

UN SISTEMA GENERADOR DE NÚMEROS PSEUDO ALEATORIOS

Guillermo Becerra Córdova¹

¹ Universidad Autónoma Chapingo, Dpto. de Preparatoria Agrícola, Área de Física.

E-mail: gllrmbecerra@yahoo.com

México

Abstract

The simulation is a numeric technique to carry out experiments in a computer. These experiments involve certain types of mathematical and logical models that describe systems of business, economic, social, biological, physical or chemical through long periods of time. To be able to carry out a simulation that includes variability inside their events, it is necessary to generate a series of numbers that they are aleatory for themselves, and that their randomness is extrapolated to the simulation pattern that is building. The aleatory numbers constitute the main part of the simulation of stochastic processes and they are generally used to generate the behavior of aleatory variables, so much continuous as discrete. Because it is not possible to generate really aleatory numbers, we consider them as pseudo aleatory numbers, generated by means of algorithms deterministic that require outburst parameters. In this work a system is presented that generates pseudo aleatory numbers by means of different algorithms. The algorithms that are used in the system are the congruential and the non congruential. To verify that they are really independent and uniform the generated numbers, the system has routines that make possible to subject them to a series of tests. If the pseudo aleatory numbers overcome the tests, they will be able to use in some simulation; otherwise they won't be considered. To use the numbers that have been generated and they have been accepted, the system has an option to export them to some calculation leaf.

Keywords. Simulation, pseudo aleatory numbers, stochastic processes, continuous and discrete variables.

Resumen

La simulación es una técnica numérica para realizar experimentos en una computadora. Estos experimentos involucran ciertos tipos de modelos matemáticos y lógicos que describen sistemas de negocios, económicos, sociales, biológicos, físicos o químicos a través de largos períodos de tiempo. 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. Los números aleatorios constituyen la parte principal de la simulación de procesos estocásticos y generalmente se usan para generar el comportamiento de variables aleatorias, tanto continuas como discretas. Debido a que no es posible generar números realmente aleatorios, los consideramos como números pseudo aleatorios, generados por medio de algoritmos determinísticos que requieren parámetros de arranque. En este trabajo se presenta un sistema que genera números pseudo aleatorios por medio de diferentes algoritmos. Los algoritmos que se utilizan en el sistema son los congruenciales y los no congruenciales. Para verificar que realmente son independientes y uniformes los números generados, el sistema tiene rutinas que hacen posible someterlos a una serie de pruebas. Si los números pseudo aleatorios superan las pruebas, podrán utilizarse en alguna simulación; de lo contrario no se deberán considerar. Para utilizar los números que se han generado y han sido aceptados, el sistema tiene una opción para exportarlos a alguna hoja de cálculo.

Palabras Clave. Simulación, números pseudo aleatorios, procesos estocásticos, variables continuas y discretas.

Introducción

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. En la construcción de un modelo de simulación, los números aleatorios juegan un papel importante.

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

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

Para realizar una simulación se requieren números pseudo aleatorios en el intervalo $(0, 1)$, es decir, una secuencia de números $r_i = \{r_1, r_2, r_3, \dots, r_n\}$ que contenga n números, todos ellos diferentes; n recibe el nombre de período o ciclo de vida del generador que creó la secuencia.

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

Dada la importancia de contar con un conjunto de números pseudo aleatorios suficientemente grande, en este trabajo se presentan diferentes algoritmos determinísticos para obtenerlos. Por otra parte, es conveniente señalar que el conjunto de números pseudo aleatorios debe ser sometido a una variedad de pruebas para verificar si son realmente independientes y uniformes. Si las pruebas son superadas, podrán utilizarse en la simulación; de lo contrario, simplemente debemos desecharlos.

Generar un conjunto de números pseudo aleatorios es una tarea relativamente sencilla, sólo es necesario diseñar un algoritmo de generación. Lo que resulta difícil es diseñar un algoritmo que genere un conjunto de números pseudo aleatorios con período de vida suficientemente grande y además pase sin problemas las pruebas de uniformidad e independencia, lo cual implica evitar problemas como éstos:

- Que los números pseudo aleatorios no estén uniformemente distribuidos, es decir, que haya demasiados números en un subintervalo y en otro muy pocos o ninguno.
- Que los números pseudo aleatorios generados sean discretos en lugar de continuos.

- Que la media del conjunto sea muy alta o muy baja, es decir, que esté por arriba o por debajo de $\frac{1}{2}$.
- Que la varianza del conjunto sea muy alta o muy baja, es decir, que se localice por arriba o por debajo del $\frac{1}{12}$.

Material y Métodos

A continuación se presentan diferentes algoritmos determinísticos para generar números pseudo aleatorios, los cuales se clasifican en congruenciales y no congruenciales.

Algoritmos no congruenciales

Los algoritmos no congruenciales que se incluyen en este proyecto son cuadrados medios, productos medios y multiplicador constante.

Algoritmo de cuadrados medios

Este algoritmo no congruencial fue propuesto en la década de los cuarenta del siglo XX por Von Neumann y Metropolis. Requiere un número entero detonador con D dígitos, el cual es elevado al cuadrado para seleccionar del resultado los D dígitos del centro; el primer número se determina simplemente anteponiendo el "0." a esos dígitos. Para obtener el segundo número se sigue el mismo procedimiento, solo que ahora se eleva al cuadrado los D dígitos del centro que se seleccionaron para obtener el primer número.

Este método se repite hasta obtener n números. A continuación se presentan con más detalle los pasos para generar números con el algoritmo de cuadrados medios.

1. Seleccionar la 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 $i = 1, 2, 3, \dots, n$.
4. Repetir el paso 3 hasta obtener los n números r_i deseados.

El algoritmo de cuadrados medios generalmente es incapaz de generar una secuencia de números con período de vida n grande. Además, en ocasiones solo es capaz de generar un solo número.

Algoritmo de productos medios

La mecánica de generación de números pseudo aleatorios de este algoritmo no congruencial es similar a la del algoritmo de cuadrados medios. La diferencia entre ambos radica en que el algoritmo de productos medios requiere dos semillas, ambas con D dígitos; además, en lugar de elevarlas al cuadrado, las semillas se multiplican y del producto se seleccionan los D dígitos del centro, los cuales formarán el primer número pseudo aleatorio $r_1 = 0.D$. Después se elimina una semilla y la otra se multiplica por el primer número de D dígitos, para luego

seleccionar del producto los D dígitos que conformarán un segundo número r_i . Entonces se elimina la segunda semilla y se multiplican el primer número de D dígitos por el segundo número de D dígitos; del producto se obtiene el tercer número r_i . Siempre se irá eliminando el número más antiguo, y el procedimiento se repetirá hasta generar los n números pseudo aleatorios. A continuación se presentan con más detalle los pasos del método para generar números con el algoritmo de productos medios.

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

Algoritmo de multiplicador constante

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

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

Algoritmos congruenciales

Entre los algoritmos congruenciales se encuentran los algoritmos congruenciales lineales y los no lineales.

Algoritmos congruenciales lineales

En este trabajo abordaremos los algoritmos congruenciales lineales como el algoritmo congruencial lineal, el multiplicativo y el aditivo.

Algoritmo Lineal

Este algoritmo congruencial fue propuesto por D. H. Lehmer en 1955. Según Law y Kelton, este algoritmo ha sido el más utilizado. El algoritmo congruencial lineal genera una secuencia de números enteros por medio de la siguiente ecuación recursiva:

$$X_{i+1} = (a X_i + c) \bmod(m)$$

Con $i = 1, 2, 3, \dots, n$

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

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

Con $i = 1, 2, 3, \dots, n$

Para que el algoritmo sea capaz de lograr el máximo período de vida n , es preciso que los parámetros X_0 , a , y m cumplan con ciertas condiciones. Banks, Carson, Nelson y Nicol sugieren lo siguiente: m debe ser múltiplo de 2^g , donde g debe ser entero, $a = 1 + 4k$, donde k debe ser entero y c debe ser relativamente primo a m .

Bajo estas condiciones se obtiene un período de vida máximo: $N = m = 2^g$.

Algoritmo congruencial multiplicativo

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

$$X_{i+1} = (a X_i) \bmod(m)$$

Con $i = 0, 1, 2, 3, \dots, n$

En comparación con el algoritmo congruencial lineal, la ventaja del algoritmo multiplicativo es que implica una operación menos a realizar. Los parámetros de arranque de este algoritmo son

X_0 , a y m , los cuales deben ser enteros y mayores que cero. Para transformar los números X_i en el intervalo (0, 1) se usa la ecuación:

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

Con $i = 0, 1, 2, 3, \dots, n$

De acuerdo con Banks, Carson, Nelson y Nicol, las condiciones que deben cumplir los parámetros para que el algoritmo congruencial multiplicativo alcance su máximo período son:

m debe ser múltiplo de 2^g , donde g debe ser entero, $a = 3 + 8k$, donde $k = 0, 1, 2, 3, \dots$, X_0 debe ser un número impar.

Bajo estas condiciones se logra un período de vida máximo: $N = k / 4 = 2^{g-2}$.

Algoritmo congruencial aditivo

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

$$X_i = (X_{i-1} + X_{i-2}) \bmod(m)$$

Con $i = n+1, n+2, n+3, \dots, N$

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

Algoritmos congruenciales no lineales

Dentro de los algoritmos congruenciales no lineales se tiene el algoritmo congruencial cuadrático y el de Blum, Blum, y Shub.

Algoritmo congruencial cuadrático

Este algoritmo tiene la ecuación recursiva:

$$X_{i+1} = (a X_i^2 + b X_i + c) \bmod(m)$$

Con $i = 0, 1, 2, 3, \dots, n$

En este caso, los números r_i pueden ser generados por la ecuación

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

De acuerdo con L'Ecuyer, las condiciones que deben cumplir los parámetros m , a , b y c para alcanzar un período máximo de $N = m$ son: m debe ser múltiplo de 2^g , donde g debe ser entero, a debe ser un número par, m debe ser un número impar, y $(b-1) \bmod 4 = 1$. De esta manera se logra un período de vida máximo $N = m$.

Algoritmo 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:

$$X_{i+1} = (X_i^2) \bmod(m)$$

Con $i = 0, 1, 2, 3, \dots, n$

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

PRUEBAS ESTADÍSTICAS PARA LOS NÚMEROS PSEUDO ALEATORIOS

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

A continuación se presentan las pruebas estadísticas básicas que se emplean generalmente para determinar si un conjunto de números pseudo aleatorios entre cero y uno cumplen con las propiedades básicas de independencia y uniformidad. El objetivo es validar que el conjunto realmente está conformado por números aleatorios.

Prueba de Medias

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

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

y

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

La prueba de medias consiste en determinar el promedio de los n números generados mediante la ecuación siguiente:

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

Posteriormente, se calcula los límites inferior y superior con las ecuaciones siguientes:

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

y

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

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

de los límites de aceptación se utiliza el estadístico $Z_{\alpha/2}$, el cual se determina por medio de la tabla de distribución normal estándar.

Prueba de Varianza

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

$$H_0 : \sigma_{r_i}^2 = 0.5$$

y

$$H_1 : \sigma_{r_i}^2 \neq 0.5$$

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

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

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

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

y

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

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

Prueba Chi-cuadrada

Una de las propiedades más importantes que debe cumplir un conjunto de números r_i es la uniformidad. Para comprobar su acatamiento se han desarrollado pruebas estadísticas como la prueba Chi-cuadrada. Para probar la uniformidad de los números de un conjunto bajo esta prueba es necesario formular las siguientes hipótesis:

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

y

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

La prueba Chi-cuadrada busca determinar si los números del conjunto r_i se distribuyen uniformemente en el intervalo $(0,1)$. Para llevar a cabo esta prueba es necesario dividir el intervalo en m subintervalos en donde es recomendable que $m = \sqrt{n}$. Posteriormente se clasifica 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 llama frecuencia esperada (E_i); teóricamente, la r_i es igual a n/m . A partir de los valores de O_i y de E_i se determina el estadístico χ_0^2 mediante la ecuación:

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

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

Resultados y Discusión

En la figura 1 se presenta la ventana principal del sistema que se elaboró como resultado del proyecto. En ella se muestra el nombre del sistema, el lugar donde se realizó y el autor. Si el usuario desea abandonar el sistema, simplemente es necesario hacer click en el botón de comando **Salir**. Si desea continuar sólo se requiere hacer click en botón **Continuar**.



Figura 1

Al activar esta opción, aparecerá una ventana similar a la venta que se muestra en la figura 2. En ella se encuentran las diversas formas como se pueden generar números pseudo aleatorios. Los algoritmos que se incluyen en el sistema son: cuadrados medios, productos medios, multiplicador constante, lineal, congruencial multiplicativo, congruencial aditivo, congruencial cuadrático y Blum, Blum y Shub.

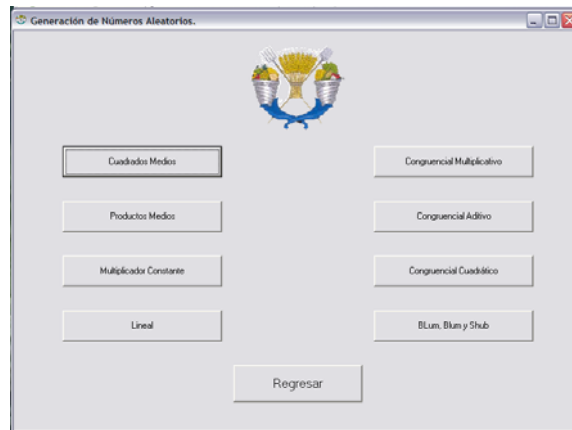


Figura 2

Al hacer click en la opción **Cuadrados Medios** aparecerá una ventana como la mostrada en la figura 3. En ella se observa la ventana de despliegue de los números pseudo aleatorios generados por el correspondiente algoritmo, la caja de texto para introducir la semilla, el botón de que activa la generación de los números pseudo aleatorios y el botón para encontrar el período relacionado con los números pseudo aleatorios. De igual forma se muestra la caja de texto para introducir el nivel de aceptación y la caja de texto para desplegar el promedio que es calculado por el sistema. Al activar la opción **Prueba de Medias** el sistema desplegará si son aceptados o rechazados los números generados. El siguiente cuadro de texto despliega la varianza de los números pseudo aleatorios y los siguientes dos cuadros de texto se utilizan para introducir los límites inferior y superior para calcular la prueba de varianza. El cuadro de texto correspondiente a esta muestra, despliega si la prueba fue aceptada o rechazada. Finalmente, aparecen otras tres cajas de texto que se utilizan para introducir el valor de la prueba **Chi-Cuadrada Teórica**. La caja de texto siguiente sirve para desplegar la prueba **Chi-Cuadrada Experimental** que es calculada por el sistema y finalmente aparece una caja de texto que muestra si la prueba es aceptada o rechazada bajo estos límites.

Cuadrados Medios:

0.8902
0.2496
0.0319
0.0176
0.3050
0.5976
0.7126
0.7759
0.8244
0.9635

Introducir Semilla: 5735

Exportar Datos

Regresar

Generar Números

Encontrar el Período: No se repite la numeración

Nivel de Aceptación: 1.36

Promedio: 0.4932

Prueba de Medias: No se rechaza

Varianza: 0.0987

Límite Inferior: 32.56

Límite Superior: 71.42

Prueba de Varianza: Se rechaza

Chi-Cuadrada Teórica: 15.9

Chi-Cuadrada Exp: 10.4

Prueba Chi-cuadrada: No se rechaza

Figura 3

La figura 3 muestra, a manera de ejemplo, los resultados de una generación de números pseudo aleatorios por medio del algoritmo de cuadrados medios. En la figura aparecen los valores de los números pseudo aleatorios generados con el algoritmo de cuadrados medios. Para este caso, el valor de la semilla es de 5735. La primera caja de texto nos indica que no se repite la numeración, por lo que no tiene período. La segunda caja de texto corresponde al valor de la tabla de distribución normal estándar bajo la prueba de medias. Para ello, el usuario

debe introducir el valor de la tabla de distribución normal estándar al que quiere someter los números generados. Se introdujo el valor de 1.96 por el nivel de aceptación del 95%. Después de introducir este valor, para saber si son aceptados los números pseudo aleatorios, el usuario debe hacer click en el botón de comando **Prueba de Medias**. En el ejemplo se muestra que no se rechaza la muestra. Si se redujera el valor del nivel de aceptación o el valor de la tabla de distribución normal estándar, es posible que no se acepten los números generados. El siguiente bloque de opciones corresponde a la prueba de varianza. Para ello, el usuario debe introducir el valor de las tablas de la distribución Ji-cuadrada correspondiente a un nivel de aceptación del 95% y un grado de libertad de 49, ya que son 50 los números generados. Al activar la opción **Prueba de Varianza** se puede ver si se rechazan o no los números generados. Para este caso se tiene que $\alpha = 5\%$ y $n = 50$, por lo que los valores de la tabla Ji-cuadrada son: $\chi^2_{0.025, 49} = 71.42$ y $\chi^2_{0.975, 49} = 32.36$. Al calcular la varianza de los números generados pseudo aleatoriamente, se observa que el valor no se encuentra dentro de los límites calculados por el sistema, por lo que no son aceptados. De igual forma, es posible aumentar el intervalo de prueba para ver si se rechaza o no la serie. Finalmente, el último bloque corresponde a la prueba Chi-cuadrada, en la cual el usuario puede introducir el valor de la **Chi-Cuadrada Teórica** para compararla con el valor de la **Chi-Cuadrada Experimental** para saber si son aceptados los valores de los números pseudo aleatorios generados por medio de este algoritmo. Al activar la opción **Chi-Cuadrada** se puede ver si se rechazan o no los números generados. En este caso se tiene que el valor de la prueba Chi-cuadrada teórica para estos datos es de $\chi^2_{0.05, 9} = 16.9$, que comparado con el valor del Chi-cuadrada experimental $\chi^2_0 = 10.4$, concluimos que se aceptan los números pseudo aleatorios generados por el algoritmo de cuadrados medios. Al igual que en los casos anteriores, se puede reducir el valor de **Chi-cuadrada Teórica**, hasta que sean rechazados los números generados pseudo aleatoriamente.

Todas las opciones descritas para el algoritmo de **Cuadrados Medios** se encuentran en cada uno de los algoritmos de generación de números pseudo aleatorios contenidos en el sistema, por lo que cada algoritmo puede comprobar si los números generados son aceptados o no.

Después de generarse los números pseudo aleatorios, el usuario puede exportarlos a una hoja de cálculo para poder ser utilizados en alguna simulación.

Conclusiones

- El sistema puede generar números pseudo aleatorios por medio de diferentes algoritmos congruenciales y no congruenciales.
- Para cada algoritmo, el sistema incluye las pruebas de Medias, de Varianza y de Chi cuadrada para saber si los números generados son aceptados o rechazados.
- El sistema tiene la opción de exportar los números pseudo aleatorios a una hoja de cálculo para que sean utilizados en alguna simulación.
- El sistema puede calcular el período de la secuencia de números pseudo aleatorios generados en cada algoritmo.
- El sistema presenta una interface gráfica de fácil manejo.
- El sistema es un apoyo a la labor docente.

Referencias

- Ceballos Sierra, F. J. 1996. *Enciclopedia de Visual Basic 4*, México, D. F.: Alfaomega Grupo Editor.
- García Dunna E.; Gracia Reyes, H. y Cárdenas Barrón, L. E. 2006. *Simulación y Análisis de Sistemas con ProModel*, México, D. F.: Pearson Educación.
- García Mora, F.; Sierra Acosta, J. y Guzmán Ibarra, V. 2005. *Simulación de Sistemas para Administración e Ingeniería*, México, D. F.: CECSA.
- Kreyszig, Erwin. 1978. *Introducción a la Estadística Matemática*, México, D. F.: Editorial Limusa.