Universidad Autónoma del Estado de Hidalgo

Instituto de Ciencias Básicas e Ingeniería

Área Académica de Ingeniería y Arquitectura

Programa Educativo

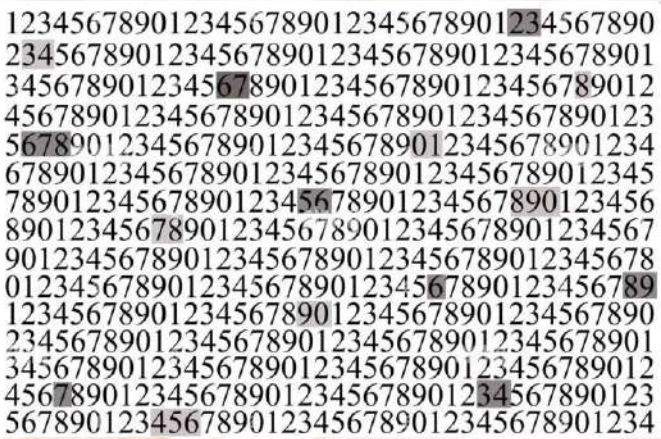
Doctorado en Ciencias en Ingeniería, con Énfasis en Análisis y Modelación de Sistemas

Asignatura

Optativa III (Análisis de Sistemas de Eventos Discretos)

Tema

Números pseudo aleatorios



Juan Carlos Seck Tuoh Mora
Norberto Hernández Romero
Joselito Medina Marín

17 de abril de 2023

Resumen

En este tema se presentan conceptos de números pseudo aleatorios así como su importancia en la generación de modelos de simulación. Además, se describen algoritmos de generación de números pseudo aleatorios, las propiedades que deben presentar, y las pruebas estadísticas a las que deben someterse para validar su uniformidad e independencia entre los números generados.

Palabras clave: Simulación, Números pseudo aleatorios, algoritmos.

Abstrac

In this topic, fundamentals of pseudo random numbers and its importance for developing simulation models. Moreover, some algorithms for pseudo random number generation, properties that algoritms must fulfilled, and statistical testing to validate the uniformity and independence of presudo random numbers are described.

Keywords: Simulation, Pseudo random numbers, algorithms.

Los números pseudo aleatorios

- Para poder realizar una simulación que incluya variabilidad dentro de sus eventos, es preciso generar una serie de números que sean aleatorios por sí mismos, y que su aleatoriedad se extrapole al modelo de simulación que se está construyendo.
- La construcción del modelo los números aleatorios juegan un papel relevante.
- Se debe determinar si los números que se utilizan para "correr" o ejecutar la simulación son realmente aleatorios o no

Los números pseudo aleatorios

- Se puede asegurar con altos niveles de confiabilidad que el conjunto de números que se utilizarán en una simulación se comportan de manera muy similar a un conjunto de números totalmente aleatorios.
- Por lo anterior, es que se les denomina *números pseudo aleatorios.*

Generación de números pseudo aleatorios

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

Generación de números pseudo aleatorios

- Debido a que no es posible generar números realmente aleatorios, consideramos los r_i como números *pseudo aleatorios*, generados por medio de algoritmos determinísticos que requieren parámetros de arranque.
- Para simular el comportamiento de una o más variables aleatorias es necesario contar con un conjunto suficientemente grande de r_i. que permita, por ejemplo, que la secuencia tenga al menos un periodo de vida de $n = 2^{31} = 2,147,483,648$.

Generación de números pseudo aleatorios

- De acuerdo con L'Ecuyer, una secuencia de r_i con periodo de vida de $n = 2^{31}$ es relativamente pequeña.
- Incluso una secuencia de r_i que contenga un ciclo de vida de $n = 2^{64}$ se considera pequeña.
- En la actualidad se cuentan con generadores y procesadores capaces de construir una secuencia de r_i con periodo de vida de $n = 2^{200}$.

Generación de números pseudo aleatorios

- ¿Por qué construir secuencias de números r_i suficientemente grandes?
- Simular atención a clientes en un banco que tiene 5 cajeros en paralelo, cada uno de los cuales atiende aproximadamente 50 clientes diarios. Hacer la simulación para 5 días.
- Simular tiempos de arribo de los clientes al banco, 5 cajeros en paralelo con 50 clientes diarios para cada uno, durante 5 días.
- Se requerirían 2,500 números pseudo aleatorios r_i.

Generación de números pseudo aleatorios

- ¿Qué pasa si quisiéramos ejecutar 2 réplicas de la simulación?
- Se requerirían entonces 5,000 r_i.
- ¿Cuántos números r_i serán necesarios para simular la operación del banco durante un año con 9 réplicas?

Generación de números pseudo aleatorios

- Dada la importancia de contar con un conjunto de r_i suficientemente grande, existen diferentes algoritmos determinísticos para obtenerlo.
- Es conveniente señalar que el conjunto de r_i debe ser sometido a una variedad de pruebas para verificar si los números que lo conforman son realmente independientes y uniformes.

Generación de números pseudo aleatorios

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

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

- Generar un conjunto de r_i es una tarea relativamente sencilla; sólo se tiene que diseñar un algoritmo propio de generación.
- Lo que resulta difícil es diseñar un algoritmo que genere un conjunto de r_i con periodo de vida suficientemente grande *(N),y* que además pase sin problema las pruebas de uniformidad e independencia.

2.2.1 Algoritmo de Cuadrados Medios

- Algoritmo No Congruencial propuesto en la década de los cuarenta del siglo XX por Von Neumann y Metrópolis.

1. Seleccionar una semilla (X_0) con *D* dígitos (D > 3).

2. Sea Y_0= resultado de elevar X_0 al cuadrado; sea X_1 = los D dígitos del centro, y sea r_1 = 0.D dígitos del centro.

3. Sea Y_i = resultado de elevar X_i al cuadrado; sea X_{i+1} = los *D* dígitos del centro, y sea r_{i+1} = 0.D dígitos del centro para toda i = 1,2,3,..., *n.*

4. Repetir el paso 3 hasta obtener los *n* números r_i deseados.

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

2.2.1 Algoritmo de Cuadrados Medios – Ejemplo

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

2.2.1 Algoritmo de Cuadrados Medios – Ejemplo

- Generar los primeros 5 números r_i a partir de una semilla $X_0 = 5735$, de donde se puede observar que D = 4 dígitos.
  Solución:
- $Y_0 = (5735)^2 = 32890225$ $X_1 = 8902$ $r_1 = 0.8902$
- $Y_1 = (8902)^2 = 79245604$ $X_2 = 2456$ $r_2 = 0.2456$
- $Y_2 = (2456)^2 = 06031936$ $X_3 = 0319$ $r_3 = 0.0319$
- $Y_3 = (0319)^2 = 101761$ $X_4 = 0176$ $r_4 = 0.0176$
- $Y_4 = (0176)^2 = 030976$ $X_5 = 3097$ $r_5 = 0.3097$

2.2.2 Algoritmo de Productos Medios

- Similar al de cuadrados medios, pero en éste se requieren 2 semillas.

1. Seleccionar una semilla (X_0) con *D* dígitos (D > 3).
2. Seleccionar una semilla *(X_1)* con D dígitos (D > 3).
3. Sea $Y_0 = X_0 * X_1$; sea X_2 = los *D* dígitos del centro, y sea $r_i = 0.D$ dígitos del centro.
4. Sea $Y_i = = X_i * X_{i+1}$; sea X_{i+2} = los D dígitos del centro, y sea r_{i+1} = 0.D dígitos del centro para toda *i* = 1,2,3,...,n;
5. Repetir el paso 4 hasta obtener los *n* números r_i deseados.

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

2.2.2 Algoritmo de Productos Medios – Ejemplo

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

2.2.2 Algoritmo de Productos Medios – Ejemplo

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

$Y_0 = (5015)(5734) = 28756010$ $X_2 = 7560$ $r_1 = 0.7560$

$Y_1 = (5734)(7560) = 43349040$ $X_3 = 3490$ $r_2 = 0.3490$

$Y_2 = (7560)(3490) = 26384400$ $X_4 = 3844$ $r_3 = 0.3844$

$Y_3 = (3490)(3844) = 13415560$ $X_5 = 4155$ $r_4 = 0.4155$

$Y_4 = (3844)(4155) = 15971820$ $X_6 = 9718$ $r_5 = 0.9718$

2.2.3 Algoritmo de Multiplicador Constante

- Algoritmo no congruencial similar al algoritmo de productos medios.

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

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

2.2.3 Algoritmo de Multiplicador Constante – Ejemplo

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

2.2.3 Algoritmo de Multiplicador Constante – Ejemplo

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

$Y_0 = (6965)(9803) = 68277895$ $X_1 = 2778$ $r_1 = 0.2778$

$Y_1 = (6965)(2778) = 19348770$ $X_2 = 3487$ $r_2 = 0.3487$

$Y_2 = (6965)(3487) = 24286955$ $X_3 = 2869$ $r_3 = 0.2869$

$Y_3 = (6965)(2869) = 19982585$ $X_4 = 9825$ $r_4 = 0.9825$

$Y_4 = (6965)(9825) = 68431125$ $X_5 = 4311$ $r_5 = 0.4311$

2.2.4 Algoritmo Lineal

- Algoritmo congruencial propuesto por D. H. Lehmer en 1951.
- Genera una secuencia de números enteros por medio de la siguiente ecuación recursiva:

$$X_i + 1 = (ax_i + c) mod\ (m), para\ i = 0, 1, 2, 3, \dots, n$$

Donde: X_0 es la semilla, $X_0 > 0$

a es la constante multiplicativa, $a > 0$

c es una constante aditiva, c > 0,

m es el módulo, m > 0. y todos deben ser números enteros

Se genera una secuencia de números enteros S = {0,1,2, 3,…,m-1}, y para la obtención de los números pseudo aleatorios se requiere la sig. ecuación:

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

2.2.4 Algoritmo Lineal – Ejemplo

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

2.2.4 Algoritmo Lineal – Ejemplo

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

$X_1 = (19*37 + 33) \bmod 100 = 36$ $r_1 = 36/99 = 0.3636$

$X_2 = (19*36 + 33) \bmod 100 = 17$ $r_2 = 17/99 = 0.1717$

$X_3 = (19*17 + 33) \bmod 100 = 56$ $r_3 = 56/99 = 0.5656$

$X_4 = (19*56 + 33) \bmod 100 = 97$ $r_4 = 97/99 = 0.9797$

2.2.4 Algoritmo Lineal

- En el ejemplo anterior se colocaron de manera arbitraria cada uno de los parámetros requeridos: X_0, a, c, m. Sin embargo, para que el algoritmo sea capaz de lograr el máximo periodo de vida N, es preciso que dichos parámetros cumplan ciertas condiciones. Banks y otros sugieren que:
  $m = 2^g$
  $a = 1 + 4k$
  k debe ser entero
  c relativamente primo a m
  g debe ser entero
- Bajo estas condiciones se obtiene un periodo de vida máximo: $N = m = 2^g$.

2.2.4 Algoritmo Lineal – Ejemplo.a

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

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

2.2.4 Algoritmo Lineal – Ejemplo.a

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

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

$X_0 = 6$

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

2.2.4 Algoritmo Lineal – Ejemplo.b

- Consideremos nuevamente el ejemplo anterior, pero tratemos de violar de manera arbitraria alguna de las condiciones. Supongamos que $a = 12$; se sabe que a no es el resultado de $1 + 4k$, donde k es un entero. Veamos el comportamiento del algoritmo.

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

2.2.4 Algoritmo Lineal – Ejemplo.b

- Consideremos nuevamente el ejemplo anterior, pero tratemos de violar de manera arbitraria alguna de las condiciones. Supongamos que a = 12; se sabe que a no es el resultado de 1 + 4k, donde k es un entero. Veamos el comportamiento del algoritmo.

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

$X_0 = 6$

$X_1 = (12*6 + 7) \bmod 8 = 7$ $\quad r_1 = 7/7 = 1.000$

$X_2 = (12*7 + 7) \bmod 8 = 3$ $\quad r_1 = 3/7 = 0.428$

$X_3 = (12*3 + 7) \bmod 8 = 3$ $\quad r_1 = 3/7 = 0.428$

2.2.5 Algoritmo congruencial multiplicativo

- Surge del algoritmo congruencial lineal cuando c = 0
- La ecuación recursiva:

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

Donde: X_0 es la semilla, $X_0 > 0$

a es la constante multiplicativa, $a > 0$

m es el módulo, m > 0. y todos deben ser números enteros

Para la obtención de los números pseudo aleatorios se requiere la sig. ecuación:

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

2.2.5 Algoritmo congruencial multiplicativo

- Banks y otros sugieren que:

  $m = 2^g$

  $a = 3 + 8k$ ó $a = 5 + 8k$

  k debe ser entero

  X_0 debe ser un número impar

  g debe ser entero

- Bajo estas condiciones se obtiene un periodo de vida máximo: $N = 2^{g-2}$.

2.2.5 **Algoritmo congruencial multiplicativo** – Ejemplo

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

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

2.2.5 **Algoritmo congruencial multiplicativo** – Ejemplo

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

$a = 5 + 8(2) = 21$ y m = 32

$X_0 = 17$

$X_1 = (21*17) \bmod 32 = 5$	$r_1 = 5/31 = 0.612$
$X_2 = (21*5) \bmod 32 = 9$	$r_2 = 9/31 = 0.2903$
$X_3 = (21*9) \bmod 32 = 29$	$r_3 = 29/31 = 0.9354$
$X_4 = (21*29) \bmod 32 = 1$	$r_4 = 1/31 = 0.3225$
$X_5 = (21*1) \bmod 32 = 21$	$r_5 = 21/31 = 0.6774$
$X_6 = (21*21) \bmod 32 = 25$	$r_6 = 25/31 = 0.8064$
$X_7 = (21*25) \bmod 32 = 13$	$r_7 = 13/31 = 0.4193$
$X_8 = (21*13) \bmod 32 = 17$	$r_8 = 17/31 = 0.5483$

2.2.5 **Algoritmo congruencial multiplicativo** – Ejemplo

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

2.2.6 **Algoritmo congruencial aditivo**

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

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

Para la obtención de los números pseudo aleatorios se requiere la sig. ecuación:

$$r_i = \frac{X_i}{m-1}$$

2.2.6 **Algoritmo congruencial aditivo**

- Generar 7 números pseudo aleatorios entre cero y uno a partir de la siguiente secuencia de números enteros: 65,89,98,03,69; m = 100.
- Sean $X_1 = 65$, $X_2 = 89$, $X_3 = 98$, $X_4 = 03$, $X_5 = 69$. Para generar r_1, r_2, r_3, r_4, r_5, r_6 y r_7 antes es necesario generar X_6, X_7, X_8, X_9, X_{10}, X_{11}, X_{12}

2.2.6 Algoritmo congruencial aditivo

- Generar 7 números pseudo aleatorios entre cero y uno a partir de la siguiente secuencia de números enteros: 65,89,98,03,69; *m* = 100.
- Sean $X_1 = 65$, $X_2 = 89$, $X_3 = 98$, $X_4 = 03$, $X_5 = 69$. Para generar r_1, r_2, r_3, r_4, r_5, r_6 y r_7 antes es necesario generar X_6, X_7, X_8, X_9, X_{10}, X_{11}, X_{12}

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

2.2.7 Algoritmos congruenciales no lineales – AC Cuadrático

- Este algoritmo tiene la siguiente ecuación recursiva:

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

Para la obtención de los números pseudo aleatorios se requiere la sig. ecuación:

$$r_i = \frac{X_i}{m - 1}$$

2.2.7 Algoritmos congruenciales no lineales – AC Cuadrático

- De acuerdo con L'Ecuyer, las condiciones que deben cumplir los parámetros *m, a, b y c* para alcanzar un periodo máximo de $N = m$ son:

$m = 2^g$

a debe ser un número par

c debe ser un número impar

g debe ser entero

$(b - 1) \bmod 4 = 1$

2.2.7 Algoritmos congruenciales no lineales – AC Cuadrático – ejemplo

- Generar, a partir del algoritmo congruencial cuadrático, suficientes números enteros hasta alcanzar el periodo de vida, considerando los parámetros $X_0 = 13$, m = 8, $a = 26$, b = 27 y c = 27. Como todas las condiciones estipuladas para los parámetros se satisfacen, es de esperarse que el periodo de vida del generador sea N = m = 8.

.

2.2.7 Algoritmos congruenciales no lineales – AC Cuadrático – ejemplo

- Generar, a partir del algoritmo congruencial cuadrático, suficientes números enteros hasta alcanzar el periodo de vida, considerando los parámetros $X_0 = 13$, m = 8, $a = 26$, b = 27 y c = 27. Como todas las condiciones estipuladas para los parámetros se satisfacen, es de esperarse que el periodo de vida del generador sea N = m = 8.

$X_1 = (26*13^2 + 27*13 + 27) \bmod (8) = 4$
$X_2 = (26*4^2 + 27*4 + 27) \bmod (8) = 7$
$X_3 = (26*7^2 + 27*7 + 27) \bmod (8) = 2$
$X_4 = (26*2^2 + 27*2 + 27) \bmod (8) = 1$
$X_5 = (26*1^2 + 27*1 + 27) \bmod (8) = 0$
$X_6 = (26*0^2 + 27*0 + 27) \bmod (8) = 3$
$X_7 = (26*3^2 + 27*3 + 27) \bmod (8) = 6$
$X_8 = (26*6^2 + 27*6 + 27) \bmod (8) = 5$
$X_9 = (26*5^2 + 27*5 + 27) \bmod (8) = 4$

2.2.7 Algoritmos congruenciales no lineales – Alg de Blum, Blum y Shub

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

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

$m = p*q$, números primos muy grandes y congruentes a 3 (mod 4)

Para la obtención de los números pseudo aleatorios se requiere la sig. ecuación:

$$r_i = \frac{X_i}{m-1}$$

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

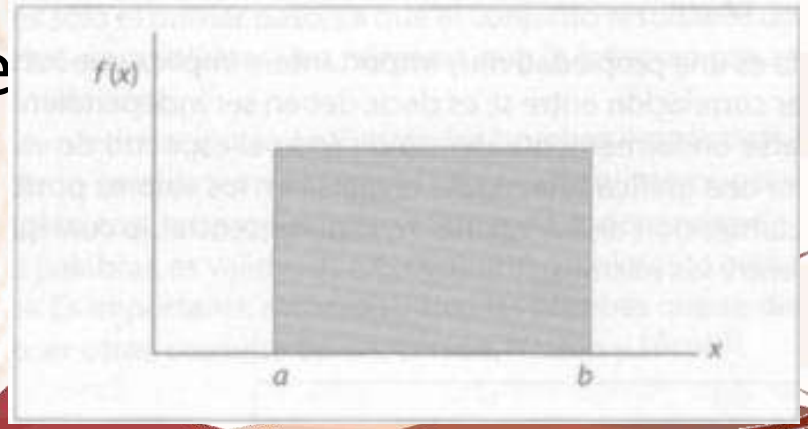
2.3 Propiedades de los números pseudo aleatorios entre 0 y 1

- ¿De qué manera se puede garantizar que tales números son realmente aleatorios entre 0 y 1?
- ¿Cuáles son las características que los identifican?
- ¿Cuáles son sus parámetros?
- En gran medida, conocer las propiedades que deben tener estos números aleatorios garantiza una buena simulación, por ello, se enumeran a continuación.

2.3 Propiedades de los números pseudo aleatorios entre 0 y 1

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

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



2.3 Propiedades de los números pseudo aleatorios entre 0 y 1

- **Media de los aleatorios entre 0 y 1.** Para obtener la media de la distribución se multiplica la función de densidad por *x*, y se integra en el rango de la misma distribución:

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

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

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

Por lo tanto, el valor esperado es : $\mu = 0.5$

2.3 Propiedades de los números pseudo aleatorios entre 0 y 1

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

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

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

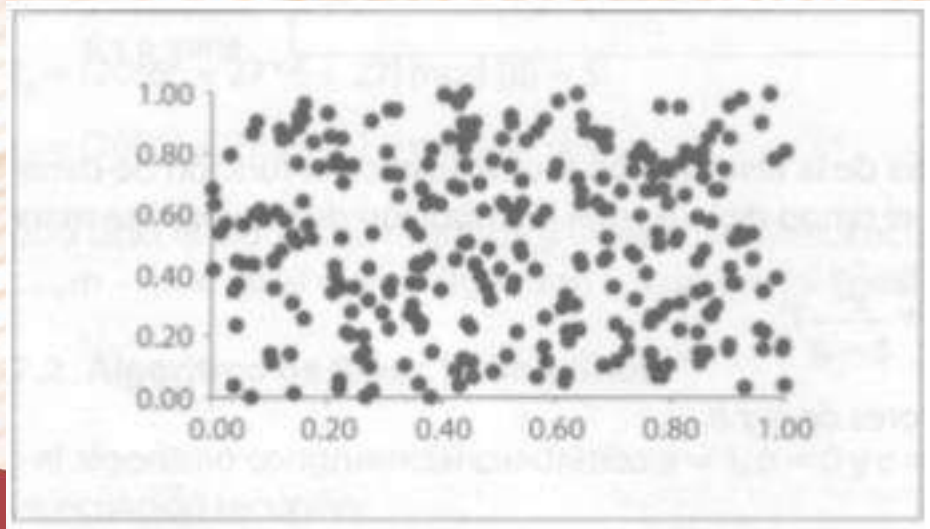
Por lo tanto: $V(x) = \frac{1}{3} - \left(\frac{1}{2}\right)^2 = \frac{1}{12}$

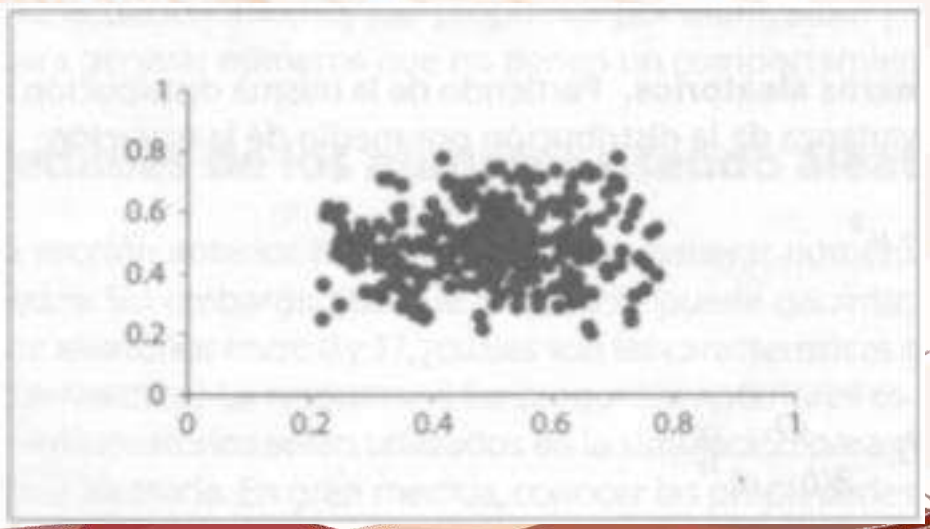
De estos resultados podemos decir que los números aleatorios entre 0 y 1 deben tener:

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

2.3 Propiedades de los números pseudo aleatorios entre 0 y 1

- **Independencia.** Ésta es una propiedad muy importante, e implica que los números aleatorios no deben tener correlación entre sí; es decir, deben ser independientes, de manera que puedan dispersarse uniformemente





2.4 Pruebas estadísticas para los números pseudoaleatorios

- **Prueba de medias.** consiste en determinar el promedio de los *n* números que contiene el conjunto r_i, mediante la ecuación siguiente: $$\bar{r} = \frac{1}{n}\sum_{i=1}^{n} r_i$$
- Después se calculan los límites de aceptación inferior y superior:
$$LI_{\bar{r}} = \frac{1}{2} - z_{\frac{\alpha}{2}}\left(\frac{1}{\sqrt{12n}}\right) \qquad LS_{\bar{r}} = \frac{1}{2} + z_{\frac{\alpha}{2}}\left(\frac{1}{\sqrt{12n}}\right)$$
- Si el valor de r se encuentra entre los límites de aceptación, no se puede rechazar que el conjunto r_i que tiene un valor esperado de 0.5 con un nivel de aceptación de 1 - α. En caso contrario se rechaza. Para el cálculo de los límites de aceptación se utiliza el estadístico Zα / 2, el cual se determina por medio de la tabla de distribución normal estándar

2.4 Pruebas estadísticas para los números pseudoaleatorios

- **Prueba de medias.** Ejemplo: Considere los 40 números del conjunto r_i que se genere aleatoriamente, y determine si tienen un valor esperado de ½ con un nivel de aceptación de 95%.

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

2.4 Pruebas estadísticas para los números pseudoaleatorios

- **Prueba de Varianza.** consiste en determinar la varianza de los *n* números que contiene el conjunto r_i, mediante la ecuación siguiente:

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

- Después se calculan los límites de aceptación inferior y superior:

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

- Si el valor de *V(r)* se encuentra entre los limites de aceptación, no se puede rechazar que el conjunto r_i que tiene una varianza de 1/12.

2.4 Pruebas estadísticas para los números pseudoaleatorios

- **Prueba de Varianza.** Realizar la prueba de varianza a los 40 números r_i. del ejemplo anterior.

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

2.4 Pruebas estadísticas para los números pseudoaleatorios

- **Pruebas de Uniformidad: Prueba Chi-cuadrada.**
  1. Dividir el intervalo (0,1) en *m* subintervalos,en donde es recomendable $m = \sqrt{n}$.
  2. Clasificar cada número pseudo aleatorio del conjunto r_i en los *m* intervalos. A la cantidad de números r_i que se clasifican en cada intervalo se le denomina *frecuencia observada* (O_i), y a la cantidad de números r_i que se espera encontrar en cada intervalo se le llama frecuencia esperada (E_i); teóricamente, la E_i es igual *n/m*.
  3. A partir de los valores de O_i y E_i se determina el estadístico χ_0^2 mediante la ecuación.

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

2.4 Pruebas estadísticas para los números pseudoaleatorios

- **Pruebas de Uniformidad: Prueba Chi-cuadrada (cont.).**

4. Si el estadístico $\chi_0{}^2$ es menor al valor de tablas de $\chi^2_{1-\alpha,m-1}$ entonces no se puede rechazar que el conjunto de números r sigue una distribución uniforme. En caso contrario, se rechaza.

2.4 Pruebas estadísticas para los números pseudoaleatorios

- **Pruebas de Uniformidad: Prueba Chi-cuadrada.** Realizar la prueba de varianza a los 100 números r_i. del siguiente conjunto.

0.347	0.832	0.966	0.472	0.797	0.101	0.696	0.966	0.404	0.603
0.993	0.371	0.729	0.067	0.189	0.977	0.843	0.562	0.549	0.992
0.674	0.628	0.055	0.494	0.494	0.235	0.178	0.775	0.797	0.252
0.426	0.054	0.022	0.742	0.674	0.898	0.641	0.674	0.821	0.19
0.46	0.224	0.99	0.786	0.393	0.461	0.011	0.977	0.246	0.881
0.189	0.753	0.73	0.797	0.292	0.876	0.707	0.562	0.562	0.821
0.112	0.191	0.584	0.347	0.426	0.057	0.819	0.303	0.404	0.64
0.37	0.314	0.731	0.742	0.213	0.472	0.641	0.944	0.28	0.663
0.909	0.764	0.999	0.303	0.718	0.933	0.056	0.415	0.819	0.444
0.178	0.516	0.437	0.393	0.268	0.123	0.945	527	0.459	0.652

Intervalo	O_i	$E_i = n/m$	$(E_i - O_i)^2/E_i$

2.4 Pruebas estadísticas para los números pseudoaleatorios

- **Pruebas de Uniformidad: Prueba Chi-cuadrada.** Realizar la prueba de varianza a los 100 números r_i. del siguiente conjunto.

Intervalo	O_i	$E_i = n/m$	$(E_i - O_i)^2/E_i$
0.1	7	10	0.9
0.2	9	10	0.1
0.3	8	10	0.4
0.4	9	10	0.1
0.5	14	10	1.6
0.6	6	10	1.6
0.7	11	10	0.1
0.8	14	10	1.6
0.9	9	10	0.1
1	12	10	0.4

$\chi_0^2 = 6.9$

$\chi^2_{1-\alpha,m-1} = 16.9190$

2.4 Pruebas estadísticas para los números pseudoaleatorios

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

$$\mu_{C_O} = \frac{2n-1}{3} \qquad \sigma^2{}_{C_O} = \frac{16n-29}{90} \qquad Z_0 = \left|\frac{C_0 - \mu_{C_0}}{\sigma_{C_0}}\right|$$

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

2.4 Pruebas estadísticas para los números pseudoaleatorios

- **Pruebas de Independencia: Prueba de Corridas arriba y abajo.**

Considere el siguiente conjunto r_i de 21 números:

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

- La secuencia de unos y ceros se construye de esta manera: se coloca un cero si el número r_i es menor que o igual al número r_i anterior; en caso de ser mayor que el número r_i anterior, se pone un uno. Considerando la secuencia de los 21 números, la secuencia de unos y ceros es:

S = {0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 0, 0, 1, 0, 1, 0, 0, 1, 0, 1}

$|S| = n - 1$, $C_0 = 14$

2.4 Pruebas estadísticas para los números pseudoaleatorios

- **Pruebas de Independencia: Prueba de Corridas arriba y abajo.**

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

0.34	0.83	0.96	0.47	0.79	0.99	0.37	0.72	0.06	0.18
0.67	62	0.05	0.49	0.59	0.42	0.05	0.02	0.74	0.67
0.46	0.22	0.99	0.78	0.39	0.18	0.75	0.73	0.79	0.29
0.11	0.19	0.58	0.34	0.42	0.37	0.31	0.73	0.74	0.21

$C_0 = 24$ $Z_0 = 0.8955$ $Z_{a/2} = 1.\ 9600$

REFERENCIAS O BIBLIOGRAFÍA

1. Altiok, T., & Melamed, B. (2010). Simulation modeling and analysis with Arena. Elsevier.
2. Banks, J. (1999, December). Introduction to simulation. In Proceedings of the 31st conference on Winter simulation: Simulation---a bridge to the future-Volume 1 (pp. 7-13).
3. Dunna, E. G., Reyes, H. G., & Barrón, L. E. C. (2006). Simulación y análisis de sistemas con ProModel. Pearson Educación.

REFLEXIÓN

Los números pseudo aleatorios juegan un papel importante en el desarrollo de modelos de simulación, para generar la aleatoriedad esperada en los sistemas analizados. Existen algoritmos con buen rendimiento en la generación de números pseudoaleatorios, y el avance tecnológico ha permitod contar con periodos de vida suficientes para modelos de simulación con un alto grado de complejidad.

Por su atención …

Gracias

Información de contacto

Dr. Joselito Medina Marín
Instituto de Ciencias Básicas e Ingeniería
Área Académica de Ingeniería y Arquitectura
Teléfono: (+52) 771 717-2000 ext. 4007
Correo electrónico:jmedina@uaeh.edu.mx

Instituto de Ciencias Básicas e Ingeniería

Área Académica de Ingeniería y Arquietctura

Material desarrollado en el

Doctorado en Ciencias en Ingeniería, con Énfasis en Análisis y Modelación de Sistemas