
If a < 0.4 Then a = 1	
If a < 0.85 Then a = 2	
If a < 0.95 Then a = 3	
If a < 1 Then a = 4	
final:	
If a = 0 Then diaria(1) = 0: diaria(2) = 0: diaria(3) = 0: diaria(4) = 0	la diferencia con el anterior es que ahora solo escribiremos ingresos por 3$ cada vez que alquilemos la película que corresponde a esa memoria (la primera película, la tercera, etc) y esto va a ocurrir siguiendo una distribución de probabilidad que estamos generando. El resultado será acumulado en la variable "acumula()" (1 para cada cantidad de película disponible, de 1 a 4) y ahí va a existir la sumatoria de todo lo recaudado por película (primera, tercera, etc.).
If a = 1 Then diaria(1) = 3: diaria(2) = 0: diaria(3) = 0: diaria(4) = 0	
If a = 2 Then diaria(1) = 3: diaria(2) = 3: diaria(3) = 0: diaria(4) = 0	
If a = 3 Then diaria(1) = 3: diaria(2) = 3: diaria(3) = 3: diaria(4) = 0	
If a = 4 Then diaria(1) = 3: diaria(2) = 3: diaria(3) = 3: diaria(4) = 3	
For w = 1 To 4	vamos a guardar con el índice w lo que recaudamos hoy por ventas por películas
acumula(w) = acumula(w) + diaria(w)	por ejemplo, si solo tenemos 2 clientes, acumularemos a lo que había en acumula(1) $3 más y en acumula(2) otros $3 más. En las demás acumula() se agrega cero.
Next w	próximo índice, o, si ya completamos los 4, seguimos con lo de abajo
Next I	Próximo día , si se terminó el mes, seguimos con lo de abajo
Cells(2, 2).Value = "EL PRO...UB"	escribimos en la celda (Cells) renglón2, columna 2. Es equivalente a "Range" con la ventaja de que así podremos direccionar celdas y columnas usando operaciones matemáticas o lógicas
Range("B3").Value = "NumCopias"	se intercaló "Range" para mostrar la equivalencia con Cells.
Cells(3, 3).Value = "Ganancias"	
Range("B4").Value = 0	
Range("C4").Value = 0	
For w = 1 To 4	usamos el índice w nuevamente para hacer la contabilidad para cada cantidad de películas que podemos disponer de 1 a 4.
NumeroRenglon = w + 4	Como usamos el índice w también para avanzar por los renglones y ya usamos los primeros 4 renglones de la hoja de cálculo, le sumamos 4 a w para comenzar desde el renglón 5 e ir hasta el 8

[14] . Es similar a una dirección postal: Sarmiento 453. Si hay cuatro departamentos, será Sarmiento 453, Depto. 2. Si nosotros queremos escribir 7600 en el "departamento" 2 de "*diaria*" tendremos que indicarlo así: **diaria(2)=7600**

Cells(NumeroRenglon, 2).Value = w	escribimos que ese renglón corresponde a w copias
For a = 1 To w	vamos a buscar en todos los acumula() desde acumula(1) hasta acumula(w) (numero de copias disponibles) para saber la ganancia total que daría tener w copias. Para eso definimos el índice ***a***
sumatoria = sumatoria + acumula(a)	calculamos las ventas totales por todas las películas que se alquilaron cuando disponíamos w copias (1 +2+…+w)
Next a	avanzamos ***a*** en una unidad. Si ya llegó a w seguimos con el próximo renglón
Cells(NumeroRenglon, 3).Value = sumatoria - 25 * w + 10 * w	contabilidad correspondiente a w copias
sumatoria = 0	ponemos en cero la variable sumatoria por que es una variable que, al ser un acumulador, no se iniciará por sí misma.
Next w	siguiente número de copias. Si ya hicimos las cuatro, se sigue abajo, que es el fin de la rutina
End Sub	

	A	B	C	D
1				
2		EL PROBLEMA DEL VIDEO CLUB		
3		NumCopias	Ganancias	
4		0	0	
5		1	66	
6		2	111	
7		3	108	
8		4	96	

Una de las ventajas de tomarse el trabajo de hacer un simulador como este, es que, con pocos agregados o modificaciones, se pueden ampliar las prestaciones para simular varios meses o períodos de más días o variaciones similares. Podemos partir de otras hipótesis, por ejemplo: cambio de las probabilidades según el mes del año, según el día de la semana, según un determinado pronóstico de lluvias, etc.

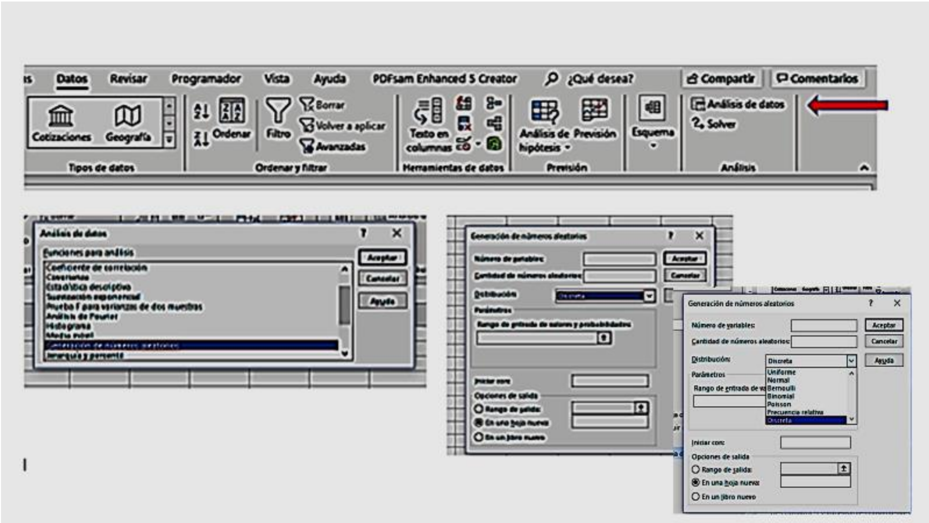
Nota sobre las funciones en hoja de cálculo

Excel y Calc generan números aleatorios que cambian al recalcular la celda. Podemos, no obstante, generar un número aleatorio que quede fijo, para lograrlo, usamos la función **ALEATORIO()** escribiéndola en la barra de fórmulas y, en lugar de ***Enter*** pulsamos **F9**. De esta manera, en la celda se aparecerá el número aleatorio y no la función. También es posible generar números aleatorios entre otros valores diferentes a cero y uno, usando la expresión

=ALEATORIO()*(b-a)+a

siendo ***a*** y ***b*** los extremos entre los cuales se quiere generar el número real aleatorio. Por ejemplo para generar un número entre 0 y 100 debe usarse **=ALEATORIO()*100**. Existe, además, la función **=ALEATORIO.ENTRE(m;M)** que es similar a la expresión que mostramos antes, ya que genera series entre m y M, pero, en este caso, de números **enteros**, por lo tanto, el número menor del rango, ***m***, no debe ser inferior a uno.

Por último, en Excel, existe otra alternativa para generar números aleatorios, que se encuentra disponible en la pestaña "*Datos*", en el bloque "*Análisis de Datos*"/"*Solver*"[15] (en las figuras siguientes se ve, paso a paso, como se accede)



Como podemos ver, esta opción nos permite realizar más de una serie de variables, y, aplicando el método de Montecarlo, elegir distribuciones de probabilidad diferente a la uniforme.

Metodología de la simulación

Uno de los inconvenientes con que podemos tropezar al trabajar con simulación es que nos basamos en conjuntos definidos de datos, por tanto, podemos llegar a estimaciones erróneas (sobreestimar el valor de la ganancia de la raspadita, por ejemplo). Para evitar esto, conviene aumentar el número de ensayos, por ejemplo, realizar hasta 1 millón de tarjetas a raspar, pero siempre utilizando análisis estadístico de los resultados, incluyendo la dispersión y grado de confianza. Es impensable construir un simulador sin

[15] Igual que en el caso de *Solver*, ya comentado en el Libro 1, esta opción puede requerir instalación y/o habilitación.

incorporarle las herramientas estadísticas que requiera. Otra corrida del problema de las cintas de video iniciada con una semilla 23, dio $ 92 de ganancia para dos copias y $ 96 para tres copias. Por otro lado, cada problema requiere su propio diseño de simulación.

Obviamente la solución sería, si fuera posible, utilizar técnicas analíticas, sin embargo, las técnicas de simulación se utilizan muy a menudo, aprovechando ventajas comparativas, como las siguientes:

1. Cuando el problema es complejo, la simulación nos permite resolverlo, aunque no se conozcan los métodos analíticos adecuados: en el caso del alquiler de películas, por ejemplo, en vez de suponer una distribución de probabilidades igual para cada día, en la práctica es seguro que la demanda varía en los fines de semana, o en determinadas épocas del año, o que algunos clientes no la devuelvan de un día para el otro. Todas estas alternativas son relativamente sencillas de resolver por simulación, pero resulta muy difícil hacerlo analíticamente (aun conociendo la distribución de la probabilidad de la devolución).
2. La posibilidad de modificar varios parámetros para experimentar sin por ello aumentar costos o impactar sobre el mundo real.
3. Se pueden hace ensayos comprimiendo el tiempo, por ejemplo: impacto sobre la economía de una obra social al cabo de un año si se disminuye el precio a pagar por el afiliado en los medicamentos, usando series históricas de prestaciones por afiliado. Esto se puede simular con pocos minutos sin esperar un año en un ensayo piloto, por ejemplo.

Clasificación del sistema

Cualquier tipo de simulación que hagamos siempre será un modelo, físico, conceptual o virtual. Ese modelo, a su vez, será un sistema que representará a otro sistema más complejo o imposible de modificar para analizar cambios. Por ello, conocer conceptualmente los sistemas será el primer paso en el diseño de un modelo de simulación. Para nuestra mirada sobre este tema, podemos simplificar clasificando a los sistemas en ***continuos*** o ***discretos***:

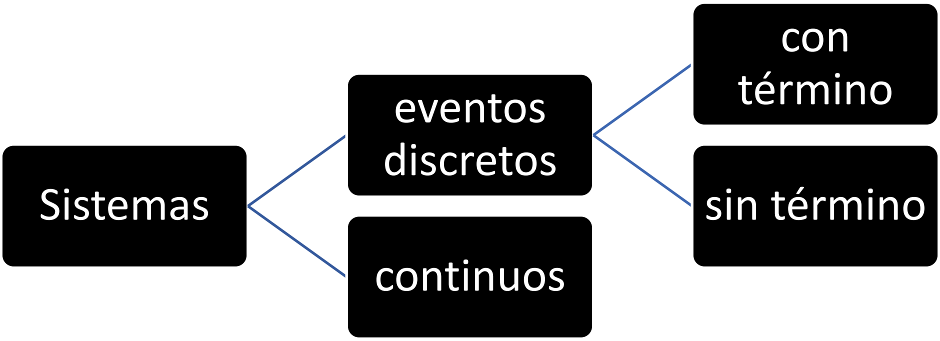
1) **De eventos discretos**. El estado del sistema solo cambia en determinados puntos del tiempo. Ejemplo un modelo de colas: el estado del sistema cambia solo cuando un cliente es atendido (sale de la cola o del sistema) o cuando un nuevo cliente llega.

   Pueden ser:

   a. **Sistemas con término** son los que tienen puntos definidos de inicio y finalización: se compran las películas (inicio) y se venden 30 días después (terminación). En estos casos la longitud de la simulación (o cantidad de tiempo sobre la cual se hará la simulación) va desde el punto de inicio al de terminación y deberemos determinar cuántas veces repetiremos el análisis.

b. **Sistemas sin término** al contrario de los anteriores no tienen esos puntos: un proceso de fabricación en donde las condiciones de finalización del lunes serán las de inicio del martes. En estos casos habrá que determinar las repeticiones y también la longitud de cada simulación.

2) **Continuo**. El sistema cambia continuamente en el tiempo. Estos sistemas no se incluyen en este texto.



Identificación de componentes generales de un sistema de simulación

Los componentes del sistema de simulación se pueden identificar en grandes grupos genéricos en cuanto a la función que representan. Así, podremos mencionarlos en los términos siguientes:

1) **Condiciones iniciales**: valores que expresan el estado del sistema al principio de la simulación. En el caso de los sistemas caóticos, recordamos que tienen alta sensibilidad a las condiciones iniciales. Por contrario un sistema estacionario, prácticamente, es independiente de ellas.

2) **Datos determinísticos**. Valores conocidos necesarios para procesar y así calcular la salida del modelo. (por ejemplo, el número de empleados disponibles y las horas diarias de labor de cada uno de ellos).

3) **Datos estocásticos**. Son valores de los cuales se conoce solo su probabilidad de ocurrencia (a veces subjetiva) necesarios para procesar y así calcular la salida del modelo. Puede ocurrir que no sea posible conocer o asignar probabilidad. Por ejemplo, en un restaurante, si se conocen cuantos empleados trabajan y cuánto tiempo cada uno (datos determinísticos), la llegada de clientes y el tiempo de permanencia en el local, son datos probabilísticos, cuyo valor es incierto pero que su comportamiento general se conoce a través de una función de probabilidad (si el valor es discreto) o de densidad, (si el valor es continuo).

En resumen, un sistema de simulación contará con los siguientes componentes:

Podemos identificar estos componentes en el primer ejemplo visto de alquiler de película, para lo cual se transcribe la parte del código que solo tiene que ver con el problema (hemos quitado, por ejemplo, las definiciones de variables) y le agregamos comentarios:

RANDOMIZE	**Generamos una secuencia de números aleatorios con distribución uniforme que representarán eventos**
FOR i = 1 TO 30	**Iniciamos 30 ciclos, uno por cada día del mes**
a = RND	**generamos un número aleatorio entre 0 y 1 que representa la probabilidad de clientes asignada a ese día**
IF a < .15 THEN a = 0: GOTO final IF a < .4 THEN a = 1: GOTO final IF a < .85 THEN a = 2: GOTO final IF a < .95 THEN a = 3 ELSE a = 4	**Convertimos la distribución uniforme en la distribución de probabilidades específica del caso, asignando un grupo discreto a cada valor original.** **De esta manera se generó una cantidad concreta de clientes para ese día. Luego de convertir, el programa continúa en la etiqueta “final”**
final:	**Etiqueta**

IF a = 0 THEN diaria = 0	**Hacemos la contabilidad del día: cuantos clientes llegaron y cuánto dinero ingresó por alquiler.**
IF a = 1 THEN diaria = 3	
IF a >= 2 THEN diaria = 6	
acumula = acumula + diaria	**Acumulamos a la ganancia de lo que va del mes la ganancia del día.**
NEXT i	**Vamos al próximo día. Si ya fueron 30 seguimos con el renglón inferior**
Range("B2").Value = acumula	**En la celda B2 escribimos los ingresos mensuales acumulados**
Range("B3").Value = acumula – 40	**Escribimos en la celda B3 la ganancia neta**

Generación de números aleatorios

En los ejemplos anteriores hemos usado números aleatorios que aparecen con la misma probabilidad (uniformemente) entre 0 y 1 y luego hemos transformado esos números con el fin de obtener salidas con una distribución de probabilidad discreta, conocida y diferente de la uniforme.

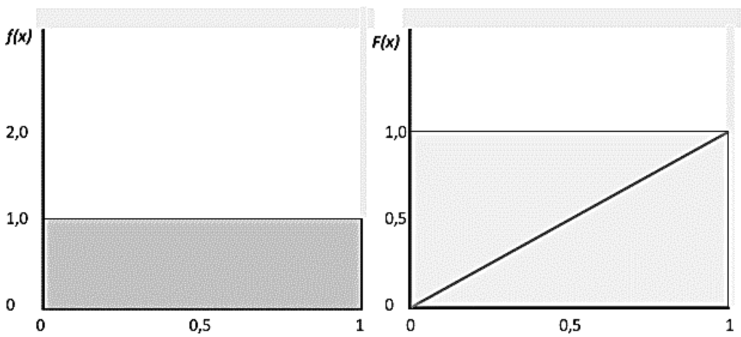
Cuando estudiamos los fenómenos de colas vimos que hay casos en los que las distribuciones de probabilidad son continuas. Por ejemplo, el proceso de llegadas suele ser descripto con una distribución exponencial, del tipo:

$$f(t) = \frac{1}{\lambda} e^{-\lambda t}$$

Generación de números aleatorios con distribución uniforme

La función de densidad ***f(x)*** de los números aleatorios con distribución uniforme usados hasta ahora es: ***f(x)*** = 1; mientras que la función de distribución acumulativa es:

F(x) = 1



Los lenguajes de computación tienen capacidad de generar series pseudo—aleatorias mediante un algoritmo. Requieren de un valor inicial que determina la serie y que se denomina ***semilla***. Los números generados están distribuidos uniformemente entre cero y uno y son estadísticamente independientes entre sí. Estas series se denominan "*pseudo—aleatorias*" debido a que tienen dos particularidades clave:

a) a igualdad de semilla, dos o más series de números generados son idénticas entre sí. Significa que, si en un programa se usa una semilla dad con la que el primer número obtenido es 0,6667, el segundo es 0,234, y el duodécimo es 0,8007; la próxima vez que se ejecute el programa, van a salir esos números en esas mismas posiciones.
b) Las series de números a obtener podrán ser muy numerosas, por ejemplo, una serie puede ser de más de 200 millones de números aleatorios. Pero por más grande que sea esa cantidad siempre es finita. "Agotada" se va a repetir exactamente igual hasta completar otra serie, y así sucesivamente.

Para solucionar (si es que necesitamos solucionarlo) el primer inconveniente, se puede recurrir a que el generador "busque" una semilla impredecible en fuentes que el operador no pueda manipular. Excel, por ejemplo, busca un número inicial en el registro de tiempo que cada sistema operativo tiene para contar el uso del equipo o de un programa. Linux busca en el ruido blanco de la fuente de alimentación. Pero no siempre se necesita un número aleatorio así. Al contrario, muchas veces es necesario comparar series para verificar simuladores, y en esos casos es útil tener una semilla y saber cómo será la serie para comparar distintas versiones de nuestro simulador.

Generación de una serie continua distinta de la uniforme.

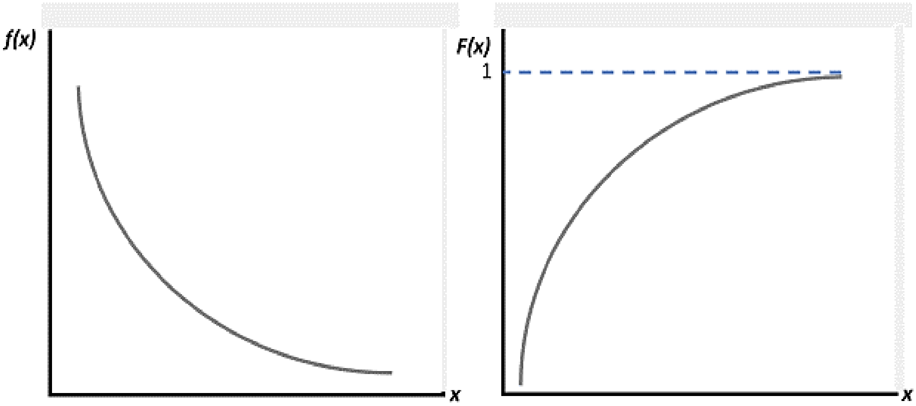
En el ejemplo de las películas usamos números de una serie uniforme para generar una serie discreta conocida, enunciada con el problema, que variaba de entre 0 y 4 y con probabilidades diferentes para cada valor.

Para generar una distribución continua, tenemos el método del siguiente ejemplo:

Disponemos de un servidor capaz de atender, en promedio, 12 clientes/hora (λ). Se supone que hay una variable aleatoria asociada, *T*, que sigue la misma función exponencial con media en 12.

$$f(t) = \frac{1}{\lambda} e^{-\lambda t} = 12 e^{-12t}$$

$$F(T) = P(T \le t) = 1 - e^{-\lambda t} = 1 - e^{-12t}$$



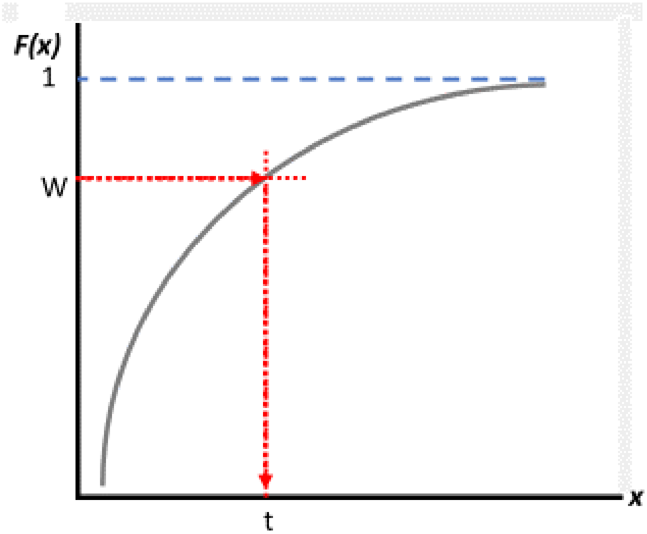
Generaremos un número aleatorio uniforme (***W***), con el cual vamos a usar la ***F(t)*** que nos permitirá hallar el valor de ***t*** que satisfaga ***F(t) = W***. El número ***t*** hallado será el número aleatorio con distribución continua:

$$F(t) = W$$

$$1 - e^{-\lambda t} = W$$

$$e^{-\lambda t} = 1 - W$$

$$t = -\frac{1}{\lambda}\ln(1 - W)$$



También podríamos generar números aleatorios con distribución normal con media μ y desviación estándar σ, usando el método siguiente

- Generamos un número aleatorio uniforme (***W***) entre 0 y 1
- Con ese número ***W*** encontraremos el valor de ***t,*** tal que

$$F(t) = P(n \le t) = W$$

En otros términos, encontraremos el valor de ***t*** para que el área bajo la curva normal a la izquierda de ***t*** sea ***W***. Procedemos así: en una tabla buscaremos el valor de ***Z*** correspondiente y así podemos calcular ***t***:

$$Z = \frac{t - \mu}{\sigma}$$
$$t = \mu + \sigma Z$$

Ejemplo

Generar la llegada de clientes (n) con una distribución normal de media 830 clientes/dia y desviación estándar de 100 clientes:

Supongamos, arbitrariamente, que generamos un número aleatorio, digamos que es ***W*** = 0,1515
Usamos una tabla de Z, o mediante hoja de cálculo, la función

=DISTR.NORM.ESTAND.INV(W)
=INV.NORM.ESTAND(W)

tendremos el valor de ***Z*** buscado, ya que ***W*** representa la probabilidad y en las funciones de arriba lo único que debemos hacer es cambiar ***W*** por el nombre de la celda donde generamos ***W*** con la función ALEATORIO().

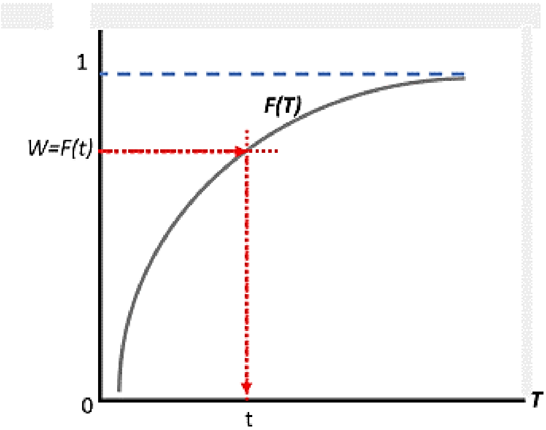
En tablas impresas, como la del apéndice de este texto, deberemos restar a 0,5 el valor generado de ***W*** (si ***W***<0,5) para hallar ***Z*** cambiado de signo o restarle 0,5 a ***W***, si ***W***>0,5 y hallar ***Z*** con su signo.

En este caso ***Z*** = —1,03
$\boldsymbol{t = \mu + \sigma z}$ = 830 + 100 x (—1,03) = 727 clientes

O, directamente, con hoja de cálculo, utilizamos la función:

=DISTR.NORM.INV(W;830;100) =DISTR.NORM.INV(0,1515;830;100)
=INV.NORM(W;830;100)=INV.NORM(0,1515;830;100)

donde, en el grupo de la izquierda, ***W*** es el nombre de la celda (por ejemplo, $C$4) que contiene el número aleatorio (en el ejemplo 0,1515), lo que da 726,997 clientes, con una media de 750 y una desviación estándar de 100.



En general aplicaremos el siguiente teorema: *Si la función de densidad* ***f(t)*** *tiene una función de distribución acumulativa* ***F(t),*** *y* ***W*** *es una variable aleatoria uniformemente distribuida entre 0 y 1, entonces el valor* ***T*** *que satisface* ***W = F(T)*** *es una variable aleatoria cuya distribución acumulativa es* ***F(t).***

Simulando colas con WinQSB

Los problemas de modelos de colas son típicamente aplicables a simulación. Con WinQSB se puede acceder a simulaciones mediante los módulos *Queuing Analysis* y *Queuing System Simulation*. En primer término, veremos el módulo de análisis, que ya hemos utilizado en el capítulo de colas. Para eso, retomamos el problema del alquiler de una central telefónica de 15 líneas sin capacidad de espera cuyos datos repetimos en la siguiente figura:

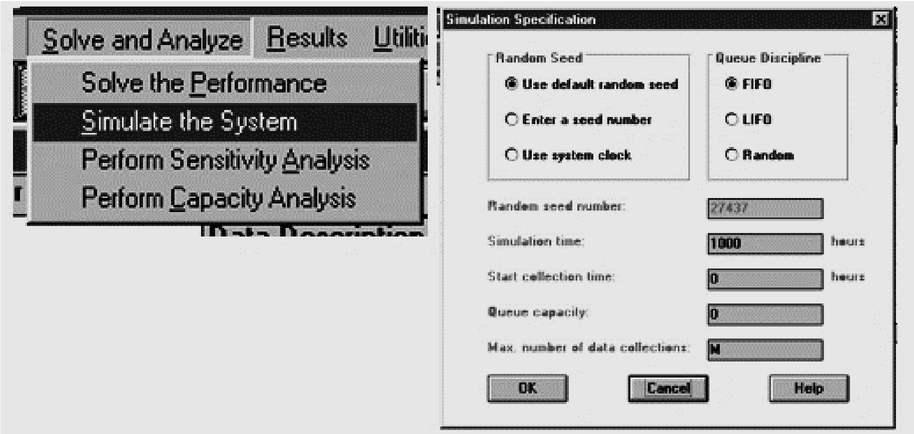
Data Description	ENTRY
Number of servers	15
Service rate (per server per hour)	12
Customer arrival rate (per hour)	150
Queue capacity (maximum waiting space)	0
Customer population	M
Busy server cost per hour	1.25
Idle server cost per hour	1.25
Customer waiting cost per hour	0
Customer being served cost per hour	0
Cost of customer being balked	10
Unit queue capacity cost	

los indicadores hallados por el método analítico son:

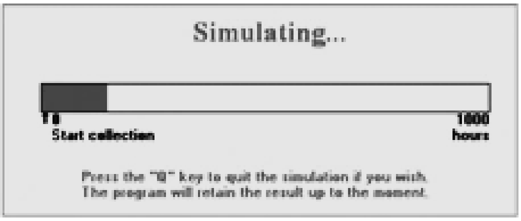
1	System: M/M/15/15	From Formula
2	Customer arrival rate (lambda) per hour =	150,0000
3	Service rate per server (mu) per hour =	12,0000
4	Overall system effective arrival rate per hour =	134,9266
5	Overall system effective service rate per hour =	134,9266
6	Overall system utilization =	74,9593 %
7	Average number of customers in the system (L) =	11,2439
8	Average number of customers in the queue (Lq) =	0
9	Average number of customers in the queue for a busy system (Lb) =	0
10	Average time customer spends in the system (W) =	0,0833 hours
11	Average time customer spends in the queue (Wq) =	0 hour
12	Average time customer spends in the queue for a busy system (Wb) =	0 hour
13	The probability that all servers are idle (Po) =	0,0005 %
14	The probability an arriving customer waits (Pw) or system is busy (Pb) =	10,0489 %
15	Average number of customers being balked per hour =	15,0734
16	Total cost of busy server per hour =	$14,0549
17	Total cost of idle server per hour =	$4,6951
18	Total cost of customer waiting per hour =	$0
19	Total cost of customer being served per hour =	$0
20	Total cost of customer being balked per hour =	$150,7336
21	Total queue space cost per hour =	$0
22	Total system cost per hour =	$169,4836

Ahora vamos a proceder a encontrar los indicadores empleando simulación. Para ello seleccionamos el menú *Solve and analyze*, y allí el submenú *Simulate the system*. Vamos a encontrar un cuadro de diálogo como el de la figura, donde podremos acceder a las siguientes opciones:

1) Elegir la semilla, para lo cual tenemos tres opciones:
   a. la que propone WinQSB (27437, en este caso),
   b. la que el operador incluya manualmente como semilla
   c. la que WinQSB determine usando el reloj de la máquina.
2) Elegir la disciplina de la cola entre FIFO, LIFO o aleatoria
3) Elegir el tiempo de simulación, en este caso aparece por defecto que se va a simular un lapso de 1000 horas de operación de la central
4) Elegir el momento en que se comienza a recolectar datos, en este caso al inicio de la simulación, aunque se puede optar por otro momento, por ejemplo, a las 100 horas del inicio, suponiendo que recién allí hay estabilidad.



Pulsando "OK", WinQSB comenzará una *corrida* de simulación que inicia en cero con la recolección de datos y termina, en este caso, en 1000. Se puede interrumpir el proceso en cualquier momento, presionando Q e ir a la pantalla de resultados.



Cuando finaliza el proceso de simulación, accedemos a una planilla de informes, como la que vemos a continuación y que podremos comparar con la que hemos obtenido mediante proceso en el paso anterior:

1	System: M/M/15/15	From Simulation
2	Customer arrival rate (lambda) per hour =	150,0000
3	Service rate per server (mu) per hour =	12,0000
4	Overall system effective arrival rate per hour =	134,3540
5	Overall system effective service rate per hour =	134,3410
6	Overall system utilization =	74,9942 %
7	Average number of customers in the system (L) =	11,2491
8	Average number of customers in the queue (Lq) =	0
9	Average number of customers in the queue for a busy system (Lb) =	0
10	Average time customer spends in the system (W) =	0,0837 hours
11	Average time customer spends in the queue (Wq) =	0 hour
12	Average time customer spends in the queue for a busy system (Wb) =	0 hour
13	The probability that all servers are idle (Po) =	0,0004 %
14	The probability an arriving customer waits (Pw) or system is busy (Pb) =	10,1604 %
15	Average number of customers being balked per hour =	15,1850
16	Total cost of busy server per hour =	$14,0614
17	Total cost of idle server per hour =	$4,6886
18	Total cost of customer waiting per hour =	$0
19	Total cost of customer being served per hour =	$0
20	Total cost of customer being balked per hour =	$151,8499
21	Total queue space cost per hour =	$0
22	Total system cost per hour =	$170,5999
23	Simulation time in hour =	1000,0000
24	Starting data collection time in hour =	0
25	Number of observations collected =	134341
26	Maximum number of customers in the queue =	0
27	Total simulation CPU time in second =	124,5780

Capítulo 18. Estudios de Casos:

a) Inventarios. b) proyección de Rentabilidad.

Este capítulo está destinado a dos casos concretos en los cuales aplicamos modelos de simulación, desarrollados paso a paso.

En el primero de ellos, abordamos un caso de inventarios con la particularidad de que no se conoce el tiempo guía, ***L***, y tenemos un plazo de obsolescencia, que es el máximo tiempo que el producto puede permanecer en el almacén. Estas particularidades son las que nos dificultan usar los modelos de inventarios que hemos visto anteriormente, por lo tanto, vamos a abordarlo mediante técnicas de simulación.

El segundo es un estudio de rentabilidad en condiciones de incertidumbre con un análisis hecho por simulación en un escenario de demanda desconocida y márgenes de ganancia fluctuantes.

Caso a) Inventarios

Una empresa provee pan de molde a varias bocas de venta. El pan se compra a terceros a $ 1,5 el kg y se almacena en depósito. De ahí el pan es distribuido a los puntos de venta por demanda a $ 2 el kg. La demanda diaria puede ser considerada como buena aproximación a una distribución normal con una media de 750 kg y una desviación estándar de 100.

Cada pedido a los terceros cuesta $100 y cubre los gastos de entrega y de procesamiento.

El pan se pide a los elaboradores por la mañana y se recibe, en el 80% de los casos, también por la mañana del día siguiente, y en el 20% restante a la mañana subsiguiente (dos días después de hecho el pedido)

El costo de mantenimiento del almacén es de H = $ 0,02 por kg y por día. Si el pan permanece más de siete días sin ser distribuido se vende como descarte a $ 0,5 el kg.

Política de inventario

Como tenemos la opción de elegir la política de inventario, que puede ser de revisión continua o de revisión periódica, optaremos por la segunda, aprovechando que, cada mañana debemos satisfacer una demanda y, por tanto, revisar el inventario en ese momento parece lo más apropiado. Lo efectuamos de la siguiente manera:

- Haremos el inventario de pan disponible en la mañana, ***I***, y le sumaremos el pan que fue pedido en días anteriores y que aún no se recibió (***B***).
- Hacemos el nuevo pedido, el correspondiente al día de la fecha. Si la suma de ***I*** + ***B*** está por debajo del nivel mínimo establecido como punto de reorden (***R***), entonces pediremos una cantidad tal que obtengamos un total de ***S*** kg entre inventario y espera de entrega.
  Significa que pediremos $\quad\quad q = S - (I + B)$

El objetivo será determinar los valores de ***R*** y de ***S*** que minimicen el costo de inventario y maximicen las ganancias.

Condiciones de simulación

Para poder desarrollar el modelo de simulación debemos establecer condiciones de partida. Una práctica recomendable es comenzar a trabajar sobre algunos valores concretos para verificar que no se cometen errores y que entendemos la mecánica que vamos a usar.

Establecen los siguientes valores como parámetros del análisis:

El inventario actual más la cantidad pedida y entregas en curso será

$\boldsymbol{S}$ = 4000 kg. ($\boldsymbol{S} = \boldsymbol{q} + (\boldsymbol{I}+\boldsymbol{B})$)

Nivel mínimo de inventario aceptable al inicio del día,

$\boldsymbol{R}$ = 900 kg.

Costo del producto,

$\boldsymbol{C}$ = 1,50 $/kg

Costo del pedido

$\boldsymbol{K}$ = 100 $

Costo de almacén

$\boldsymbol{H}$ = 0,02 $/kg dia

Valor de reventa

$\boldsymbol{V}$ = 0,50 $/kg

Demanda de pan:

$\boldsymbol{D'}$ = 750 kg/dia $\boldsymbol{\sigma}$ = 100

Tiempo guía

$\boldsymbol{L}$ = 1 o 2 días.

Probabilidad $\boldsymbol{P(L)}$ = 0,8 y 0,2, respectivamente. (El proveedor demora 1 día en entregar, aunque el 20% de las veces tarda 2 días)

Ahora veremos un ejemplo del **panorama que se puede encontrar** quien revisa el inventario una mañana cualquiera, la que llamaremos Día 1:

No solo revisa cuanto pan hay, sino cual es la antigüedad de ese pan. En el siguiente ejemplo se supone que no hay pedidos pendientes de entrega y que existe en inventario la siguiente cantidad de pan:

Días de almacén		Inventario (kg)
0	I_0	0
1	I_1	500
2	I_2	300
3	I_3	0
4	I_4	0
5	I_5	100
6	I_6	100

Luego generaremos un número aleatorio uniforme. Si el número es menor a 0,8 establecemos que ***L*** = 1 día, si es igual o mayor a 0,8 entonces ***L*** = 2 días.

Así hemos generado el evento **plazo de entrega del pedido**.

Día 1

Ahora comenzaremos a trabajar sobre el ***Día 1***:

- no se recibe pan porque no hay pedidos pendientes
- el inventario inicial es el que está en la tabla (0+500+300+0+0+100+100 = 1000), que está por encima de los 900 establecidos como piso
- ahora generaremos el evento **demanda del día**. Para ello:
  En primer lugar, generaremos un número aleatorio uniforme y con él encontraremos el ***Z*** que corresponde (el número uniforme es el área a la izquierda de una curva de distribución normal).
  La demanda esperada para ese día es:
  Dem = ***D'*** + ***σ Z*** = 750 + 100Z
  Supongamos que el Z obtenido con GNU es 0,73, entonces la
  Dem = 750 + 100 x 073 = 823 kg
- Actualizamos el inventario: la demanda se cubre con la existencia. Como este es el caso, hay que distribuir la venta: los 823 kg se completan con los 100 kg de pan I_6, más 100 kg de pan I_5 más 300 kg de pan I_2 y 323 kg de pan I_1.
- Ahora hay que actualizar el inventario y llevarlo a condiciones de "día siguiente" (la mañana del día 2):

Dias de almacén		Inicio Dia 1	Entregas	Inicio día 2
0	I_0	0	0	–
1	I_1	500	323	500 – 323=177
2	I_2	300	300	300 – 300=0
3	I_3	0	0	0
4	I_4	0	0	0
5	I_5	100	100	100 – 100=0
6	I_6	100	100	100 – 100=0
7	I_7	0	0	0

Si un kg de pan se vende en boca deja una ganancia de 0,50 $ y si se vende a descarte una pérdida que en términos de "*ganancias*" es de – 1 $

La ganancia neta será:

[kg vendidos en las bocas * ganancia/kg] + [kg descarte * ganancia des/kg] – costo del pedido diario – costo de mantenimiento de inventario diario

En este caso:

Ganancia en bocas: 823 x 0,50 = 411,50

Ganancia descarte: kg de descarte x (– 1) = 0

Costo del pedido: 100 x pedidos del dia 1 = 0

Costo mantenimiento kg de pan con menos de 7 días al fin del día

1 x H = 177 x 0,02 = 3,54

Ganancia neta: 411,50 – 0 – 0 – 3,54 = 407,96

Día 2

- Luego trabajamos sobre el ***Dia 2***:
  No se recibe pan, ya que no hubo pedidos anteriores. El inventario está igual que al fin del día 1: hay un inventario total de 177 kg.

  Deberemos hacer un pedido para que el inventario suba a ***S*** = 4000. El pedido se hace así:

  $$q = S - I + B = 4000 - 177 + 0 = 3823 \text{ kg}$$

- Calculamos el día de llegada del pan: Dia 3 (80%) de probabilidades o Día 4 (20%). Para hacerlo, generamos un número aleatorio (***NA***). Supongamos que obtuvimos un ***NA*** menor que 0,8, por lo tanto, el nuevo pedido llega a inicio del día 3.
- Calculamos la demanda: generamos otro ***NA***. Supongamos que se obtiene que la demanda es de 560 kg de pan.
- Calculamos el cumplimiento o satisfacción de la demanda: No se satisface, pues el número de kg a entregar es mayor que la existencia. Todo el inventario queda en cero.

  Ganancia del día 3: 177 x 0,5 – 0 x 1 – 100 – 0 = – 11,50 $

Día 3

- Comenzamos el cálculo del ***Día 3***:
  Se recibe pan, porque hay un pedido del día anterior:

Días de almacén		Inventario (kg)
0	I_0	3823
1	I_1	0
2	I_2	0
3	I_3	0
4	I_4	0
5	I_5	0
6	I_6	0

El inventario total está por encima de 900, no hay que pedir más pan.

- Calculamos la demanda del día: generamos un ***NA***, supongamos que con él se obtiene una demanda de 607 kg de pan.
- Calculamos la satisfacción o cumplimiento de la demanda. Se cumple, pues se necesitan 607 kg y hay 3823 kg.

Actualizamos el inventario

Dias de almacén		Inicio Dia 3	Entregas	Inicio día 4
0	I_0	3823	607	-
1	I_1	0	0	3823 – 607 =3216
2	I_2	0	0	0
3	I_3	0	0	0
4	I_4	0	0	0
5	I_5	0	0	0
6	I_6	0	0	0
7	I_7	0	0	0

Cálculo de la ganancia neta:

607 x 0,50 – 0 x 1 – 3216 x 0,02 = 239,18 $

Días subsiguientes

De esta manera, podremos calcular una ganancia diaria neta para cada día subsiguiente y, una vez terminado el ensayo, dividiendo por el número de ganancias netas obtenidas encontraremos una ganancia neta promedio, una máxima, una mínima y la desviación estándar.

Obviamente hay que encontrar aquellos valores de ***S*** que maximicen la ganancia neta promedio diaria y hacer simulaciones para determinadas cantidades de días a efectos de elegir la mejor combinación ***R/S***.

Solución alternativa

Una alternativa para solucionar el mismo caso es la siguiente:

- Calculamos el valor de ***R*** como si el inventario fuera determinístico y con ***L*** conocido e igual a 1 día, ***R1***:

  R1 = DL = 750x1 = 750
- Calculamos ***R*** para ***L*** = 2 días, ***R2***

  R2 = 750 x 2 = 1500
- Calculamos la esperanza matemática del punto de reposición:

  E(R) = 750 x 0,8 + 1500 x 0,2 = 900 (que es valor supuesto como inicial)

- Calculamos, con WinQSB, la cantidad ***Q*** suponiendo un problema determinístico con demanda diaria de 750. (debe dar ***Q*** = 2738,6 kg)

- Calculamos el stock de seguridad para $\boldsymbol{\alpha}$ = 0,95,

$$S = Z\sigma = 1{,}645 \times 100 = 164{,}5$$

en un modelo de revisión continua

- Establecemos un primer ***Q*** usando el hallado más ***S*** más la demanda media del tiempo de revisión:

$$750 + Z\sigma + Q = 750 + 164.5 + 2738{,}5 = 3653$$

(redondeamos en 4000)

- Ingresamos valores variando ***R*** y ***S*** hasta encontrar el óptimo.

En www.optimiza.org se puede encontrar un modelo en hoja de cálculo capaz de abordar este caso. Se trata de un desarrollo en dos fases:

1. una hoja de cálculo convencional, donde se cargan los datos de arranque, (como las supuestas en la descripción anterior) y se dan las condiciones iniciales para un simulador: “Condiciones de inicio del tercer día”

2. una macro en VBA que nos permite obtener datos de ganancias medias y estimadores simulando, a partir del tercer día una determinada cantidad de días sucesivos.

CASO DE INVENTARIOS RESUELTO POR SIMULACIÓN - CAPÍTULO 18 / OPTIMIZA11

R	900	kg	CONDICIONES INICIALES (DIA 1)						CONDICIONES INICIALES DIA 2					Inicio
C	1,5	$/kg	Dias de almacen		Inicio D1	Entregas	FinD1	saldo real 1		Inicio D2	Entregas	FinD2	saldo real 2	Dia 3
K	100	$	0	I0	0	0	0	0	0 I0	0	0	0	0	238
H	0,02	$/kg dia	1	I1	500	262	238	238	1 I1	0	0	0	0	0
V	0,5	$/kg	2	I2	300	300	0	0	2 I2	238	238	0	0	0
D	750	kg/dia	3	I3	0	0	0	0	3 I3	0	0	0	0	0
sigma	100		4	I4	0	0	0	0	4 I4	0	0	0	0	0
L	1	dia p=0,8	5	I5	100	100	0	0	5 I5	0	0	0	0	0
	2	dia p=0,2	6	I6	100	100	0	0	6 I6	0	0	0	0	0
descarte	-1		7	I7	0	0	0	0	7 I7	0	0	0	0	0
S	4000	kg									No se cumple demanda			
			Inventario total	I	1000	<<Suma del inventario, sin I7>>				238				0
D1)q=S-(I+B) >>	3000		Inventario en transito	B	0			Inventario en transito		0				3762
D2)q=S-(I+B) >>	3762		I+B		1000			I+B		238				3762
			Demanda espera		762			demanda dia 3		799	Pedido para D4		demanda dia 4	696
			Pedidos pD2		0			Pedidos pD3		3762	0			
			Ganancia dia 1		486			Ganancia dia 2		124				
D3)q=S-(I+B)	4000		[Suma "aporte real"*gananciaV] + [I7*descarte] + [si hay mercaderia en tránsito=costo compra, si no 0] + [saldos de pan*co											

La misma imagen en varios detalles:

CASO DE

NVENTARIOS RESUELTO POR SIMULACIÓN - CAPÍTULO 18 / OPTIMIZA11

R	900	kg
C	1,5	$/kg
K	100	$
H	0,02	$/kg dia
V	0,5	$/kg
D	750	kg/dia
sigma	100	
L	1	dia p=0,8
	2	dia p=0,2
descarte	-1	
S	4000	kg
D1)q=S-(I+B) >>	3000	
D2)q=S-(I+B) >>	3762	
D3)q=S-(I+B)	4000	

CONDICIONES INICIALES (DIA 1)						CONDICIONES INICIALES DIA 2					Inicio
Dias de almacen		Inicio D1	Entregas	FinD1	saldo real 1		Inicio D2	Entregas	FinD2	saldo real 2	Dia 3
0	I0	0	0	0	0	0 I0	0	0	0	0	238
1	I1	500	262	238	238	1 I1	0	0	0	0	0
2	I2	300	300	0	0	2 I2	238	238	0	0	0
3	I3	0	0	0	0	3 I3	0	0	0	0	0
4	I4	0	0	0	0	4 I4	0	0	0	0	0
5	I5	100	100	0	0	5 I5	0	0	0	0	0
6	I6	100	100	0	0	6 I6	0	0	0	0	0
7	I7	0	0	0	0	7 I7	0	0	0	0	0

No se cumple demanda

				No se cumple demanda		
Inventario total I	1000	<<Suma del inventario, sin I7>>	238			0
Inventario en transito B	0	Inventario en transito	0			3762
I+B	1000	I+B	238			3762
Demanda espera	762	demanda dia 3	799	Pedido para D4	demanda dia 4	696
Pedidos pD2	0	Pedidos pD3	3762	0		
Ganancia dia 1	486	Ganancia dia 2	124			

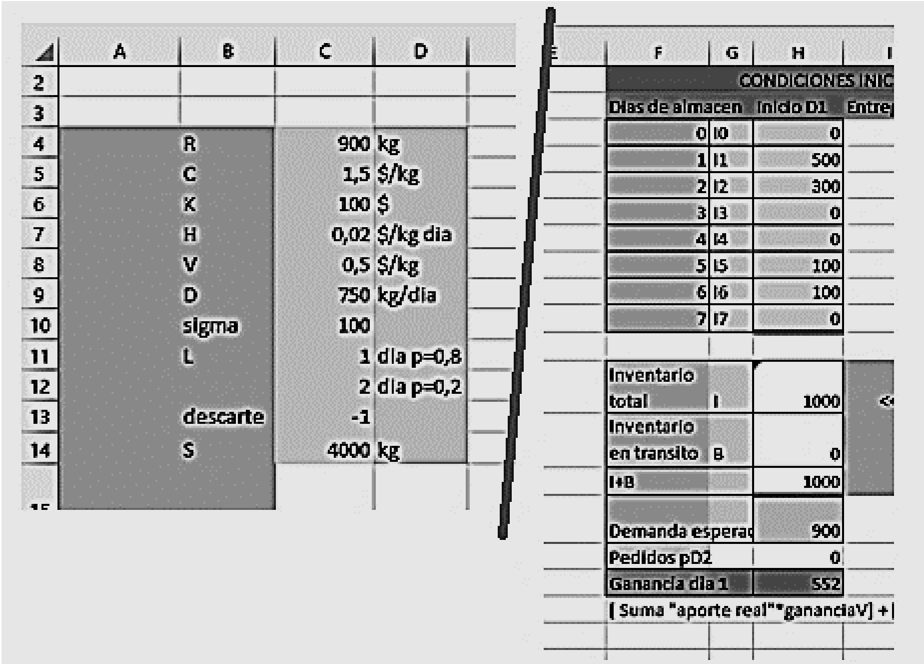
[Suma "aporte real"*gananciaV] + [I7*descarte] + [si hay mercaderia en tránsito=costo compra, si no 0] + [saldos de pan*co

D1)q=S-(I+B) >>	3000
D2)q=S-(I+B) >>	3762
D3)q=S-(I+B)	4000
Días simulados	574
ganancia máxima	600
ganancia mínima	-12
ganancia media diaria	309
desvest	80
S ensayado	3500
R ensayado	900

De esta manera, se pueden ingresar manualmente datos de partida, como ***S***, ***R***, stock de partida, etc. y obtener condiciones de rutina para el tercer día.

También deberemos indicar un stock inicial agrupado por días de antigüedad.

De esta manera, tendremos el punto de partida una macro que nos va a entregar un reporte con un listado de días, el costo medio de inventario y los parámetros estadísticos necesarios. Las figuras que acompañan son capturas de pantalla del libro mencionado. Una vez más, cada uno puede hacerlo en forma diferente y en cada ensayo los resultados mostrados serán diferentes al de las figuras, pues dependen de los números aleatorios generados.



Caso b) Rentabilidad proyectada

RECOMENDAMOS AVANZAR EN LA LECTURA DE ESTE CASO REPITIENDO LOS PASOS EN UN LIBRO DE CÁLCULO.

La empresa ***Telefonía TAC*** desea lanzar una campaña de ventas, a una cartera de clientes corporativos, usando llamadas telefónicas personalizadas.

Debido a que los resultados y datos son inciertos, la empresa desea comprender mejor el comportamiento de la estrategia de ventas y pronosticar sus resultados económicos. Para lograrlo piensa utilizar un modelo de simulación que permita ver la evolución del negocio a medio plazo.

La información disponible y el análisis preliminar de la situación son los siguientes:

- Utilizaremos simulación con el método de Montecarlo y con el objetivo de evaluar probabilidades de rentabilidad y de riesgos.
- El modelo matemático básico empleado será

“***Beneficio = ingreso – gastos***”,
que, aunque es muy sencillo sus parámetros son inciertos, en este caso.

Desarrollo y búsqueda de datos

En primer lugar, deberemos trabajar con el término “***Ingreso***”, cuyos parámetros son:

- la cantidad de ventas (***Q***)
- ingreso por cada venta (***P***), que es una ganancia conocida en cada caso respecto a los costos de todos los estados previos a la venta en análisis, aunque es variable en términos de la manera en que el vendedor cierra cada operación con cada cliente.

Queda entonces la expresión:

Ingreso=Q*P

El rango de variación de ***P*** es de 47 $ a 53 $

$$47\$ \leq \boldsymbol{P} \leq 53\$$$

Para estimar el valor de las ventas efectuadas utilizaremos la experiencia en *purchasing leads* de la empresa. Se analiza que es razonable usar una tasa de conversión (éxito) (***R***) que varía del 1 al 5% aplicada sobre una cartera de clientes potenciales que oscila entre 1200 y 1800 en función de la posibilidad de contactos que se pueden efectivizar en un mes.

Entonces, queda que

Q=R*L

Para

$$1{,}0\% \leq \boldsymbol{R} \leq 5{,}0\%$$

Y, además

$$1200 \leq \boldsymbol{L} \leq 1800$$

Quedando, en definitiva:

Ingreso=R*L*P

Ahora debemos estimar el miembro “Gastos” de nuestra ecuación simplificada. Los calcularemos en base a costos fijos (***H***) que tiene el equipo de ventas y un costo que dependerá de cada contacto (exitoso o no) con los potenciales clientes (***C***).

Gastos=H+C*L

En base a estimaciones históricas, se supone que los costos unitarios de contacto con un cliente dependen de varios factores, tales como búsqueda, necesidad de repetir el intento de contacto, etc. Por eso, están sujeto a incertidumbre y pueden variar:

$$0{,}20 \leq \boldsymbol{C} \leq 0{,}80$$

Finalmente, el modelo general de partida queda expresado de la siguiente manera:

Beneficio=R*L*P-(H+C*L)

Q**ue podemos considerar como una función **y=f(x)

$$\boldsymbol{y = x_1 x_2 x_3 - (H + x_4 x_2)\,)}$$

Generación de valores aleatorios en cada una de las variables

El primer paso será generar valores aleatorios dentro de los rangos permitidos que satisfagan cada una de las variables, según anotamos en la siguiente tabla:

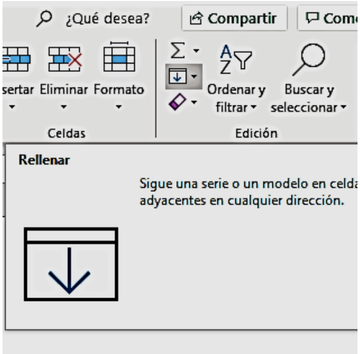
Variable	Valor medio	Mínimo	Máximo
Tasa de conversión R	**0,03**	0,01	0,05
Cartera mensual de clientes L	**1500**	1200	1800
Beneficio por venta P	**50**	47	53
Costo de contacto C	**0,50**	0,20	0,80
Costo fijo mensual H	**800**	800	800

En nuestro caso, hemos construido un modelo completo en hoja de cálculo, que es que describimos a continuación paso a paso. Una vez más recordamos que cada uno puede desarrollarlo de diferentes maneras.

Habilitamos dos hojas en un libro. La primera será destinada a los datos del problema (tabla de arriba) y a las salidas del simulador. La segunda al simulador propiamente dicho.

Hoja 1, datos.

En la primera parte de esta Hoja, como vemos en las figuras, destinaremos celdas a los límites de variación de cada uno de los parámetros (columnas “MINIMO” y “MAXIMO”) y en la columna “MEDIA” solamente escribiremos la función Promedio. Por ejemplo, en la celda C5 irá: “=PROMEDIO(D5;E5)”. Esta celda se puede “estirar o rellenar” hasta la C9 y quedará toda la columna completa (figura inferior)



BENEFICIOS = L*R*P - (H+L*C)

	MEDIA	MINIMO	MAXIMO	P()
L - N° Clientes	=PROMEDIO(D5:E5)	1200	1800	unif
C - Costo del lead	=PROMEDIO(D6:E6)	0,2	0,8	unif
R - Tasa de concreción	=PROMEDIO(D7:E7)	0,01	0,05	unif
P - Precio de venta	=PROMEDIO(D8:E8)	47	53	unif
H - Gastos Fijos	=PROMEDIO(D9:E9)	800	800	unif

queda, una vez finalizada, con el aspecto siguiente:

BENEFICIOS = L*R*P - (H+L*C)

	MEDIA	MINIMO	MAXIMO	P()
L - N° Clientes	1500	1200	1800	unif
C - Costo del lead	0,5	0,2	0,8	unif
R - Tasa de concreción	0,03	0,01	0,05	unif
P - Precio de venta	50	47	53	unif
H - Gastos Fijos	800	800	800	unif

Mas adelante, volveremos a esta hoja para agregarle el informe de resultados.

Hoja 2

Ahora construiremos en esta nueva hoja el simulador propiamente dicho. Vamos a destinar una columna a cada una de las variables. De esta manera, completaremos — para cada una de las columnas — el renglón 2:

- **Columna A**, variable ***L***, expresión en **A2**:

  "=ALEATORIO.ENTRE(Hoja1!$D$5;Hoja1!$E$5)"

  (Estas son referencias "ancladas" con los signos $ de la tabla de la Hoja1. De esta manera, cualquier dato que se pretenda cambiar en el simulador en el futuro se podrá hacer solamente modificando la tabla de la Hoja1 y no todo el simulador)

- **Columna B**, variable ***C***, expresión en **B2**:

  "=Hoja1!$D$6 + ALEATORIO()*(Hoja1!$E$6—Hoja1!$D$6)"

  Ídem.

- **Columna C**, variable ***R***, expresión en **C2**:

  "=Hoja1!$D$7 + ALEATORIO()*(Hoja1!$E$7—Hoja1!$D$7)".

  Ídem.

- **Columna D**, variable ***P***, expresión en **D2**:

  "=ALEATORIO.ENTRE(Hoja1!$D$8;Hoja1!$E$8)"

  Idem

- **Columna E**, valor fijo de ***H*** (800). Lo consideramos una variable y lo dejamos expresado como tal. Si en el futuro resulta conveniente usar ***H*** como variable no debemos modificar nada. Mientras tanto, al ser idénticos los valores *máximo* y *mínimo*, se comporta como constante. La expresión en **E2 es:**

  "=ALEATORIO.ENTRE(Hoja1!$D$9—Hoja1!$E$9)"

- **Columna G**, valor de ***Ganancia***, expresión en **G2**:

  "=C2*A2*D2-(E2+C2*A2)"

Tendremos así el encabezado que vemos en la figura siguiente, parte superior, con sus fórmulas en la parte inferior (obviamente los valores numéricos en la parte superior son aleatorios y diferentes en cada caso):

D2 | =ALEATORIO.ENTRE(Hoja1!$D$8;Hoja1!$E$8)

	A	B	C	D	E	F	G	H
1	L	C	R	P	H		BENEFICIO	
2	1213	0,74192845	0,03349769	53	800		453,57	

1	L	C	R
2	=ALEATORIO.ENTRE(Hoja1!$D$5;Hoja1!$E$5)	=Hoja1!$D$6+ALEATORIO()*(Hoja1!$E$6-Hoja1!$D$6)	=Hoja1!$D$7+ALEATORIO()*(Hoja1!$E$7-Hoja1!$D$7)

De esta manera el renglón 2 será un mes. Para realizar la simulación simplemente arrastraremos estos valores de manera tal de realizar 5000 ensayos o meses, que podremos repetir cada vez que pulsemos F9.

	A	B	C	D	E	F	G
1	L	C	R	P	H		BENEFICIO
2	1578	0,62907189	0,02491738	52	800		251,94
3	1406	0,68270535	0,02140066	49	800		-285,51
4	1708	0,24772805	0,02563142	52	800		1053,36
5	1494	0,56367594	0,04393209	49	800		1573,96
6	1340	0,52327067	0,01957734	51	800		-163,27
7	1409	0,59565517	0,01801116	51	800		-345,01
8	1496	0,54518169	0,03822182	51	800		1300,58
9	1560	0,3382656	0,03791469	51	800		1688,80
10	1648	0,37613297	0,03792065	49	800		1642,30
11	1665	0,22276555	0,03780318	53	800		2165,04
12	1400	0,64887032	0,01534645	49	800		-655,65
13	1406	0,21152567	0,01193616	49	800		-275,08
14	1649	0,3544674	0,03186322	49	800		1190,06
15	1605	0,39559657	0,02048365	47	800		110,25
16	1385	0,478535	0,04495116	47	800		1463,32
17	1722	0,37231466	0,0137758	52	800		-207,59
18	1399	0,58680636	0,04972456	47	800		1648,60
19	1428	0,29480308	0,01828415	48	800		32,29
20	1710	0,42152812	0,02585568	48	800		601,42
21	1606	0,23537356	0,03571321	53	800		1861,83
22	1560	0,36676611	0,0240014	48	800		425,07

La siguiente tarea, será efectuar la contabilidad y obtener los datos estadísticos para hacer un informe completo. Comenzaremos por preparar todo a fin de realizar un gráfico de barras. Para hacer esto, en la misma Hoja2 vamos a agregar los siguientes elementos:

H	I	J	K	L
	MINIMO	-1233,95	J2 = MIN(G:G)	
	MAXIMO	3241,57	J3 = MAX(G:G)	
	PUNTOS	6100	J4 = CONTAR(G:)	
	HISTOGRAMA 40 COLUMNAS			
	LIM SUP	FRECUENCIA	FREC.ACUM	
	-1233,9	1	1	
	-1122,1	19	20	
	-1010,2	48	68	
	-898,3	77	145	
	-786,4	124	269	
	-674,5	137	406	
	-562,6	158	564	

Cada una de las celdas tienen las siguientes funciones (también aparecen explicadas en la figura en la columna K):

Celda **J2**: “=MIN(G:G)”
Celda **J3**: “=MAX(G:G)”
Celda **J4**: “=CONTAR(G:G)”

El objetivo es disponer en un lugar visible y fijo los valores de ganancias mínima, máxima y el tamaño del ensayo (en el caso de la figura se han realizado 6100 ensayos o meses).

En un segundo grupo de celdas, en la misma columna “I”, comenzaremos a preparar la tabla de datos del histograma de frecuencias. Lo haremos de la siguiente manera:

COLUMNA I
La primera celda de la columna “I” bajo la etiqueta “LIM SUP” copia el valor de J2, (Mínimo) y es el extremo izquierdo del eje de abscisas. (en la figura, es la celda I11)

En este caso decidimos dividir dicho eje (Rango[16]) en 40 valores (intervalos), por lo tanto, la segunda celda deberá tener como función el incremento desde la celda anterior (I11). El ancho del inervalo, a su vez será la diferencia entre el máximo y el mínimo que calculamos más arriba, (la diferencia entre J3 y J2 o Rango), dividida entre 40. Es la siguiente expresión:

“=I11 + ($J$3 - $J$2)/40”

Se lee de esta manera: “*al valor de la celda anterior le sumamos la 40ava parte de la diferencia entre el mayor y el menor valor de la ganancia*”

[16] Recordemos que el Rango es la diferencia entre el máximo valor y el mínimo valor de los datos observados. La clase o ancho del intervalo será el cociente entre el Rango y el número de clases que deseamos obtener, en este ejemplo, 40

Las demás celdas de esa columna se rellenan arrastrando esta segunda que acabamos de hacer, hasta totalizar las 40 deseadas. Entonces, cada renglón se refiere al renglón anterior más el rango dividido 40.

COLUMNA J

Es la columna "Frecuencia" de la figura. Allí vamos a colocar la cantidad de observaciones que hemos hecho dentro de cada clase. Lo haremos de la siguiente manera: seleccionamos previamente todo el rango de 40 celdas que aún está vacío: desde J11 hasta J50. Con ese conjunto de celdas seleccionado escribimos la siguiente función (o usamos el asistente de funciones ***fx***):

"{=FRECUENCIA(G:G;I11:I52)}"

donde G:G es el rango donde están los datos e I11:I52 es donde están las referencias. Observemos que hemos escrito entre corchetes la función. Esto significa que es una de las funciones matriciales que tienen las hojas de cálculo. Al escribirla asignándoles un conjunto de celdas vacías estamos indicando que en ese espacio se debe escribir la matriz resultante (en este caso, un vector columna).

MUY IMPORTANTE: como toda función matricial, no debe aceptarse pulsando "Enter" (ni con el botón "Aceptar" si usamos el asistente) como en las demás funciones. **Debemos pulsar la combinación de teclas *CTR+SHIFT+ENTER* en lugar de Aceptar)**

COLUMNA K

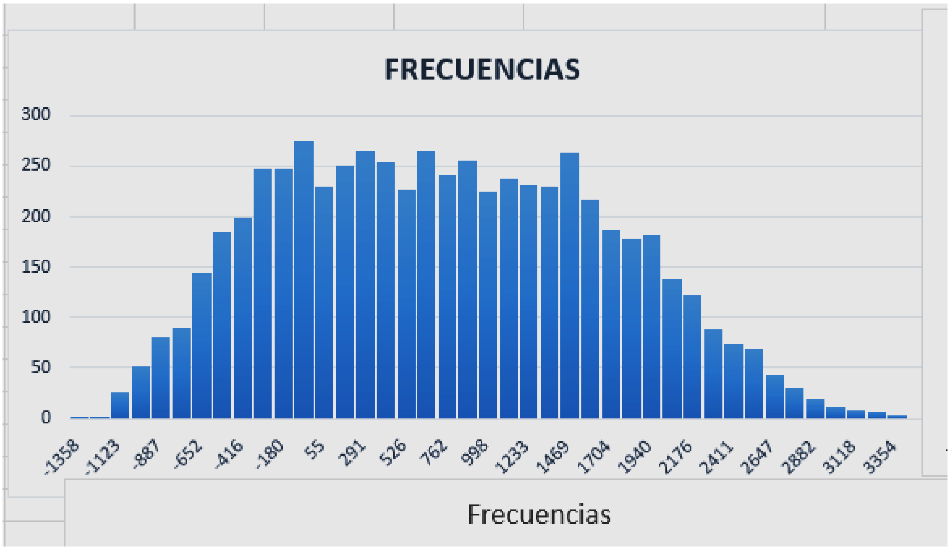
Destinaremos esta columna a anotar las frecuencias acumuladas, desde K11 a K52. Para completarla deberemos escribir en la primera, K11 la expresión "=J11". En la segunda, K12, la expresión:

"=K11+J12"

y luego rellenar, arrastrando, el resto hasta completar las 40 celdas.

ANÁLISIS ESTADÍSTICO

En este momento ya estamos en condiciones de hacer un análisis estadístico de los resultados. Vamos a comenzar con un histograma a partir de la tabla que acabamos de construir. Para lograrlo, simplemente seleccionaremos las dos primeras columnas de la tabla que acabamos de construir y usando el asistente de gráficos, construimos el histograma de frecuencias similar al de la figura siguiente:



El siguiente paso será agregar a la hoja una tabla de resultados de análisis. Para eso utilizamos las celdas desde N2 hasta N10, para escribir las funciones que describimos en la columna O. Usamos la columna M para los nombres. La figura siguiente nos muestra el resultado:

J	K	L	M	N	O	P	Q
			ANALISIS ESTADÍSTICO DE LA SIMULACIÓN				
-1358,33	J2 = MIN(G:G)		TAMAÑO	6100,00	N2 = J4		
3353,63	J3 = MAX(G:G)		PROMEDIO	691,26	N3 = PROMEDIO (G:G)		
6100	J4 = CONTAR(G:)		MEDIANA	651,62	N4 = MEDIANA(G:G)		
			DESV EST	929,73	N5 = DESVEST.M(G:G)		
			COEF ASIM	0,18	N6 = COEFICIENTE.ASIMETRIA (G:G)		
			CURTOSIS	-0,77	N7 = CURTOSIS (G:G)		
A 40 COLUMNAS			Prob.Ganar	0,7466	N9 =SUMAR.SI(I11:I52;">0";J11:J52)/(SUMA(J11:J52))		
CUENCIA	FREC.ACUM		Prob.Perder	0,2534	N10 = SUMAR.SI(I11:I52;"<0";J11:J52)/(SUMA(J11:J52))		
1	1		1,0000				
2	3						
26	29				FRECUENCIAS		

Lo que hacemos es un informe, en el cual aparece, como vemos arriba, la cantidad de ensayos, la media, la mediana, la desviación estándar de la muestra, el coeficiente asimetría y la curtosis.

En las celdas N9 y N10 contamos cuántas frecuencias de abscisa negativa y cuántas positivas hay. Esas sumas, divididas entre la cantidad de casos darán la medida de la superficie bajo la curva de cada grupo, que es equivalente a la probabilidad de ocurrencia. ¿Cómo lo logramos? Lo obtenemos sumando el número de ensayos que dieron ganancia (que están en la columna J entre las celdas 11 y 52) lo que se encuentra si se cumple con la condición de que la columna I, desde 11 hasta 52 tenga valores positivos. El resultado lo dividimos entre la suma algebraica de todas las ganancias, positivas y negativas:

“=SUMAR.SI(I11:52;”<0”;J11:J52/(SUMA(J11:J52))”

sintaxis de SUMAR.SI:

=SUMAR.SI(rango a comparar; valor de comparación; rango a sumar si la comparación es VERDADERA)

En otras palabras, lo que hemos hecho es averiguar que fracción del área del histograma corresponde a valores de ganancia positiva. El total del área es equivalente a uno. En el renglón siguiente usamos la misma expresión para contar los negativos. Sumando ambas deberíamos obtener un uno, lo cual es una verificación de que no hay errores.

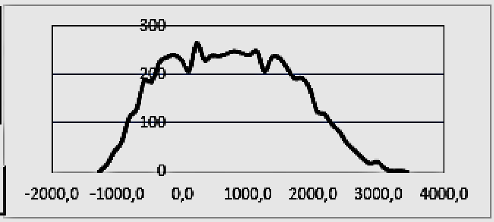
La siguiente figura nos muestra el aspecto de la hoja1 a la que volvemos y agregamos los datos de informe y un pequeño gráfico adicional. Cada vez que pulsemos, desde esa hoja, la tecla “F9” obtendremos un nuevo ensayo. En el caso de la figura le hemos agregado un botón de comando que, simplemente, equivale a pulsar F9.

BENEFICIOS = L*R*P - (H+L*C)

	MEDIA	MINIMO	MAXIMO	P()
L - N° Clientes	1500	1200	1800	unif
C - Costo del lead	0,5	0,2	0,8	unif
R - Tasa de concreción	0,03	0,01	0,05	unif
P - Precio de venta	50	47	53	unif
H - Gastos Fijos	800	800	800	unif

BENEFICIOS MEDIOS	700
BENEFICIOS SIMULADOS	688,64
Probabilidad ganar	0,73

Simular



En ***www.optimiza.org*** se encuentra una hoja de cálculo con este modelo. (debes seguir el siguiente enlace:
https://drive.google.com/open?id=1GzrlnJtZpO2JpxmnGl_ClZpZXOEvd0ok)

Válido solo para estudiantes registrados en la asignatura.

EL RUIDO DE UN TRUENO

[Cuento. Texto completo]

Ray Bradbury[17]

El anuncio en la pared parecía temblar bajo una móvil película de agua caliente. Eckels sintió que parpadeaba, y el anuncio ardió en la momentánea oscuridad:

SAFARI EN EL TIEMPO S.A. SAFARIS A CUALQUIER AÑO DEL PASADO. USTED ELIGE EL ANIMAL NOSOTROS LO LLEVAMOS ALLÍ, USTED LO MATA.

Una flema tibia se le formó en la garganta a Eckels. Tragó saliva empujando hacia abajo la flema. Los músculos alrededor de la boca formaron una sonrisa, mientras alzaba lentamente la mano, y la mano se movió con un cheque de diez mil dólares ante el hombre del escritorio.

–¿Este safari garantiza que yo regrese vivo?

–No garantizamos nada –dijo el oficial–, excepto los dinosaurios. –Se volvió–. Este es el señor Travis, su guía safari en el pasado. Él le dirá a qué debe disparar y en qué momento. Si usted desobedece sus instrucciones, hay una multa de otros diez mil dólares, además de una posible acción del gobierno, a la vuelta.

Eckels miró en el otro extremo de la vasta oficina la confusa maraña zumbante de cables y cajas de acero, y el aura ya anaranjada, ya plateada, ya azul. Era como el sonido de una gigantesca hoguera donde ardía el tiempo, todos los años y todos los calendarios de pergamino, todas las horas apiladas en llamas. El roce de una mano, y este fuego se volvería maravillosamente, y en un instante, sobre sí mismo. Eckels recordó las palabras de los anuncios en la carta. De las brasas y cenizas, del polvo y los carbones, como doradas salamandras, saltarán los viejos años, los verdes años; rosas endulzarán el aire, las canas se volverán negro ébano, las arrugas desaparecerán. Todo regresará volando a la semilla, huirá de la muerte, retornará a sus

[17] [17] Ray Bradbury, escritor nacido en 1920 en Estados Unidos y fallecido en 2012. Escribió cuentos y novelas de diversos géneros, desde el policial hasta el realista y costumbrista, pero se le conoce como un escritor clásico de la ciencia ficción por sus obras de la década de 1950, como "Crónicas Marcianas, serie de relatos cortos acerca de los seis primeros viajes hacia Marte y su posterior colonización y "Fahrenheit 451", novela distópica llevada al cine en varias oportunidades, la última en 2018. Trabajó como argumentista y guionista en numerosas películas y series de televisión, entre las que cabe destacar su colaboración con John Huston en la adaptación de "Moby Dick" para la película que este dirigió en 1956. Además, escribió poemas y ensayos. Hay un asteroide llamado (9766) *Bradbury* en su honor. El presente relato se publicó en Argentina en un libro de cuentos llamado "Las doradas manzanas del sol" de la colección Minotauro, año 2006, ISBN: 978-950-547-143-0 y fue escrito entre 1952 y 1953. **Se incluye en el presente texto solamente con fines didácticos y académicos.**

principios; los soles se elevarán en los cielos occidentales y se pondrán en orientes gloriosos, las lunas se devorarán al revés a sí mismas, todas las cosas se meterán unas en otras como cajas chinas, los conejos entrarán en los sombreros, todo volverá a la fresca muerte, la muerte en la semilla, la muerte verde, al tiempo anterior al comienzo. Bastará el roce de una mano, el más leve roce de una mano.

–¡Infierno y condenación! – murmuró Eckels con la luz de la máquina en el rostro delgado–. Una verdadera máquina del tiempo. –Sacudió la cabeza–. Lo hace pensar a uno. Si la elección hubiera ido mal ayer, yo quizá estaría aquí huyendo de los resultados. Gracias a Dios ganó Keith. Será un buen presidente.

–Sí – dijo el hombre detrás del escritorio–. Tenemos suerte. Si Deutscher hubiese ganado, tendríamos la peor de las dictaduras. Es el antídoto, militarista, anticristo, antihumano, antintelectual. La gente nos llamó, ya sabe usted, bromeando, pero no enteramente. Decían que si Deutscher era presidente, querían ir a vivir a 1492. Por supuesto, no nos ocupamos de organizar evasiones, sino safaris. De todos modos, el presidente es Keith. Ahora su única preocupación es...

Eckels terminó la frase:

–Matar mi dinosaurio.

–Un *Tyrannosaurus rex*. El lagarto del Trueno, el más terrible monstruo de la historia. Firme este permiso. Si le pasa algo, no somos responsables. Estos dinosaurios son voraces.

Eckels enrojeció, enojado.

–¿Trata de asustarme?

–Francamente, sí. No queremos que vaya nadie que sienta pánico al primer tiro. El año pasado murieron seis jefes de safari y una docena de cazadores. Vamos a darle a usted la más extraordinaria emoción que un cazador pueda pretender. Lo enviaremos sesenta millones de años atrás para que disfrute de la mayor y más emocionante cacería de todos los tiempos. Su cheque está todavía aquí. Rómpalo.

El señor Eckels miró el cheque largo rato. Se le retorcían los dedos.

–Buena suerte –dijo el hombre detrás del mostrador–. El señor Travis está a su disposición.

Cruzaron el salón silenciosamente, llevando los fusiles, hacia la Máquina, hacia el metal plateado y la luz rugiente.

Primero un día y luego una noche y luego un día y luego una noche, y luego día–noche–día–noche–día. Una semana, un mes, un año, ¡una década! 2055, 2019, ¡1999! ¡1957! ¡Desaparecieron! La Máquina rugió. Se pusieron los cascos de oxígeno y probaron los intercomunicadores. Eckels se balanceaba en el asiento almohadillado, con el rostro pálido y duro. Sintió un temblor en los brazos y bajó los ojos y vio que sus manos apretaban el fusil. Había otros cuatro hombres en esa máquina. Travis, el jefe del safari, su asistente, Lesperance, y dos otros cazadores, Billings y Kramer. Se miraron unos a otros y los años llamearon alrededor.

–¿Estos fusiles pueden matar a un dinosaurio de un tiro? –se oyó decir a Eckels.

–Si da usted en el sitio preciso –dijo Travis por la radio del casco–. Algunos dinosaurios tienen dos cerebros, uno en la cabeza, otro en la columna espinal. No les tiraremos a éstos, y tendremos más probabilidades. Aciérteles con los dos primeros tiros a los ojos, si puede, cegándolo, y luego dispare al cerebro.

La máquina aulló. El tiempo era una película que corría hacia atrás. Pasaron soles, y luego diez millones de lunas.

–Dios santo –dijo Eckels–. Los cazadores de todos los tiempos nos envidiarían hoy. África al lado de esto parece Illinois.

El sol se detuvo en el cielo.

La niebla que había envuelto la Máquina se desvaneció. Se encontraban en los viejos tiempos, tiempos muy viejos en verdad, tres cazadores y dos jefes de safari con sus metálicos rifles azules en las rodillas.

–Cristo no ha nacido aún –dijo Travis–. Moisés no ha subido a la montaña a hablar con Dios. Las pirámides están todavía en la tierra, esperando. Recuerde que Alejandro, Julio César, Napoleón, Hitler... no han existido.

Los hombres asintieron con movimientos de cabeza.

–Eso –señaló el señor Travis– es la jungla de sesenta millones dos mil cincuenta y cinco años antes del presidente Keith.

Mostró un sendero de metal que se perdía en la vegetación salvaje, sobre pantanos humeantes, entre palmeras y helechos gigantescos.

–Y eso –dijo– es el Sendero, instalado por Safari en el Tiempo para su provecho. Flota a diez centímetros del suelo. No toca ni siquiera una brizna, una flor o un árbol. Es de un metal antigravitatorio. El propósito del Sendero es impedir que toque usted este mundo del pasado de algún modo. No se salga del Sendero. Repito. No se salga de él. ¡Por ningún motivo! Si se cae del Sendero hay una multa. Y no tire contra ningún animal que nosotros no aprobemos.

–¿Por qué? –preguntó Eckels. Estaban en la antigua selva. Unos pájaros lejanos gritaban en el viento, y había un olor de alquitrán y viejo mar salado, hierbas húmedas y flores de color de sangre.

–No queremos cambiar el futuro. Este mundo del pasado no es el nuestro. Al gobierno no le gusta que estemos aquí. Tenemos que dar mucho dinero para conservar nuestras franquicias. Una máquina del tiempo es un asunto delicado. Podemos matar inadvertidamente un animal importante, un pajarito, un coleóptero, aun una flor, destruyendo así un eslabón importante en la evolución de las especies.

–No me parece muy claro –dijo Eckels.

–Muy bien –continuó Travis–, digamos que accidentalmente matamos aquí un ratón. Eso significa destruir las futuras familias de este individuo, ¿entiende?

–Entiendo.

–¡Y todas las familias de las familias de ese individuo! Con sólo un pisotón aniquila usted primero uno, luego una docena, luego mil, un millón, ¡un billón de posibles ratones!

–Bueno, ¿y eso qué? –inquirió Eckels.

–¿Eso qué? –gruñó suavemente Travis–. ¿Qué pasa con los zorros que necesitan esos ratones para sobrevivir? Por falta de diez ratones muere un zorro. Por falta de diez zorros, un león muere de hambre. Por falta de un león, especies enteras de insectos, buitres, infinitos billones de formas de vida son arrojadas al caos y la destrucción. Al final todo se reduce a esto: cincuenta y nueve millones de años más tarde, un hombre de las cavernas, uno de la única docena que hay en todo el mundo, sale a cazar un jabalí o un tigre para alimentarse. Pero usted, amigo, ha aplastado con el pie a todos los tigres de esa zona al haber pisado un ratón. Así que el hombre de las cavernas se muere de hambre. Y el hombre de las cavernas, no lo olvide, no es un hombre que pueda desperdiciarse, ¡no! Es toda una futura nación. De él nacerán diez hijos. De ellos nacerán cien hijos, y así hasta llegar a nuestros días. Destruya usted a este hombre, y destruye usted una raza, un pueblo, toda una historia viviente. Es como asesinar a uno de los nietos de Adán. El pie que ha puesto usted sobre el ratón desencadenará así un terremoto, y sus efectos sacudirán nuestra tierra y nuestros destinos a través del tiempo, hasta sus raíces. Con la muerte de ese hombre de las cavernas, un billón de otros hombres no saldrán nunca de la matriz. Quizás Roma no se alce nunca sobre las siete colinas. Quizá Europa sea para siempre un bosque oscuro, y sólo crezca Asia saludable y prolífica. Pise usted un ratón y aplastará las pirámides. Pise un ratón y dejará su huella, como un abismo en la eternidad. La reina Isabel no nacerá nunca, Washington no cruzará el Delaware, nunca habrá un país llamado Estados Unidos. Tenga cuidado. No se salga del Sendero. ¡Nunca pise afuera!

–Ya veo –dijo Eckels–. Ni siquiera debemos pisar la hierba.

–Correcto. Al aplastar ciertas plantas quizá sólo sumemos factores infinitesimales. Pero un pequeño error aquí se multiplicará en sesenta millones de años hasta alcanzar proporciones extraordinarias. Por supuesto, quizá nuestra teoría esté equivocada. Quizá nosotros no podamos cambiar el tiempo. O tal vez sólo pueda cambiarse de modos muy sutiles. Quizá un ratón muerto aquí provoque un desequilibrio entre los insectos de allá, una desproporción en la población más tarde, una mala cosecha luego, una depresión, hambres colectivas, y, finalmente, un cambio en la conducta social de alejados países. O aun algo mucho más sutil. Quizá sólo un suave aliento, un murmullo, un cabello, polen en el aire, un cambio tan, tan leve que uno podría notarlo sólo mirando de muy cerca. ¿Quién lo sabe? ¿Quién puede decir realmente que lo sabe? No nosotros. Nuestra teoría no es más que una hipótesis. Pero mientras no sepamos con seguridad si nuestros viajes por el tiempo pueden terminar en un gran estruendo o en un imperceptible crujido, tenemos que tener mucho cuidado. Esta máquina, este sendero, nuestros cuerpos y nuestras ropas han sido esterilizados, como usted sabe, antes del viaje. Llevamos estos cascos de oxígeno para no introducir nuestras bacterias en una antigua atmósfera.

–¿Cómo sabemos qué animales podemos matar?

–Están marcados con pintura roja –dijo Travis–. Hoy, antes de nuestro viaje, enviamos aquí a Lesperance con la Máquina. Vino a esta Era particular y siguió a ciertos animales.

–¿Para estudiarlos?

–Exactamente –dijo Travis–. Los rastreó a lo largo de toda su existencia, observando cuáles vivían mucho tiempo. Muy pocos. Cuántas veces se acoplaban. Pocas. La vida es breve. Cuando encontraba alguno que iba a morir aplastado por un árbol u otro que se ahogaba en un pozo de alquitrán, anotaba la hora exacta, el minuto y el segundo, y le arrojaba una bomba de pintura que le manchaba de rojo el costado. No podemos equivocarnos. Luego midió nuestra llegada al pasado de modo que no nos encontremos con el monstruo más de dos minutos antes de aquella muerte. De este modo, sólo matamos animales sin futuro, que nunca volverán a acoplarse. ¿Comprende qué cuidadosos somos?

–Pero si ustedes vinieron esta mañana –dijo Eckels ansiosamente–, debían haberse encontrado con nosotros, nuestro safari.

¿Qué ocurrió? ¿Tuvimos éxito? ¿Salimos todos... vivos?

Travis y Lesperance se miraron.

–Eso hubiese sido una paradoja –habló Lesperance–. El tiempo no permite esas confusiones..., un hombre que se encuentra consigo mismo. Cuando va a ocurrir algo parecido, el tiempo se hace a un lado. Como un avión que cae en un pozo de aire. ¿Sintió usted ese salto de la Máquina, poco antes de nuestra llegada? Estábamos cruzándonos con nosotros mismos que volvíamos al futuro. No vimos nada. No hay modo de saber si esta expedición fue un éxito, si cazamos nuestro monstruo, o si todos nosotros, y usted, señor Eckels, salimos con vida.

Eckels sonrió débilmente.

–Dejemos esto –dijo Travis con brusquedad–. ¡Todos de pie! Se prepararon a dejar la Máquina. La jungla era alta y la jungla era ancha y la jungla era todo el mundo para siempre y para siempre. Sonidos como música y sonidos como lonas voladoras llenaban el aire: los pterodáctilos que volaban con cavernosas alas grises, murciélagos gigantescos nacidos del delirio de una noche febril. Eckels, guardando el equilibrio en el estrecho sendero, apuntó con su rifle, bromeando.

–¡No haga eso! –dijo Travis.– ¡No apunte ni siquiera en broma, maldita sea! Si se le dispara el arma...

Eckels enrojeció.

– ¿Dónde está nuestro *Tyrannosaurus*?

– Lesperance miró su reloj de pulsera.

–Adelante. Nos cruzaremos con él dentro de sesenta segundos. Busque la pintura roja, por Cristo. No dispare hasta que se lo digamos. Quédese en el Sendero. ¡Quédese en el Sendero!

Se adelantaron en el viento de la mañana.

–Qué raro –murmuró Eckels–. Allá delante, a sesenta millones de años, ha pasado el día de elección. Keith es presidente. Todos celebran. Y aquí, ellos no existen aún. Las cosas que nos preocuparon durante meses, toda una vida, no nacieron ni fueron pensadas aún.

–¡Levanten el seguro, todos! –ordenó Travis–. Usted dispare primero, Eckels. Luego, Billings. Luego, Kramer.

–He cazado tigres, jabalíes, búfalos, elefantes, pero esto, Jesús, esto es caza –comentó Eckels –. Tiemblo como un niño.

– Ah –dijo Travis.

–Todos se detuvieron.

Travis alzó una mano.

–Ahí adelante –susurró–. En la niebla. Ahí está Su Alteza Real.

La jungla era ancha y llena de gorjeos, crujidos, murmullos y suspiros. De pronto todo cesó, como si alguien hubiese cerrado una puerta.

Silencio.

El ruido de un trueno.

De la niebla, a cien metros de distancia, salió el *Tyrannosaurus rex*.

–Jesucristo –murmuró Eckels.

–¡Chist!

Venía a grandes trancos, sobre patas aceitadas y elásticas. Se alzaba diez metros por encima de la mitad de los árboles, un gran dios del mal, apretando las delicadas garras de relojero contra el oleoso pecho de reptil. Cada pata inferior era un pistón, quinientos kilos de huesos blancos, hundidos en gruesas cuerdas de músculos, encerrados en una vaina de piel centelleante y áspera, como la cota de malla de un guerrero terrible. Cada muslo era una tonelada de carne, marfil y acero. Y de la gran caja de aire del torso colgaban los dos brazos delicados, brazos con manos que podían alzar y examinar a los hombres como juguetes, mientras el cuello de serpiente se retorcía sobre sí mismo. Y la cabeza, una tonelada de piedra esculpida que se alzaba fácilmente hacia el cielo, En la boca entreabierta asomaba una cerca de dientes como dagas. Los ojos giraban en las órbitas, ojos vacíos, que nada expresaban, excepto hambre. Cerraba la boca en una mueca de muerte. Corría, y los huesos de la pelvis hacían a un lado árboles y arbustos, y los pies se hundían en la tierra dejando huellas de quince centímetros de profundidad. Corría como si diese unos deslizantes pasos de baile, demasiado erecto y en equilibrio para sus diez toneladas. Entró fatigadamente en el área de sol, y sus hermosas manos de reptil tantearon el aire.

–¡Dios mío! –Eckels torció la boca–. Puede incorporarse y alcanzar la luna.

–¡Chist! –Travis sacudió bruscamente la cabeza–. Todavía no nos vio.

–No es posible matarlo. –Eckels emitió con serenidad este veredicto, como si fuese indiscutible. Había visto la evidencia y ésta era su razonada opinión. El arma en sus manos parecía un rifle de aire comprimido–. Hemos sido unos locos. Esto es imposible.

–¡Cállese! –siseó Travis.

–Una pesadilla.

–Dé media vuelta –ordenó Travis–. Vaya tranquilamente hasta la máquina. Le devolveremos la mitad del dinero.

–No imaginé que sería tan grande –dijo Eckels–. Calculé mal. Eso es todo. Y ahora quiero irme.

–¡Nos vio!

–¡Ahí está la pintura roja en el pecho!

El Lagarto del Trueno se incorporó. Su armadura brilló como mil monedas verdes. Las monedas, embarradas, humeaban. En el barro se movían diminutos insectos, de modo que todo el cuerpo parecía retorcerse y ondular, aun cuando el monstruo mismo no se moviera. El monstruo resopló. Un hedor de carne cruda cruzó la jungla.

–Sáquenme de aquí –pidió Eckels–. Nunca fue como esta vez. Siempre supe que saldría vivo. Tuve buenos guías, buenos safaris, y protección. Esta vez me he equivocado. Me he encontrado con la horma de mi zapato, y lo admito. Esto es demasiado para mí.

–No corra –dijo Lesperance–. Vuélvase. Ocúltese en la Máquina. –Sí.

Eckels parecía aturdido. Se miró los pies como si tratara de moverlos. Lanzó un gruñido de desesperanza.

–¡Eckels!

Eckels dio unos pocos pasos, parpadeando, arrastrando los pies. –¡Por ahí no!

El monstruo, al advertir un movimiento, se lanzó hacia adelante con un grito terrible. En cuatro segundos cubrió cien metros. Los rifles se alzaron y llamearon. De la boca del monstruo salió un torbellino que los envolvió con un olor de barro y sangre vieja. El monstruo rugió con los dientes brillantes al sol.

Eckels, sin mirar atrás, caminó ciegamente hasta el borde del Sendero, con el rifle que le colgaba de los brazos. Salió del Sendero, y caminó, y caminó por la jungla. Los pies se le hundieron en un musgo verde. Lo llevaban las piernas, y se sintió solo y alejado de lo que ocurría atrás.

Los rifles dispararon otra vez. El ruido se perdió en chillidos y truenos. La gran palanca de la cola del reptil se alzó sacudiéndose. Los árboles estallaron en nubes de hojas y ramas. El monstruo retorció sus manos de joyero y las bajó como para acariciar a los hombres, para partirlos en dos, aplastarlos como cerezas, meterlos entre los dientes y en la rugiente garganta. Sus ojos de canto rodado bajaron a la altura de los hombres, que vieron sus propias imágenes. Dispararon sus armas contra las pestañas metálicas y los brillantes iris negros.

Como un ídolo de piedra, como el desprendimiento de una montaña, el *Tyrannosaurus* cayó. Con un trueno, se abrazó a unos árboles, los arrastró en su caída. Torció y quebró el Sendero de Metal. Los hombres retrocedieron alejándose. El cuerpo golpeó el suelo, diez toneladas de carne fría y piedra. Los rifles dispararon. El monstruo azotó el aire con su cola acorazada, retorció sus mandíbulas de serpiente, y ya no se movió. Una fuente de sangre le brotó de la garganta. En alguna parte, adentro, estalló un saco de fluidos. Unas bocanadas nauseabundas empaparon a los cazadores. Los hombres se quedaron mirándolo, rojos y resplandecientes.

El trueno se apagó.

La jungla estaba en silencio. Luego de la tormenta, una gran paz. Luego de la pesadilla, la mañana.

Billings y Kramer se sentaron en el sendero y vomitaron. Travis y Lesperance, de pie, sosteniendo aún los rifles humeantes, juraban continuamente.

En la Máquina del Tiempo, cara abajo, yacía Eckels, estremeciéndose. Había encontrado el camino de vuelta al Sendero y había subido a la Máquina. Travis se acercó, lanzó una ojeada a Eckels, sacó unos trozos de algodón de una caja metálica y volvió junto a los otros, sentados en el Sendero.

–Límpiense.

Limpiaron la sangre de los cascos. El monstruo yacía como una loma de carne sólida. En su interior uno podía oír los suspiros y murmullos a medida que morían las más lejanas de las cámaras, y los órganos dejaban de funcionar, y los líquidos corrían un último instante de un receptáculo a una cavidad, a una glándula, y todo se cerraba para siempre. Era como estar junto a una locomotora estropeada o una excavadora de vapor en el momento en que se abren las válvulas o se las cierra herméticamente. Los huesos crujían. La propia carne, perdido el equilibrio, cayó como peso muerto sobre los delicados antebrazos, quebrándolos.

Otro crujido. Allá arriba, la gigantesca rama de un árbol se rompió y cayó. Golpeó a la bestia muerta como algo final.

–Ahí está– Lesperance miró su reloj–. Justo a tiempo. Ese es el árbol gigantesco que originalmente debía caer y matar al animal.

Miró a los dos cazadores: ¿Quieren la fotografía trofeo?

–¿Qué?

–No podemos llevar un trofeo al futuro. El cuerpo tiene que quedarse aquí donde hubiese muerto originalmente, de modo que los insectos, los pájaros y las bacterias puedan vivir de él, como estaba previsto. Todo debe mantener su equilibrio. Dejamos el cuerpo. Pero podemos llevar una foto con ustedes al lado.

Los dos hombres trataron de pensar, pero al fin sacudieron la cabeza. Caminaron a lo largo del Sendero de metal. Se dejaron caer de modo cansino en los almohadones de la Máquina. Miraron otra vez el monstruo caído, el monte paralizado, donde unos raros pájaros reptiles y unos insectos dorados trabajaban ya en la humeante armadura.

Un sonido en el piso de la Máquina del Tiempo los endureció. Eckels estaba allí, temblando.

–Lo siento –dijo al fin.

–¡Levántese! –gritó Travis.

Eckels se levantó.

–¡Vaya por ese sendero, solo! –agregó Travis, apuntando con el rifle–. Usted no volverá a la Máquina. ¡Lo dejaremos aquí!

Lesperance tomó a Travis por el brazo. –Espera...

–¡No te metas en esto! –Travis se sacudió apartando la mano–. Este hijo de perra casi nos mata. Pero eso no es bastante. Diablo, no. ¡Sus zapatos! ¡Míralos! Salió del Sendero. ¡Dios mío, estamos arruinados Cristo sabe qué multa nos pondrán! ¡Decenas de miles de dólares! Garantizamos que nadie dejaría el Sendero. Y él lo dejó. ¡Oh, condenado tonto! Tendré que informar al gobierno. Pueden hasta quitarnos la licencia. ¡Dios sabe lo que le ha hecho al tiempo, a la Historia!

–Cálmate. Sólo pisó un poco de barro.

–¿Cómo podemos saberlo? –gritó Travis–. ¡No sabemos nada! ¡Es un condenado misterio! ¡Fuera de aquí, Eckels!

Eckels buscó en su chaqueta.

–Pagaré cualquier cosa. ¡Cien mil dólares!

Travis miró enojado la libreta de cheques de Eckels y escupió.

–Vaya allí. El monstruo está junto al Sendero. Métale los brazos hasta los codos en la boca, y vuelva.

–¡Eso no tiene sentido!

–El monstruo está muerto, cobarde bastardo. ¡Las balas! No podemos dejar aquí las balas. No pertenecen al pasado, pueden cambiar algo. Tome mi cuchillo. ¡Extráigalas!

La jungla estaba viva otra vez, con los viejos temblores y los gritos de los pájaros. Eckels se volvió lentamente a mirar al primitivo vaciadero de basura, la montaña de pesadillas y terror. Luego de un rato, como un sonámbulo, se fue, arrastrando los pies.

Regresó temblando cinco minutos más tarde, con los brazos empapados y rojos hasta los codos. Extendió las manos. En cada una había un montón de balas. Luego cayó. Se quedó allí, en el suelo, sin moverse.

–No había por qué obligarlo a eso – dijo Lesperance.

–¿No? Es demasiado pronto para saberlo. –Travis tocó con el pie el cuerpo inmóvil.

–Vivirá. La próxima vez no buscará cazas como ésta. Muy bien. –Le hizo una fatigada seña con el pulgar a Lesperance–. Enciende. Volvamos a casa. 1492. 1776. 1812.

Se limpiaron las caras y manos. Se cambiaron las camisas y pantalones. Eckels se había incorporado y se paseaba sin hablar. Travis lo miró furiosamente durante diez minutos.

–No me mire –gritó Eckels–. No hice nada.

–¿Quién puede decirlo?

–Salí del sendero, eso es todo; traje un poco de barro en los zapatos. ¿Qué quiere que haga? ¿Que me arrodille y rece?

–Quizá lo necesitemos. Se lo advierto, Eckels. Todavía puedo matarlo. Tengo listo el fusil.

–Soy inocente. ¡No he hecho nada!

1999, 2000, 2055.

La máquina se detuvo.

–Afuera –dijo Travis.

El cuarto estaba como lo habían dejado. Pero no de modo tan preciso. El mismo hombre estaba sentado detrás del mismo escritorio. Pero no exactamente el mismo hombre detrás del mismo escritorio.

Travis miró alrededor con rapidez.

–¿Todo bien aquí? –estalló.

–Muy bien. ¡Bienvenidos!

Travis no se sintió tranquilo. Parecía estudiar hasta los átomos del aire, el modo como entraba la luz del sol por la única ventana alta.

–Muy bien, Eckels, puede salir. No vuelva nunca.

Eckels no se movió.

–¿No me ha oído? –dijo Travis–. ¿Qué mira?

Eckels olía el aire, y había algo en el aire, una sustancia química tan sutil, tan leve, que sólo el débil grito de sus sentidos subliminales le advertía que estaba allí. Los colores blanco, gris, azul, anaranjado, de las paredes, del mobiliario, del cielo más allá de la ventana, eran... eran... Y había una sensación. Se estremeció. Le temblaron las manos. Se quedó oliendo aquel elemento raro con todos los poros del cuerpo. En alguna parte alguien debía de estar tocando uno de esos silbatos que sólo pueden oír los perros. Su cuerpo respondió con un grito silencioso. Más allá de este cuarto, más allá de esta pared, más allá de este hombre que no era exactamente el mismo hombre detrás del mismo escritorio..., se extendía todo un mundo de calles y gente. Qué suerte de mundo era ahora, no se podía saber. Podía sentirlos cómo se movían, más allá de los muros, casi, como piezas de ajedrez que arrastraban un viento seco...

Pero había algo más inmediato. El anuncio pintado en la pared de la oficina, el mismo anuncio que había leído aquel mismo día al entrar allí por vez primera.

De algún modo el anuncio había cambiado.

SEFARI EN EL TIEMPO. S. A.
SEFARIS A KUALKUIER AÑO DEL PASADO
USTE NOMBRA EL ANIMAL NOSOTROS LO
LLEBAMOS AYI. USTE LO MATA.

Eckels sintió que caía en una silla. Tanteó insensatamente el grueso barro de sus botas. Sacó un trozo, temblando.

–No, no puede ser. Algo tan pequeño. No puede ser. ¡No!

Hundida en el barro, brillante, verde, y dorada, y negra, había una mariposa, muy hermosa y muy muerta.

–¡No algo tan pequeño! ¡No una mariposa! –gritó Eckels.

Cayó al suelo una cosa exquisita, una cosa pequeña que podía destruir todos los equilibrios, derribando primero la línea de un pequeño dominó, y luego de un gran dominó, y luego de un gigantesco dominó, a lo largo de los años, a través del tiempo. La mente de Eckels giró sobre si misma. La mariposa no podía cambiar las cosas. Matar una mariposa no podía ser tan importante. ¿Podía?

Tenía el rostro helado. Preguntó, temblándole la boca:

– ¿Quién... quién ganó la elección presidencial ayer?

El hombre detrás del mostrador se rió.

–¿Se burla de mí? Lo sabe muy bien. ¡Deutscher, por supuesto! No ese condenado debilucho de Keith. Tenemos un hombre fuerte ahora, un hombre de agallas. ¡Sí, señor! –El oficial calló–. ¿Qué pasa?

Eckels gimió. Cayó de rodillas. Recogió la mariposa dorada con dedos temblorosos.

–¿No podríamos –se preguntó a sí mismo, le preguntó al mundo, a los oficiales, a la Máquina,– no podríamos llevarla allá, no podríamos hacerla vivir otra vez? ¿No podríamos empezar de nuevo? ¿No podríamos...?

No se movió. Con los ojos cerrados, esperó estremeciéndose. Oyó que Travis gritaba; oyó que Travis preparaba el rifle, alzaba el seguro, y apuntaba.

El ruido de un trueno.