



# Simulación

José María Ferrer Caja  
Universidad Pontificia Comillas

# Introducción

---

- ❑ Proceso de diseño de un modelo de un sistema real y realización de experiencias con él
- ❑ La finalidad es aprender el comportamiento del sistema y eventualmente comparar diversas estrategias para la ayuda en la toma de decisiones
  
- ❑ **Sistema:** Conjunto de objetos o ideas interrelacionados como una unidad para la consecución de un fin
- ❑ **Modelo de simulación:** Representación simplificada de un sistema estocástico
  - ✓ La intervención del azar en el sistema se representa mediante variables aleatorias
  - ✓ Es fundamental el tratamiento del tiempo

# Cuándo se aplica simulación

---

- No existe formulación matemática
- No se conocen métodos de resolución analíticos o son demasiado complejos
- Aunque existen métodos de resolución analíticos resulta menos costoso simular
- Se desea experimentar con el sistema antes de su construcción
- Es imposible experimentar con el sistema directamente, aunque éste ya exista
- Se desea reducir el tiempo de la experimentación

# Inconvenientes de la simulación

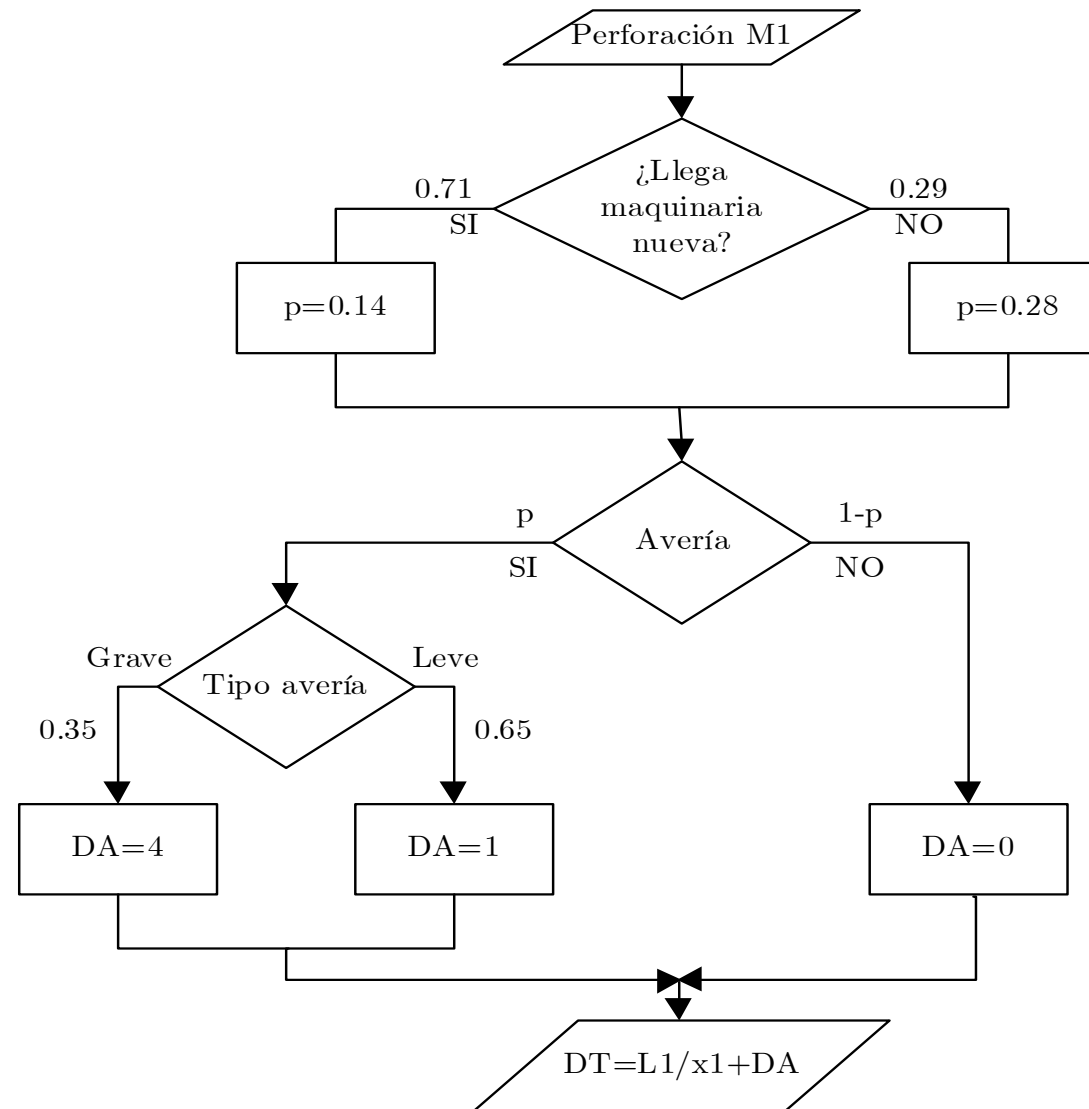
---

- La construcción del modelo puede ser costosa y compleja
- Es frecuente obtener resultados falsos, por menospreciar o soslayar aspectos importantes del sistema
- Es difícil conocer el grado de precisión de los resultados y analizarlos correctamente

# Ejemplo. Enunciado

- Se desea construir un túnel para conectar por carretera las dos vertientes de una cordillera
- Existen dos puntos posibles donde hacer el túnel, correspondientes a dos montañas M1 y M2 cercanas
- Mediante estudios preliminares se sabe que en el punto M1 la longitud del túnel habría de ser L1 y en la montaña M2, L2
- En M1, por las características del terreno, se perforaría a razón de  $x1$  unidades por jornada de trabajo, mientras que en M2 sería a razón de  $x2$  unidades.
- Se usará maquinaria nueva en caso de que se reciba a tiempo, hecho al que se ha asignado una probabilidad 0.71. La probabilidad de que la nueva maquinaria se averíe en M1 es 0.14, y 0.16 en M2
- Si no se dispone de la maquinaria nueva, se usará maquinaria vieja, cuyas probabilidades de avería son 0.28 en M1 y 0.19 en M2
- Las averías pueden ser graves, con probabilidad 0.35 y 4 jornadas de trabajo de reparación, o leves, con 1 jornada de trabajo
- ¿Dónde se debe perforar el túnel para tardar lo menos posible en la construcción de la carretera?

# Ejemplo. Modelo para M1



# Ejemplo. Número esperado de jornadas

---

- ❑ Mediante el **modelo de simulación** (50 simulaciones en cada punto):
  - ✓ 19.34 jornadas en M1
  - ✓ 20.22 jornadas en M2
  
- ❑ Mediante un **modelo teórico** (cálculos probabilísticos)
  - ✓ 19.37023 jornadas en M1
  - ✓ 20.345835 jornadas en M2

# Tipos de sistemas

---

## ❑ Sistemas continuos

- ✓ Las variables de estado cambian de forma continua respecto al tiempo

## ❑ Sistemas discretos

- ✓ Las variables de estado cambian en ciertos instantes de tiempo

## ❑ Estado de un sistema: Conjunto de valores que toman las variables del sistema, y que permiten describirlo



# Tipos de modelos de simulación

---

- ❑ Según la evolución del tiempo
  - ✓ **Estáticos**: representan un sistema en un instante particular
  - ✓ **Dinámicos**: representan la evolución de sistemas en el tiempo
- ❑ Según la existencia de aleatoriedad
  - ✓ **Deterministas**: No incluyen variables aleatorias. Dados unos datos de entrada, existe un único conjunto posible de datos de salida
  - ✓ **Estocásticos**: Contienen variables aleatorias. Los datos de salida pueden variar en diferentes simulaciones
- ❑ Según la evolución de las variables de estado
  - ✓ **Continuos**: Todas las variables de estado cambian de forma continua con el tiempo
  - ✓ **Discretos**: Todas las variables de estado cambian de forma discreta con el tiempo
  - ✓ **Híbridos**

# Simulación de eventos discretos

## □ Eventos

- ✓ Sucesos que producen un cambio en el estado del sistema

## □ Mecanismo de transición

- ✓ Muestra los cambios que se producen en el estado del sistema cuando se produce un evento

## □ Ejemplo: Sistema de colas con un servidor

- ✓ Estado: número de clientes que hay en el sistema
- ✓ Eventos: llegada de un cliente o final de un servicio
- ✓ Mecanismo de transición:

$$N(t) = \begin{cases} N(t) + 1 & \text{si llegada cliente} \\ N(t) - 1 & \text{si final de servicio cliente} \end{cases}$$

# Simulación de eventos discretos. Avance del tiempo

---

- ✓ **Reloj de simulación:** Variable que registra la cantidad de tiempo que ha sido simulada

## Métodos para incrementar el reloj

### ☐ Incremento en tiempo fijo (time step)

- ✓ Se avanza un intervalo fijo (pequeño) y se actualiza el estado del sistema
- ✓ Se considera que los eventos se producen al final del intervalo
- ✓ Origina problemas por simultaneidad de eventos

### ☐ Incremento por evento (event step)

- ✓ Se comprueba cuando se produce el evento más inminente y se avanza el reloj hasta ese instante
- ✓ No se incurre en errores en los instantes de ocurrencia de los eventos

# Simulación de eventos discretos. Otros elementos

---

## ❑ Lista de eventos

- ✓ Secuencia de instantes en las que se producen los eventos
- ✓ Puede ser generada al principio o en el transcurso de la simulación

## ❑ Contadores estadísticos

- ✓ Variables que almacenan información estadística relevante sobre el comportamiento del sistema

# Simulación de eventos discretos. Estructura

---

## ❑ Rutina de tiempo

- ✓ Determina el siguiente evento y avanza el reloj de simulación al instante en que va a ocurrir

## ❑ Rutina de evento

- ✓ Actualiza las variables de estado cuando se produce un evento. Hay una por cada tipo de evento

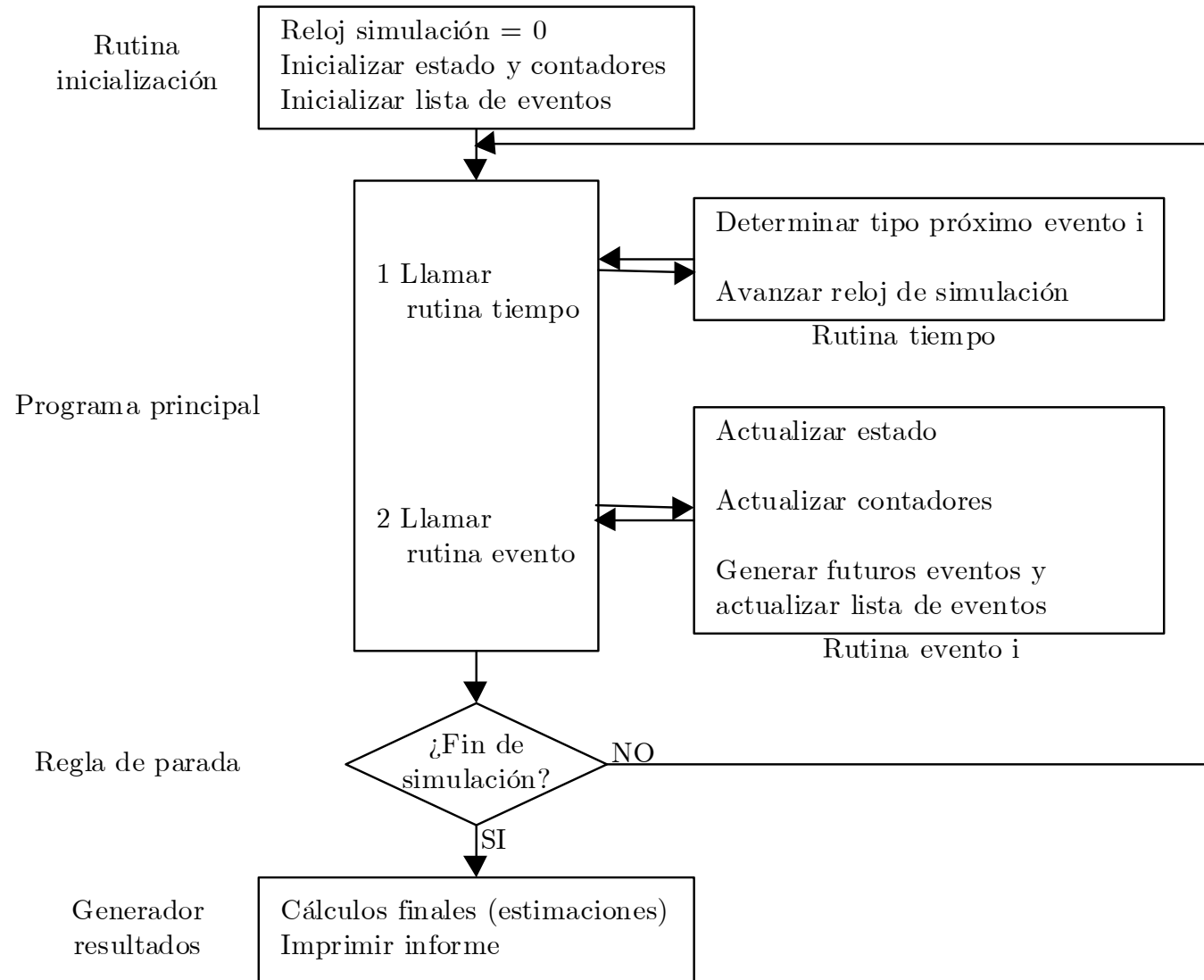
## ❑ Programa principal

- ✓ Enlaza las rutinas de tiempo y evento

## ❑ Generador de informes o resultados

- ✓ Realiza los cálculos o estimaciones de las características que se desean medir

# Simulación de eventos discretos. Diagrama de flujo



# Simulación de eventos discretos. Traza

---

## □ Traza del modelo

- ✓ Tabla en la que se recogen los valores de las variables que intervienen en el modelo en varias iteraciones
- ✓ Ayuda a planificar la programación
- ✓ Permite detectar errores en la programación o en el modelado

# Ejemplo: Línea de espera con un servidor (1)

- ❑ **Objetivo:** Analizar el número medio de clientes
- ❑ **Hipótesis y parámetros:**
  - ✓ Tiempos entre llegadas de clientes según distribución **F**
  - ✓ Tiempos de servicio según distribución **G**
  - ✓ Distribuciones de tiempos independientes entre sí
  - ✓ Tiempo máximo de simulación → **T**
- ❑ **Variable de estado:**
  - ✓ Número de clientes en el sistema → **N**
- ❑ **Eventos:**
  - ✓ Llegada de un cliente al sistema
  - ✓ Final del servicio a un cliente
- ❑ **Mecanismo de transición:**

$$N(t) \leftarrow \begin{cases} N(t) + 1 & \text{si es llegada de un cliente} \\ N(t) - 1 & \text{si es final de servicio} \end{cases}$$

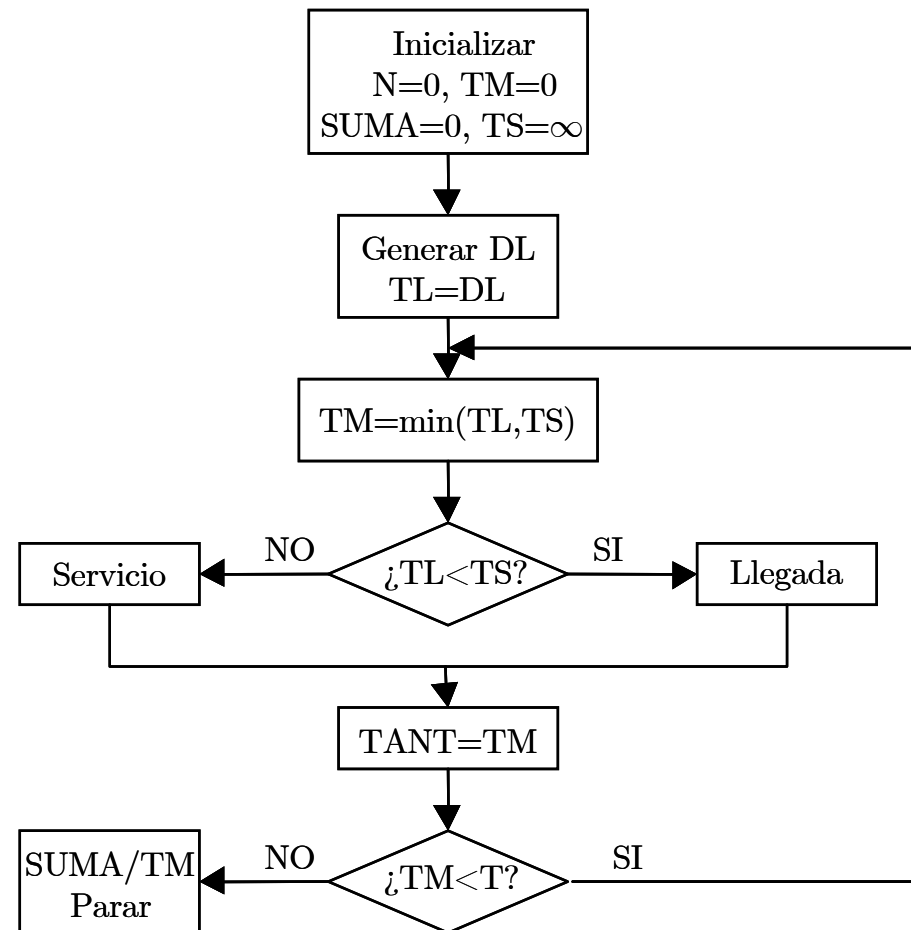


## Ejemplo: Línea de espera con un servidor (2)

### □ Otras variables :

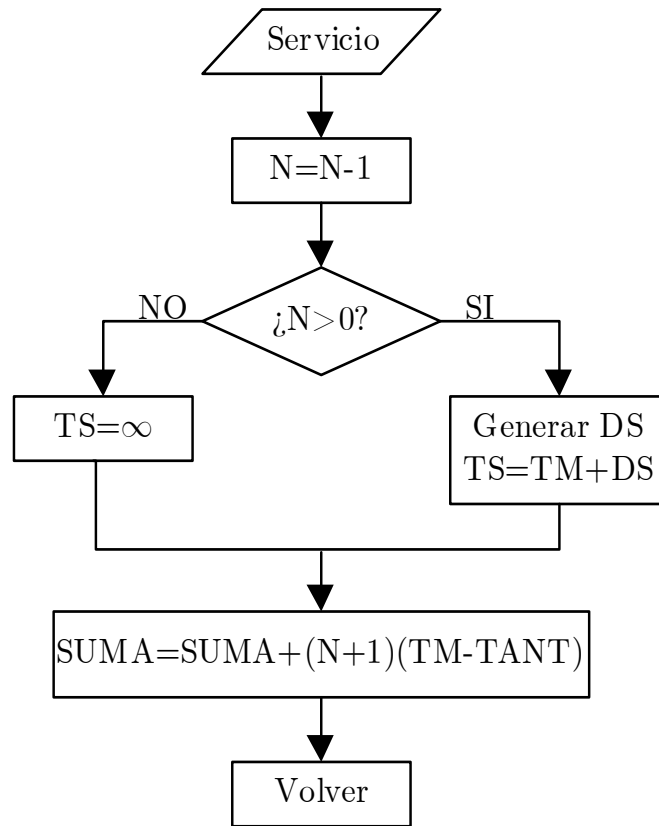
- ✓ reloj de simulación → **TM**
- ✓ Tiempos entre llegadas → **DL**  $\overset{d}{\sim}$  **F**
- ✓ Tiempos de servicio → **DS**  $\overset{d}{\sim}$  **G**
- ✓ Instante de la próxima llegada → **TL**
- ✓ Instante del próximo final de servicio → **TS**
- ✓ Contador acumulando suma de áreas de clientes en el sistema por tiempo de permanencia → **SUMA**
- ✓ Instante del último evento (auxiliar) → **TANT**

# Ejemplo: Línea de espera con un servidor (3)

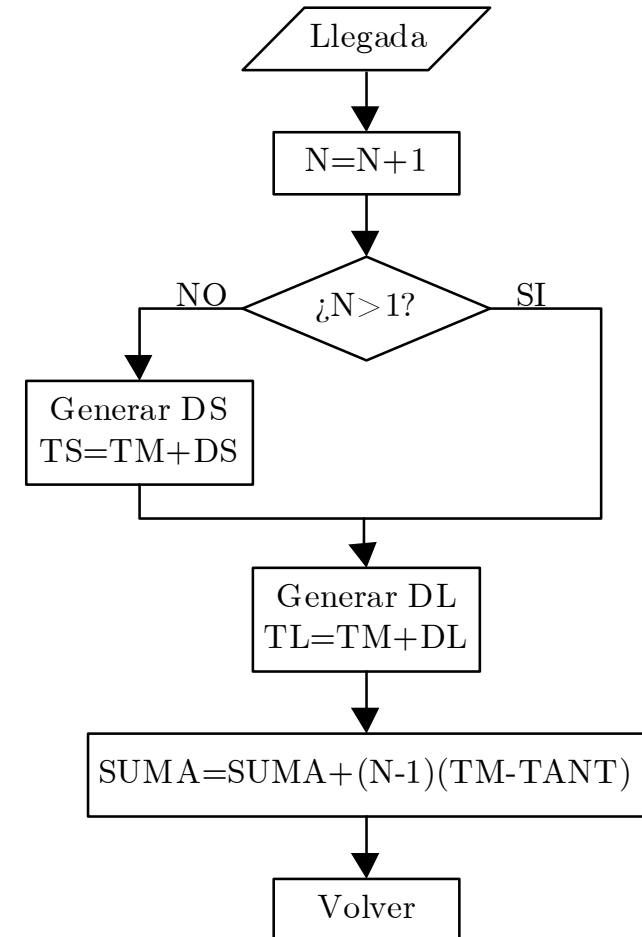


Programa principal

# Ejemplo: Línea de espera con un servidor (4)



Rutina de servicio



Rutina de llegada

# Ejemplo: Línea de espera con un servidor (5)

## □ Datos:

- ✓ Tiempos entre llegadas → **DL**: 3, 2, 5, 1, 2, 6, 6, 2, 8
- ✓ Tiempos de servicio → **DS**: 4, 1, 3, 1, 3, 2, 3, 5
- ✓ Tiempos de simulación → **T** = 35

## □ Traza:

Nº evento	Reloj	Tipo evento	N	TL	TS	SUMA
1	3	llegada	1	5	7	0+0·3=0
2	5	llegada	2	10	7	0+1·2=2
3	7	servicio	1	10	8	2+2·2=6
4	8	servicio	0	10	∞	6+1·1=7
5	10	llegada	1	11	14	7+0=7
6	11	llegada	2	13	14	8
7	13	llegada	3	19	14	12
8	14	servicio	2	19	15	15
9	15	servicio	1	19	18	17
10	18	servicio	0	19	∞	20
11	19	llegada	1	25	21	20
12	21	servicio	0	25	∞	22
13	25	llegada	1	27	28	22
14	27	llegada	2	35	28	24
15	28	servicio	1	35	33	26
16	33	servicio	0	35	∞	31
	35	final	0			31

# Ejemplo: Línea de espera con un servidor (6)

## □ Resultado:

- ✓ Estimación del número medio de clientes

$$\hat{E}[N] = \frac{31}{35} = 0.89$$

- ✓ El tiempo de simulación influye en el resultado: Si  $T = 18$

$$\hat{E}[N] = \frac{20}{18} = 1.11$$

# Etapas en la elaboración de un modelo de simulación (1)

---

## 1. Formular el problema

- ✓ objetivos del estudio
- ✓ hipótesis básicas
- ✓ parámetros que intervienen
- ✓ variables de estado del sistema

## 2. Reunir datos y crear un modelo

- ✓ Recolección de datos
- ✓ Análisis estadístico para modelar la aleatoriedad

## 3. Programar el modelo

- ✓ Lenguaje de programación de propósito general (diagrama de flujo)
- ✓ Lenguaje específico de simulación (GPSS, AutoMod, ...)

## 4. Verificar la programación

- ✓ La traza puede ayudar en esta etapa

# Etapas en la elaboración de un modelo de simulación (2)

---

## 5. Validar el modelo

- ✓ Ejecutar el modelo y comparar con el propio sistema o con soluciones teóricas de casos sencillos
- ✓ Alimentar el modelo con los mismos datos con los que se alimenta al sistema para obtener los resultados que se van a comparar

## 6. Diseñar el experimento

- ✓ Diseñar las estrategias a evaluar
- ✓ Pruebas que se van a llevar a cabo
- ✓ Número de simulaciones de cada una de ellas
- ✓ Uso de técnicas de reducción de la varianza
- ✓ Determinar la longitud del periodo transitorio y utilizar procedimientos para eliminar o atenuar su influencia

# Etapas en la elaboración de un modelo de simulación (3)

---

7. Ejecutar las simulaciones

8. Analizar los resultados

- ✓ Cada ejecución es una muestra simulada
- ✓ Recurrir al análisis estadístico para obtener conclusiones

9. Decidir si se concluye la simulación

- ✓ Si se requieren nuevas pruebas, volver a la etapa 6

10. Documentar y organizar las ejecuciones

- ✓ Recopilar y mostrar la información obtenida
- ✓ Hacer creíbles las conclusiones y decisiones que se propongan



# Sotware en simulación

---

## ❑ Lenguajes de propósito general

- ✓ Lenguajes de programación no específicos de simulación que por sus características permiten adaptarse a la construcción de modelos de simulación
- ✓ Ejemplos: C, Pascal, Fortran...

## ❑ Lenguajes de simulación

- ✓ Lenguajes de programación específicos para la construcción de modelos de simulación
- ✓ Ejemplos: AUTOMOD, GPSS, ARENA, WITNESS, ....

## ❑ Simuladores

- ✓ Programa ya creado para simular un tipo concreto de sistema
- ✓ No requiere experiencia en programación
- ✓ Ejemplos: simulador de vuelo, simulador de Fórmula 1,...

# Lenguaje general vs. Lenguaje de Simulación

Ventajas lenguajes simulación	Ventajas lenguajes propósito general
Reducen esfuerzo de programación (presentan las características necesarias para el desarrollo de modelos de simulación)	Más conocidos por los programadores
Entorno natural de trabajo (bloques básicos de programación más afines)	Disponibles en cualquier plataforma
Fácil modificación para diferentes experimentos	Menor tiempo de ejecución
Facilitan detección de errores	Mayor flexibilidad

# Características deseables del lenguaje

## □ Generales

- ✓ Flexibilidad de modelado
- ✓ Facilidad para desarrollar y depurar el modelo
- ✓ Alta velocidad de ejecución del modelo
- ✓ Admisión de modelos de gran tamaño
- ✓ Compatibilidad con diferentes plataformas informáticas
- ✓ Apto para simulación discreta y continua

## □ Animación

- ✓ Capacidad para mostrar gráficamente el comportamiento del modelo. Animación 3D
- ✓ Uso de play-back. Distintas velocidades de animación

## □ Estadística

- ✓ Variedad de distribuciones probabilísticas para datos de entrada
- ✓ Intervalos de confianza y contrastes de hipótesis

## □ Informes de resultados

# Enfoques de los lenguajes de simulación

---

- ❑ Interpretación de la realidad en la que se basa un lenguaje. Influye en
  - ✓ Selección del evento o suceso más inminente
  - ✓ Gestión del tiempo
- ❑ Dos enfoques diferentes
  - ✓ Programación de sucesos → Usado por GASP, ARENA, SLAM,...
  - ✓ Interacción de procesos → Usado por GPSS, ARENA, SIMULA,...
- ❑ Tipos de sucesos
  - ✓ Suceso incondicional: elegible para ser ejecutado cuando llega el instante de tiempo para el que ha sido programado. Depende exclusivamente del tiempo
  - ✓ Suceso condicional: además del tiempo, puede depender de condiciones adicionales relativas al estado de otros componentes del sistema

# Programación de sucesos (Event Scheduling)

---

- ❑ Secuencia de sucesos incondicionales a lo largo del tiempo
- ❑ Procedimiento ES de control del tiempo
  1. Seleccionar de la lista de sucesos el más próximo
  2. Actualizar el reloj de simulación
  3. Llamar a la rutina correspondiente al tratamiento del suceso
- ❑ Cualquier verificación de una condición diferente de la del tiempo del reloj debe realizarse dentro de las rutinas de tratamiento de los sucesos
- ❑ Durante la ejecución de las rutinas de evento no avanza el reloj de simulación

# Interacción de procesos (Process Interaction)

---

- ❑ Secuencia de procesos, cada uno de los cuales puede tener dos componentes
  - ✓ un segmento de condición, cuya ejecución identifica si se puede pasar a ejecutar la segunda componente
  - ✓ un segmento de acción
- ❑ Proceso: secuencia ordenada de sucesos interrelacionados, separados por “pasos” de tiempo
- ❑ Describe la experiencia entera de una “entidad” como su flujo a través de un sistema

# Generación de números aleatorios

- ❑ **Secuencia aleatoria:** secuencia de números obtenida al azar de algún conjunto prefijado
  - ✓ Normalmente es suficiente generar valores de la distribución uniforme  $U(0,1)$ . A partir de ella se pueden generar otras distribuciones
- ❑ **Metódos para generar secuencias aleatorias:**
  - ✓ **Métodos físicos** (Ruleta, dado, etc.): Los más efectivos para conseguir aleatoriedad, pero poco prácticos para simular grandes cantidades de variables en el ordenador
  - ✓ **Tablas de números aleatorios:** Obtenidas normalmente mediante métodos físicos. Ocupan demasiada memoria

Se requieren otros métodos más rápidos, aunque se pierda aleatoriedad → **números pseudoaleatorios**

# Generación de números pseudoaleatorios

---

- ❑ Una secuencia de números pseudoaleatorios debe verificar:
  - ✓ Distribución uniforme
  - ✓ Independencia estadística
- ❑ Un método para generar números pseudoaleatorios debe verificar:
  - ✓ Proporcionar variedad de secuencias
  - ✓ Secuencias reproducibles
  - ✓ Con un ciclo no repetitivo arbitrariamente grande
  - ✓ Velocidad de generación
  - ✓ Ocupar escasa memoria en el ordenador
- ❑ Esto se ha conseguido mediante **métodos aritméticos**



# Métodos congruenciales

- ❑ Fórmulas recurrentes basadas en el concepto de congruencia

$$x_{n+1} \equiv a x_n + b \pmod{m} \quad a, b, m \in \mathbb{N}, a < m, b < m$$
$$x_n \in \{0, 1, \dots, m - 1\} \quad \forall n \in \mathbb{N} \cup \{0\}$$

- ✓  $x_{n+1}$  es el resto de dividir  $a x_n + b$  entre  $m$
- ✓  $m \rightarrow$  **Módulo** (longitud máxima del ciclo)
- ✓  $a \rightarrow$  **Multiplicador**
- ✓  $x_0 \rightarrow$  **Semilla**
- ✓ Si  $b = 0 \rightarrow$  método multiplicativo
- ✓ Si  $b > 0 \rightarrow$  método mixto
- ✓ Si se desean valores uniformes  $(0,1)$ , se divide entre  $m$

$$u_n = \frac{x_n}{m}$$

# Métodos congruenciales. Ejemplos

□  $m = 9, a = 5, b = 1, x_0 = 1$

$$5 \cdot 1 + 1 = 6 \quad x_1 = 6 \quad u_1 = 6/9 = 0.6666$$

$$5 \cdot 6 + 1 = 31 \quad x_2 = 4 \quad u_2 = 4/9 = 0.4444$$

$$5 \cdot 4 + 1 = 21 \quad x_3 = 3 \quad u_3 = 3/9 = 0.3333$$

$$5 \cdot 3 + 1 = 16 \quad x_4 = 7 \quad u_4 = 7/9 = 0.7777$$

$$5 \cdot 7 + 1 = 36 \quad x_5 = 0 \quad u_5 = 0/9 = 0$$

$$5 \cdot 0 + 1 = 1 \quad x_6 = 1 = x_0 \quad u_6 = 1/9 = 0.1111$$

✓ A partir de aquí se repetiría indefinidamente la secuencia

**1, 6, 4, 3, 7, 0**

□  $m = 8, a = 5, b = 7, x_0 = 4$

✓ La secuencia generada es

**4, 3, 6, 5, 0, 7, 2, 1, 4, 3, ...**

✓ En este caso el ciclo alcanza su longitud máxima ( $m = 8$ )

# Métodos congruenciales. Generadores usuales

- ❑ Características deseables:
  - ✓  $m$  que facilite los cálculos ( $2^k$  o  $10^k$ )
  - ✓ Longitud del ciclo grande: lo ideal es  $m$
- ❑ Parámetros más usados para el método multiplicativo
  - ✓  $a = 7^5 = 16807$        $b = 0$        $m = 2^{31} - 1 = 2147483647$
  - ✓  $a = 63036016$        $b = 0$        $m = 2^{31} - 1$
  - ✓  $a = 2^{16} + 3$        $b = 0$        $m = 2^{31}$
- ❑ El segundo generador es el que mejores características estadísticas tiene. Semillas usadas:
  - ✓ 1973272912, 281629770, 20006270, 1280689831, 2096730329, 1933576050, 913566091

# Generación de variables aleatorias discretas

- Se quiere generar un valor de la v. a.  $X$  con función de probabilidad

$$p_i = P(X = x_i) \quad i \in \{1, \dots, K\}$$
$$\sum_{i=1}^K p_i = 1$$

- ✓ Se genera un valor  $u$  siguiendo la distribución  $U(0,1)$
- ✓ Se toma  $x_k$  tal que

$$F(x_{k-1}) < u \leq F(x_k)$$

- ✓ Siendo  $F(x)$  la función de distribución de  $X$ , es decir

$$F(x_k) = \sum_{i=1}^k p_i$$

# Generación de variables discretas. Ejemplo

- ❑ Se desea una muestra de tamaño 5 de la variable aleatoria

$x_k$	$P(X=x_k)$
0	0.1
1	0.2
2	0.5
3	0.2

a partir de la muestra siguiente de números uniformes:

0.20, 0.88, 0.89, 0.69, 0.50, 0.79, 0.98, 0.56, 0.06, 0.41

- ✓ Calculamos la función de distribución:

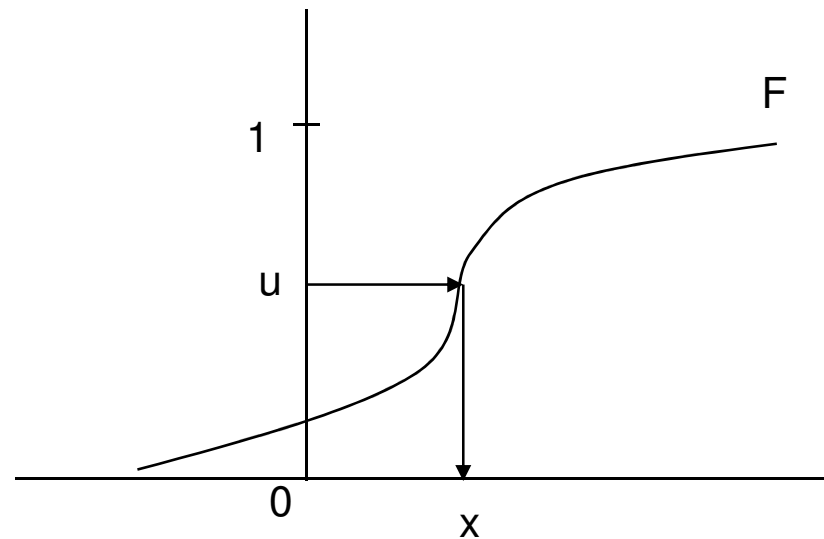
	$F(x)$
$x < 0$	0
$0 \leq x < 1$	0.1
$1 \leq x < 2$	0.3
$2 \leq x < 3$	0.8
$x \geq 3$	1

- ✓ Aplicando la fórmula se obtienen: 1, 3, 3, 2, 2, 2, 3, 2, 0, 2

# Generación de variables aleatorias continuas. Método de la transformada inversa

- ❑ Se quiere generar un valor de la v. a.  $X$  con función de distribución  $F(x)$ 
  - ✓ Se genera un valor  $u$  siguiendo la distribución  $U(0,1)$
  - ✓ Se toma  $x$  tal que

$$F(x) = u \quad \Leftrightarrow \quad x = F^{-1}(u)$$



# Método de la transformada inversa. Ejemplos

□ Uniforme:  $U(a,b)$   $F(x) = \begin{cases} 0, & x < a \\ \frac{x - a}{b - a}, & a \leq x < b \\ 1, & x \geq b \end{cases}$

- ✓ Se genera un valor  $u$  siguiendo la distribución  $U(0,1)$
- ✓ Se toma  $x$  tal que

$$x = a + (b - a)u$$

□ Exponencial:  $\exp(\lambda)$   $F(x) = \begin{cases} 0, & x < 0 \\ 1 - e^{-\lambda x}, & x \geq 0 \end{cases}$

- ✓ Se genera un valor  $u$  siguiendo la distribución  $U(0,1)$
- ✓ Se toma  $x$  tal que

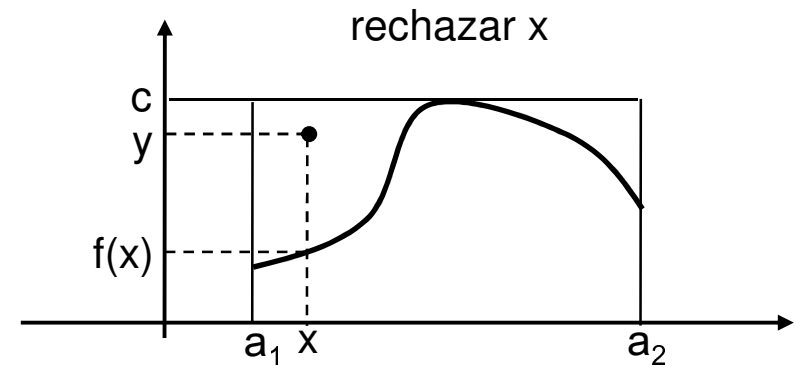
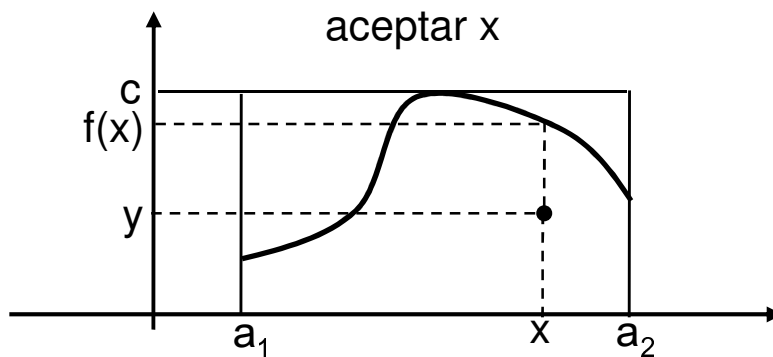
$$x = \frac{-\ln(u)}{\lambda}$$

# Método simple de aceptación-rechazo

- ❑ Se quiere generar un valor de la v. a.  $X$  con soporte acotado  $(a_1, a_2)$  y función de densidad  $f(x)$
- ❑ Sea  $c$  una cota superior de  $f(x)$  (preferentemente el valor máximo de  $f(x)$ )
  1. Se generan dos valores  $u_1, u_2$  siguiendo la distribución  $U(0,1)$
  2. A partir de ellos se obtienen  $x \in (a_1, a_2), y \in (0, c)$  mediante

$$x = a_1 + (a_2 - a_1)u_1 \quad y = cu_2$$

3. Si  $y \leq f(x)$  se **acepta**  $x$ ; si  $y > f(x)$  se **rechaza**  $x \rightarrow$  volver al paso 1





# Método simple de aceptación-rechazo. Ejemplo

- Generar un valor de una variable con distribución **Beta(2,2)**

$$f(x) = \begin{cases} 6x(1-x) & 0 \leq x \leq 1 \\ 0 & \text{resto} \end{cases}$$

$$c = \max f(x) = 1.5$$

$u_1$	$u_2$	$x$	$y$	$f(x)$	Aceptar
0,2304	0,9952	0,2304	1,4928	0,9721	NO
0,8849	0,7862	0,8849	1,1793	0,6111	NO
0,8915	0,0319	0,8915	0,0479	0,5804	SI

**solución:  $x = 0.8915$**

# Generación de la distribución normal. Método del teorema central del límite

## □ Teorema central del límite

Sean  $X_1, X_2, \dots, X_n$  v.a.i.i.d con media  $\mu$  y desviación típica  $\sigma$

$$Z = \frac{\sum_{i=1}^n X_i - n\mu}{\sigma\sqrt{n}} \xrightarrow{n \rightarrow \infty} N(0,1)$$

□ Se quiere generar un valor  $z$  de la v. a.  $Z \sim N(0,1)$

✓ Se generan  $n$  valores  $u_1, \dots, u_n$  siguiendo la distribución  $U(0,1)$

$$z = \frac{\sum_{i=1}^n u_i - n/2}{\sqrt{n/12}}$$

Ejemplo: Para  $n=12 \rightarrow z = \sum_{i=1}^{12} u_i - 6$

✓ Si se quiere generar un valor  $x$  de la v. a.  $X \sim N(\mu, \sigma)$

$$x = \sigma z + \mu$$

# Generación de la distribución normal. Método de Box-Müller

- ❑ Se quiere generar un valor  $z$  de la v. a.  $Z \sim N(0,1)$ 
  - ✓ Se generan dos valores  $u_1, u_2$  siguiendo la distribución  $U(0,1)$

$$z = \sqrt{-2 \ln u_1} \cos(2\pi u_2)$$

o bien

$$z = \sqrt{-2 \ln u_1} \sin(2\pi u_2)$$

- ✓ Este método es más eficiente
- ✓ Permite generar 2 valores normales a partir de 2 uniformes

# Análisis estadístico de resultados

□ Sea  $X_1, X_2, \dots, X_n$  una muestra aleatoria simple de la variable aleatoria  $X$

□ Media muestral: 
$$\bar{X}_n = \frac{1}{n} \sum_{i=1}^n X_i$$

✓ Es un estimador insesgado de la media poblacional  $\mu$

$$E[\bar{X}_n] = \mu$$

□ Cuasivarianza muestral: 
$$S_n^2 = \frac{1}{n-1} \sum_{i=1}^n (X_i - \bar{X}_n)^2$$

✓ Es un estimador insesgado de la varianza poblacional  $\sigma^2$

$$E[S_n^2] = \sigma^2$$

# Precisión de la media muestral

- ❑ Varianza de la media muestral

$$V[\bar{X}_n] = \frac{\sigma^2}{n} \simeq \frac{S_n^2}{n}$$

- ❑ Precisión de la media muestral

$$\beta = \frac{S_n / \sqrt{n}}{\bar{X}_n}$$

- ❑ Intervalo de confianza de nivel  $1 - \alpha$  para la media poblacional

$$\bar{X}_n \mp t_{n-1, \alpha/2} \frac{S_n}{\sqrt{n}}$$

# Cálculo iterativo de los estimadores

- Obtención de  $\bar{X}_n$  y  $S_n^2$  a partir de  $\bar{X}_{n-1}$  y  $S_{n-1}^2$

$$\bar{X}_n = \frac{1}{n} \left[ (n-1)\bar{X}_{n-1} + X_n \right]$$

$$S_n^2 = \frac{1}{n-1} \left[ (n-2)S_{n-1}^2 + \frac{n-1}{n} (\bar{X}_{n-1} - X_n)^2 \right]$$