Modelado y Simulación (505103009)
Tema 4. Generación de números aleatorios

Javier Vales Alonso
Grado en Ingeniería Telemática
2020

Universidad Politécnica de Cartagena

Tabla de contenido

¿Cómo estudiar esta unidad?

1. Haga una primera lectura de la unidad. Concéntrese en ver las ideas generales y hacer una primera revisión del algoritmo de los generadores congruenciales lineales.
2. Haga una revisión más a fondo de las propiedades matemáticas de dicho algoritmo.
3. Implemente el algoritmo GCL en MATLAB.
4. En caso de dudas, puede consultar los libros de referencia, o contactar con el profesor.

Introducción

Objetivo

El objetivo principal de este tema es **describir métodos para la obtención de muestras de una variable aleatoria con distribución (aparente)** $\mathcal{U}(0,1)$.

La método de generación debe cumplir las siguientes propiedades:

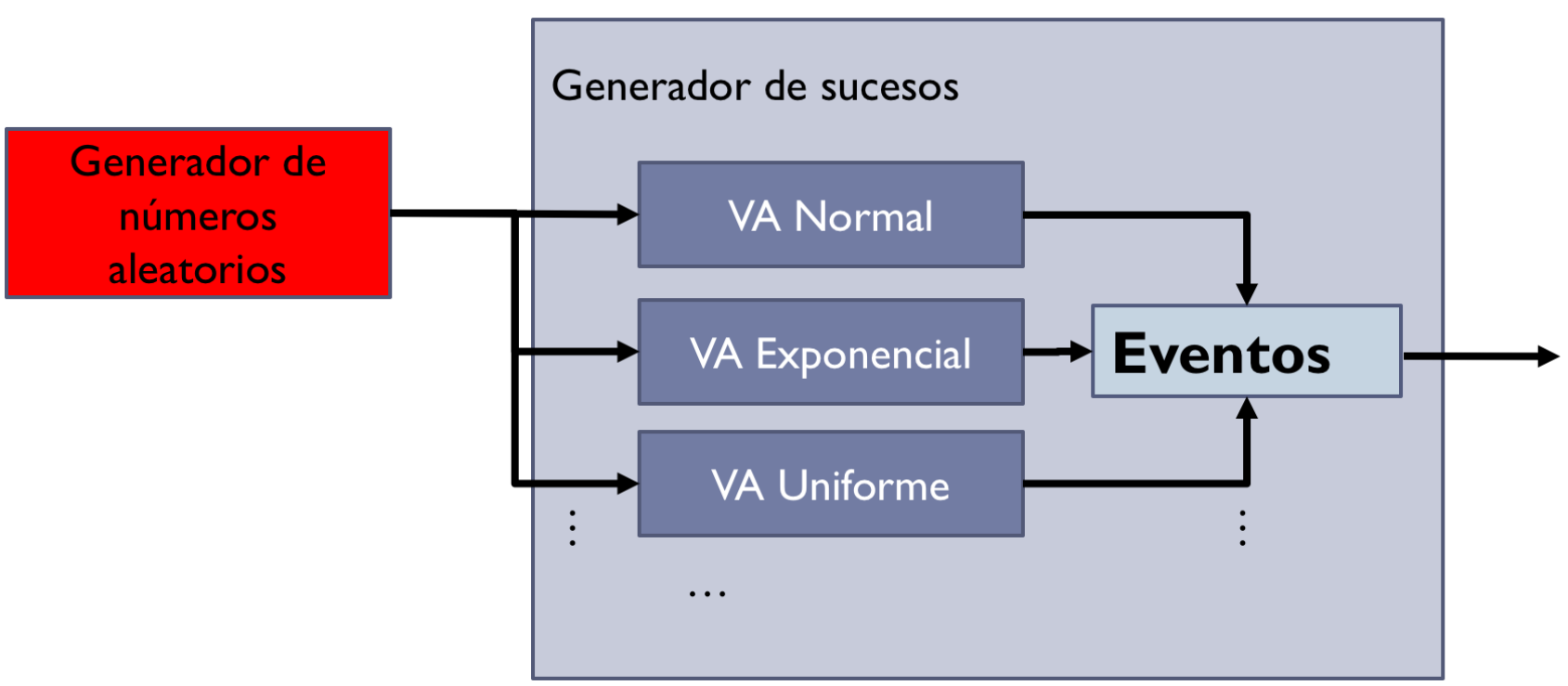
- Las muestras deben ser (parecer) independientes entre si.
- Deben estar distribuidas de modo homogéneo en el intervalo $(0,1)$.
- El método debe ser (computacionalmente) eficiente.

Objetivo (II)

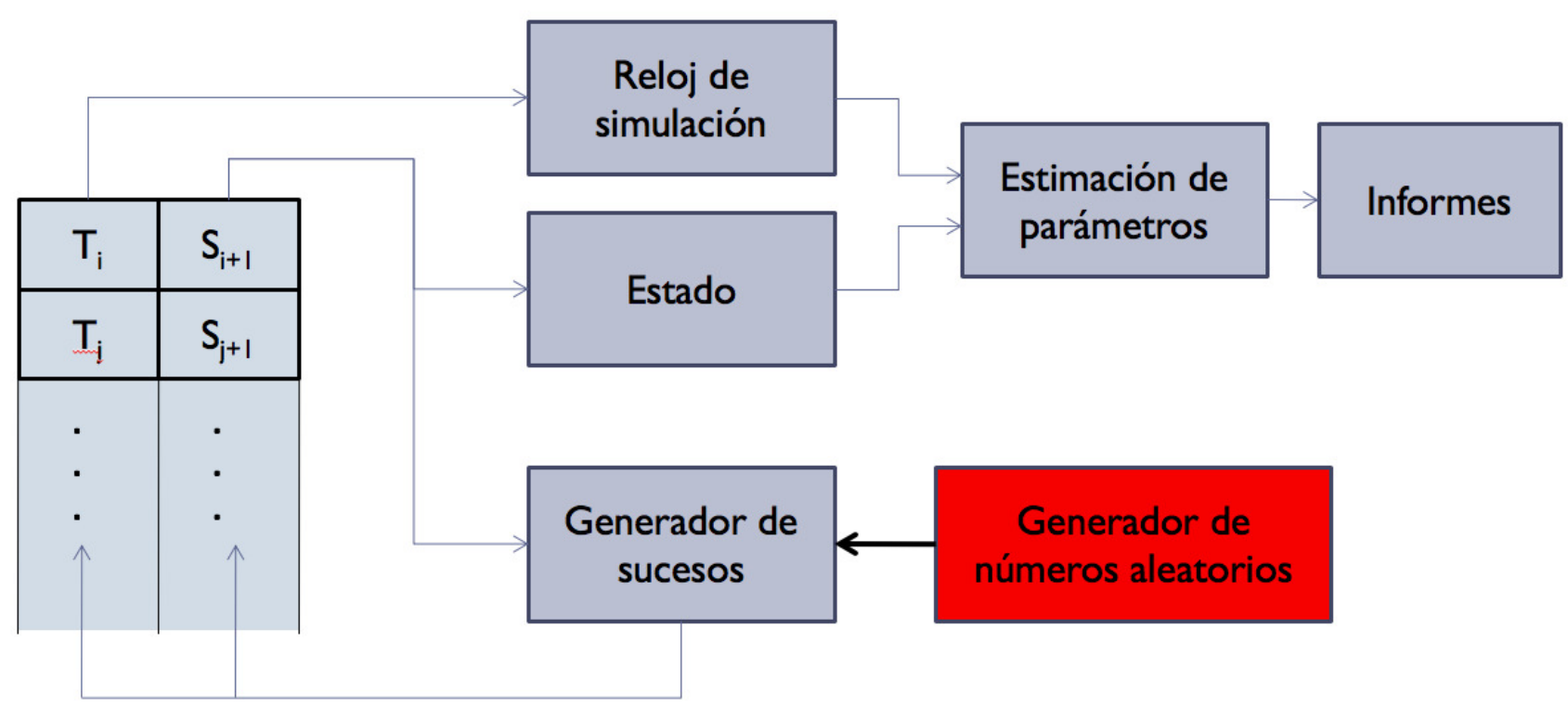
La generación de azar es útil en múltiples contextos, e.g.:

- Simulación
- Criptografía
- Juegos
- Toma de decisiones
- Optimización
- Protocolos
- Algorítmica

Objetivo (III)



Objetivo (IV)


Reloj de simulación
Estado
Estimación de parámetros
Informes
T_i
S_{i+1}
T_j
S_{j+1}
Generador de sucesos
Generador de números aleatorios

Métodos de generación

Métodos de generación

Métodos físicos basados en la medida de algún proceso natural de tipo aleatorio:

- Procesos cuánticos a nivel atómico o sub-atómico:
  - Ruido impulsivo (fuente de ruido cuántico).
  - Procesos de desintegración nuclear.
  - Detección de fotones viajando a través de un espejo semitransparente.
- Medida de ruido:
  - Medida del ruido térmico en resistencias.
  - Medida del ruido de avalancha en un diodos Zener.
  - Medida del ruido atmosférico.
- Medida de la deriva del reloj del computador

Métodos de generación (II)

Métodos artificiales basados en la generación de muestras mediante algún dispositivo:

- Métodos manuales (lanzamiento de dados, selección de cartas)
- Métodos mecánicos (ruletas, bombos). En 1939 Kendall y Babington-Smith crearon tablas de 100000 números a partir de un disco giratorio.
- Métodos electrónicos:
  - Tubos de vacío aleatoriamente pulsantes (50 muestras/s).
  - RAND Corporation publica en 1955 una tabla con 1 millón de muestras.
  - ERNIE (Electronic Random Number Indicator Equipment) un computador creado en 1957 para generar números al azar para la lotería.

Métodos de generación (III)

Métodos algorítmicos:

- Son métodos artificiales (desarrollados a partir de 1940) basados en la aplicación de una serie de reglas aritméticas o lógicas (algoritmo).
- El algoritmo comienza a partir de un número llamado **semilla** y genera una secuencias **pseudo-aleatorias**, ya que partiendo de la misma semilla se obtienen los mismos valores.
- Esta secuencia no es realmente aleatoria, pero su uso puede ser suficiente en diversos contextos (e.g., simulación, toma de decisiones, etc.).

Métodos de generación (IV)

Requisitos de los **métodos algorítmicos**:

- Aleatoriedad (aparente).
- Independencia (aparente) entre muestras.
- Periodo (número de muestras que se pueden obtener con el algoritmo antes de repetir la secuencia) máximo.
- Eficientes (en términos de computación).
- Secuencia reproducible (permite replicar una simulación).

Métodos de generación (V)

Ejemplos de métodos algorítmicos (ver aquí lista exhaustiva):

- Algoritmo de los **cuadrados medios** (Von Neumann-Metropolis, 1949).
- Generadores congruenciales lineales - **GCL** (Lehmer, Thomson, Rotenberg, 1958).
- **RC4**, algoritmo criptográfico con secuencias pseudo-generadas (Rivest, 1987).

Generadores congruenciales lineales

Generadores congruenciales lineales

Dados los parámetros a (multiplicador), c (incremento) y m (módulo) y una semilla Z_0, un GCL obtiene la muestra i-ésima, Z_i, como:

$$Z_i = (aZ_{i-1} + c) \bmod m \tag{1}$$

Y la muestra i-ésima del generador con distribución $\mathcal{U}(0,1)$, es:

$$u_i = \frac{Z_i}{m} \tag{2}$$

Donde:

- $a, m \in \mathbb{N}^+$, $c \in \mathbb{N}$
- Si $c = 0$ el GCL es **multiplicativo** (GCLM) y $Z_0 \in \mathbb{N}^+$
- Si $c > 0$ el GCL es **mixto** y $Z_0 \in \mathbb{N}$

Generadores congruenciales lineales (II)

Ejemplo 1:

a=1

m=5

c=3

$Z_0 = 1$

$$Z_i = (aZ_{i-1} + c) \bmod m$$

Generadores congruenciales lineales (III)

Ejemplo 2:

a=1

m=5

c=0

$Z_0 = 1$

$$Z_i = aZ_{i-1} \bmod m$$

Generadores congruenciales lineales (IV)

Teorema de Hull-Dobell. Un GCL posee periodo máximo m si:

1. m y c primos entre si: el único entero positivo que divide exactamente a m y a c es el 1.
2. Si q es un número primo que divide a m, entonces q también divide a $(a-1)$.
3. Si 4 divide a m, entonces 4 también divide a $(a-1)$.

Un GCL multiplicativo no verifica el teorema de Hull-Dobell ya que si c=0 no se cumple la condición 1.

Generadores congruenciales lineales (V)

Ejemplo 3:

a=4

m=5

c=0

$Z_0 = 1$

$$Z_i = aZ_{i-1} \bmod m$$

Generadores congruenciales lineales (VI)

Un GCLM puede poseer periodo máximo $m - 1$ si se verifican las siguientes condiciones:

1. m ha de ser primo
2. a ha de ser raíz primitiva de m, i.e., satisfacer $a^n \bmod m \neq 1$ para $n = 1, \ldots, m - 2$.

Generadores congruenciales lineales (VII)

Además de buscar el máximo periodo posible, deben tenerse en cuenta las propiedades estadísticas de los generadores y su eficiencia de cálculo. Algunas configuraciones de uso común son:

Nombre	m	a	c
Fishman-Moore	$2^{31} - 1$	48271	0
Kobayashi	2^{31}	314159269	453806245
Coveyou-McPherson	2^{35}	5^{15}	1
glibc	2^{31}	1103515245	12345
MMIX (Donald Knuth)	2^{64}	6364136223846793005	1442695040888963407

Aspectos prácticos en simulación

- **¿Uno o varios generadores por simulador?** Solamente un generador por simulador (nunca uno por tipo de variable aleatoria).
- **¿Iniciación aleatoria de la semilla, por ejemplo, con el reloj del sistema?** No es aconsejable. Es mejor establecer una semilla conocida para poder reproducir la realización de simulación.
- Si ejecutamos realizaciones independientes de la simulación no es aceptable partir siempre de la misma semilla (llegaríamos al mismo resultado). Debe usarse como semilla de la realización n-ésima la última muestra de la realización $(n-1)$-ésima