



SIMULACION DE EVENTOS

CLAVE : LII 210

PROFESOR: MTRO. ALEJANDRO SALAZAR GUERRERO

1. LA SIMULACIÓN DE EVENTOS DISCRETOS.

- 1.1 Definición de Eventos Discretos.
- 1.2 Estructura de la simulación de eventos discretos.
- 1.3 Característica de la simulación de eventos discretos.
- 1.4 Sistemas.
- 1.5 Modelos.
- 1.6 Control.
- 1.7 Mecanismos de tiempo fijo.
- 1.8 Metodología
 - 1.8.1 Formulación del Problema.
 - 1.8.2 Recolección de datos.
 - 1.8.3 Desarrollo del modelo.
 - 1.8.4 Verificación.
 - 1.8.5 Validación.
 - 1.8.6 Experimentación de resultados.
 - 1.8.7 Optimización de resultados.

2. NÚMEROS ALEATORIOS Y PSEUDOALEATORIOS.

- 2.1 Números aleatorios definición propiedades.
 - 2.1.1 Generadores.
 - 2.1.2 Tablas.
- 2.2 Números pseudoaleatorios propiedades.
 - 2.2.1 Técnicas para generar números pseudoaleatorios
 - 2.2.1.1 Métodos de centros al cuadrado.
 - 2.2.1.2 Métodos de congruencia.
 - 2.2.1.3 Multiplicativo.
 - 2.2.1.4 Mixto.
- 2.3 Pruebas de aleatoriedad.
- 2.4 Método de Montecarlo.
 - 2.4.1 Simulación de procesos aleatorios.
 - 2.4.2 Usando números.
 - 2.4.3 Manuales.
 - 2.4.4 Lenguajes de propósito general como.
 - 2.4.4.1 C, C++.
 - 2.4.4.2 Delphi.
 - 2.4.4.3 Visual.
 - 2.4.5 Sistemas productivos.
 - 2.4.6 Calidad.
 - 2.4.7 Inventarios.
 - 2.4.8 Económicos.

3. VARIABLES ALEATORIAS.

- 3.1 Métodos para generar variables aleatorias.
 - 3.1.1 Transformadas Inversas.
 - 3.1.2 Aceptación rechazo.
 - 3.1.3 Convolución.
 - 3.1.4 Directos.

- 3.1.4.1 Generación de variables aleatorias discretas.
- 3.1.4.2 Distribuciones Poisson.
- 3.1.4.3 Binomial.
- 3.1.4.4 Geométrica.
- 3.1.4.5 Generación de variables aleatorias continuas.
- 3.1.4.6 Distribución uniforme.
- 3.1.4.7 Exponencial.
- 3.1.4.8 Normal.
- 3.1.4.9 Erlang.
- 3.1.4.10 Gamma.
- 3.1.4.11 Beta.
- 3.1.4.12 Triangular.
- 3.1.5 Distribuciones empíricas de probabilidad.
- 3.1.6 Simulación de procesos aleatorios manuales.
- 3.1.7 Sistemas productivos.
- 3.1.8 Calidad.
- 3.1.9 Inventarios.
- 3.1.10 Económicos.

4. LENGUAJES DE SIMULACIÓN Y SIMULADORES DE EVENTOS DISCRETOS.

4.1 Lenguajes de simulación y simuladores.

- 4.1.1 Características
- 4.1.2 Aplicación y uso lenguajes
 - 4.1.2.1 SLAM
 - 4.1.2.2 ECSL
 - 4.1.2.3 SIMAN
 - 4.1.2.4 GPSS
- 4.1.3 Simuladores
 - 4.1.3.1 PROMODEL
 - 4.1.3.2 TAYLOR ED
 - 4.1.3.3 ARENA
 - 4.1.3.4 WITNESS

4.2 Aprendizaje y uso de un simulador

- 4.2.1 Características del Software
- 4.2.2 Elementos del modelo
- 4.2.3 Menús principales
- 4.2.4 Construcción del modelo
- 4.2.5 El uso del simulador de problemas aplicados a servicios
- 4.2.6 Sistemas productivos
- 4.2.7 Calidad
- 4.2.8 Inventarios
- 4.2.9 Económicos

CAP. 2. NÚMEROS ALEATORIOS Y PSEUDOALEATORIOS.

Los números aleatorios son elementos que se emplean dentro del proceso de simulación, con el propósito de evaluar el grado de respuesta de un sistema a los eventos fortuitos o de baja predictibilidad.

Un problema básico que nos encontramos habitualmente es el de obtener secuencias de números uniformemente distribuidos en un intervalo $[0,1]$

Las diferentes posibilidades para resolver dicho problema son:

- i) Buscar en tablas de números aleatorios publicadas (libros, internet...)
- ii) Observar un proceso físico tal como la desintegración radiactiva, el ruido eléctrico
- iii) Los lenguajes de programación y las hojas electrónicas incluyen una función para generarlos
- iv) Mediante algoritmos de generación de números aleatorios

Las principales ventajas de los generadores de números aleatorios son:

- Rapidez
- Comodidad
- Reproducibilidad
- Portabilidad

Y la desventaja fundamental:

- Las secuencias obtenidas no son realmente aleatorias, ya que se obtienen con operaciones deterministas. Solo podemos obtener secuencias *pseudo-aleatorias*, que a su vez satisfacen algunos criterios de aleatoriedad adecuados.

Los números generados deben cumplir ciertas características para que sean válidos. Dichas características son:

1. Uniformemente distribuidos.
2. Estadísticamente independientes.
3. Su media debe ser estadísticamente igual a $1/2$.
4. Su varianza debe ser estadísticamente igual a $1/12$.
5. Su periodo o ciclo de vida debe ser largo.
6. Deben ser generados a través de un método rápido.
7. Generados a través de un método que no requiera mucha capacidad de almacenamiento de la computadora.

Normalmente se utilizan números enteros, ya que su aritmética es exacta y rápida. Se generan enteros N_i entre 0 y $M - 1$, y $x_i = N_i / M$ da valores reales en el intervalo $[0,1)$.

En general los algoritmos utilizan relaciones de recurrencia del tipo

$$N_i = f(N_{i-1})$$

En el caso de recurrencia simple, o bien

$$N_i = f(N_{i-1}, \dots, N_{i-k})$$

Para el caso de una recurrencia de orden k .

Se necesitará dar un *valor inicial* para comenzar el algoritmo (k valores para recurrencias de orden k).

Generadores.

Un Generador de números aleatorios es un componente o funcionalidad que crea números o símbolos para un programa software en una forma que carezca de un patrón evidente, y que así parezcan ser números aleatorios.

La mayor parte de los generadores de números aleatorios son, en realidad, pseudoaleatorios: se calcula (o introduce internamente) un valor X_0 , que llamaremos semilla, y, a partir de él, se van generando X_1, X_2, X_3, \dots

Siempre que se parta de la misma semilla, se obtendrá la misma secuencia de valores.

El algoritmo básico es el método congruencial, que genera valores en el intervalo $[0,1)$, mediante el siguiente esquema: Se fijan A, B , enteros positivos (deben tener ciertas propiedades para obtener un buen generador), y, a partir de una semilla X_0 en el conjunto $0, 1, \dots, (N-1)$, se generan $X_1 = A \cdot X_0 + B \pmod{N}$ $X_2 = A \cdot X_1 + B \pmod{N}$ $X_3 = A \cdot X_2 + B \pmod{N}$... $X_{(k+1)} = A \cdot X_k + B \pmod{N}$... donde $A \cdot X + B \pmod{N}$ es el resto de la división entera de $A \cdot X + B$ entre N . Por ejemplo, $16 \pmod{7}$ es 2.

Tablas.

Es un conjunto de cifras entre 0 y 9 cuyo orden no obedece ninguna regla de formación, ellas se pueden leer individualmente o en grupos y en cualquier orden, en columnas hacia abajo, columnas hacia arriba, en fila, diagonalmente, si se desea formar números aleatorios en un determinado rango, basta con calcular la proporción, otra forma de usarlo es sumando dos números tomados de alguna posición o multiplicarlos.

Para ser presentadas estas cifras se agrupan en números de 4 dígitos, formando bloques de 5 filas y 10 columnas facilitando de esta forma su lectura que puede iniciarse desde cualquier parte de la tabla.

Una tabla de números aleatorios es útil para seleccionar al azar los individuos de una población conocida que deben formar parte de una muestra.

Números pseudoaleatorios propiedades.

Un número pseudo-aleatorio es un número generado en un proceso que parece producir números al azar, pero no lo hace realmente. Las secuencias de números pseudo-aleatorios no muestran ningún patrón o regularidad aparente desde un punto de vista estadístico, a pesar de haber sido generadas por un algoritmo completamente determinista, en el que las mismas condiciones iniciales producen siempre el mismo resultado.

Los mecanismos de generación de números aleatorios que se utilizan en la mayoría de los sistemas informáticos son en realidad procesos pseudo-aleatorios.

Una Sucesión de números pseudoaleatorios es una sucesión de números que ha sido obtenida mediante un proceso aritmético definido, pero que es una secuencia de números aleatorios efectiva para el propósito para el que se la requiere.

Si bien una sucesión de números pseudoaleatorios parece generalmente no obedecer a ningún patrón o ley de formación, todo generador de números pseudoaleatorios con un estado interior finito, se repetirá luego de una larga sucesión de números.

Técnicas para generar números pseudoaleatorios

Métodos de centros al cuadrado.

Consiste en que cada número de una sucesión es producido tomando los dígitos medios de un número obtenido mediante la elevación al cuadrado.

- P₁: Obtener semilla (valores iniciales 445)
- P₂: Aplicación de Algoritmos recursivos (elevar al cuadrado)
- P₃: Validación del conjunto de datos generados

Ejemplo: Consideremos la semilla 445

X	X ²	N° Aleatorio
445	1 9802 5	0,9802
9802	96 0792 04	0,0792
792	6 2726 4	0,2726
2726

Problemas

- Tendencia a degenerar a cero

Métodos de congruencia.

- Una secuencia de números enteros Z_1, Z_2, Z_3, \dots es definido por la fórmula recursiva.

$$Z_i = (a Z_{i-1} + b) \pmod{m}$$
 - m = modulo;
 - a = multiplicador
 - b = sesgo :
 - Z_c = semilla o valor inicial

Todos enteros no negativos.
Además $0 < m, a < m, b < m, Z_0 < m$.

Para obtener Z_i , dividir a $Z_{i-1} + b$ por m , y dejar que Z_i sea el resto de esta división

Observación:

- 1.- Cuando $b=0$ el generador se denomina Generador congruencial multiplicativo.
- 2.- Cuando $b \neq 0$ el generador se denomina Generador congruencial mixto.
- 3.- A pesar de la simplicidad una adecuada elección de los parámetros de “a, b y m”, permite obtener de manera eficiente una larga e impredecible sucesión de números como para considerarse “aleatoria”.

m =	16	i	Zi	Ui
a =	5	0	7	
c =	3	1	6	0.375
Zo =	7	2	1	0.0625
		3	8	0.5
		4	11	0.6875
		5	10	0.625
		6	5	0.3125
		7	12	0.75
		8	15	0.9375
		9	14	0.875
		10	9	0.5625
		11	0	0
		12	3	0.1875
		13	2	0.125
		14	13	0.8125
		15	4	0.25
		16	7	0.4375
		17	6	0.375
		18	1	0.0625
		19	8	0.5
		20	11	0.6875

Caso	Parámetros			
	a	b	m	xo
1	6	0	13	1
2	7	0	13	10
3	5	0	13	5
4	7	0	11	5
5	6	0	11	3

Caso	Salidas														
	6	10	8	9	2	12	7	3	5	4	11	1	6	10	
1	6	10	8	9	2	12	7	3	5	4	11	1	6	10	
2	5	9	11	12	6	3	8	4	2	1	7	10	5	9	
3	12	8	1	5	12	8	1	5	12	8	1	5	12	8	
4	2	3	10	4	6	9	8	1	7	5	2	3	10	4	
5	7	9	10	5	8	4	2	1	6	3	7	9	10	4	

Algunas observaciones de las salidas de los generadores congruenciales:

- i) Un generador congruencial tiene ciclos
- ii) La longitud del ciclo depende de la selección de los parámetros (ver caso 1) y 3)
- iii) Dentro de selecciones de parámetros que conducen a la misma longitud, algunas salidas parecen más aleatorias que otras.

Pruebas de aleatoriedad.

- Para comprobar si los números aleatorios obtenidos cumplen las propiedades deseadas de uniformidad e independencia se deben realizar una serie de pruebas.
 - Prueba de frecuencia.
 - Pruebas de series.
 - Prueba de autocorrelación.
 - Prueba de saltos.
 - Prueba de póker.
- Cuando se prueba la uniformidad las hipótesis son:
 - $H_0: R_i \sim U[0,1]$
 - $H_1: R_i \neq U[0,1]$
 - La hipótesis nula supone que la secuencia de números obtenidos está distribuida uniformemente en el intervalo $[0, 1]$.

Método de Montecarlo.

El método de Monte Carlo es un método no determinístico o estadístico numérico usado para aproximar expresiones matemáticas complejas y costosas de evaluar con exactitud. El método se llamó así en referencia al Casino de Montecarlo (Principado de Mónaco) por ser "la capital del juego de azar", al ser la ruleta un generador simple de números aleatorios. El nombre y el desarrollo sistemático de los métodos de Monte Carlo datan aproximadamente de 1944 y se mejoraron enormemente con el desarrollo de la computadora.

El uso de los métodos de Monte Carlo como herramienta de investigación, proviene del trabajo realizado en el desarrollo de la bomba atómica durante la segunda guerra mundial en el Laboratorio Nacional de Los Álamos en EE.UU. Este trabajo conllevaba la simulación de problemas probabilísticos de hidrodinámica concernientes a la difusión de neutrones en el material de fusión, la cual posee un comportamiento eminentemente aleatorio. En la actualidad es parte fundamental de los algoritmos de trazado de rayos para la generación de imágenes sintéticas.

Monte Carlo

En la primera etapa de estas investigaciones, John von Neumann y Stanislaw Ulam refinaron esta ruleta rusa y los métodos "de división" de tareas. Sin embargo, el desarrollo sistemático de estas ideas tuvo que esperar al trabajo de Harris y Herman Kahn en 1948. Aproximadamente en el mismo año, Enrico Fermi, Metropolis y Ulam obtuvieron estimadores para los valores característicos de la ecuación de Schrödinger para la captura de neutrones a nivel nuclear usando este método.

El método de Monte Carlo proporciona soluciones aproximadas a una gran variedad de problemas matemáticos posibilitando la realización de experimentos con muestreos de números

pseudoaleatorios en una computadora. El método es aplicable a cualquier tipo de problema, ya sea estocástico o determinista. A diferencia de los métodos numéricos que se basan en evaluaciones en N puntos en un espacio M -dimensional para producir una solución aproximada, el método de Monte Carlo tiene un error absoluto de la estimación que decrece como $\frac{1}{\sqrt{N}}$ en virtud del teorema del límite central.

Simulación de procesos aleatorios.

La técnica de simulación es desde hace mucho tiempo una herramienta importante para el diseñador. Durante muchos años, se han usado modelos a escala de máquinas, para simular la distribución de planta. La simulación común se usó inicialmente en la investigación de operaciones, surgió por primera vez en el trabajo de John Von Neumann y Stanislaw Ulam, en los últimos años de la década de los 40. Quienes a través del análisis de Montecarlo en conjunto con una técnica matemática, resolvieron problemas relacionados con las barreras nucleares de protección, demasiado costosas para someterlas a pruebas de experimentación o demasiado complejas para realizar sus análisis. Con el advenimiento de las computadoras, en los primeros años de la década de los 50, la simulación experimentó un avance substancial. En la actualidad se resuelven incontables problemas de negocios, puesto que la simulación en la computadora es un método económico y rápido para efectuar la vasta cantidad de cálculos que se requieren.

Las áreas de aplicación de la simulación son muy amplias, numerosas y diversas, basta mencionar sólo algunas de ellas: Análisis del impacto ambiental causado por diversas fuentes. Análisis y diseño de sistemas de manufactura. Análisis y diseño de sistemas de comunicaciones. Evaluación del diseño de organismos prestadores de servicios públicos (por ejemplo: hospitales, oficinas de correos, telégrafos, casas de cambio, etc.). Análisis de sistemas de transporte terrestre, marítimo o por aire. Análisis de grandes equipos de cómputo. Análisis de un departamento dentro de una fábrica. Adiestramiento de operadores (centrales carboeléctricas, termoeléctricas, nucleoeléctricas, aviones, etc.). Análisis de sistemas de acondicionamiento de aire. Planeación para la producción de bienes. Análisis financiero de sistemas económicos. Evaluación de sistemas tácticos o de defensa militar. La simulación se utiliza en la etapa de diseño para auxiliar en el logro o mejoramiento de un proceso o diseño o bien a un sistema ya existente para explorar algunas modificaciones. Se recomienda la aplicación de la simulación a sistemas ya existentes cuando existe algún problema de operación o bien cuando se requiere llevar a cabo una mejora en el comportamiento. El efecto que sobre el sistema ocurre cuando se cambia alguno de sus componentes se puede examinar antes de que ocurra el cambio físico en la planta para asegurar que el problema de operación se soluciona o bien para determinar el medio más económico para lograr la mejora deseada. Todos los modelos de simulación se llaman modelos de entrada-salida. Es decir, producen la salida del sistema si se les da la entrada a sus subsistemas interactuantes. Por tanto los modelos de simulación se “corren” en vez de “resolverse”, a fin de obtener la información o los resultados deseados. Son incapaces de generar una solución por sí mismos en el sentido de los modelos analíticos; solos pueden servir como herramienta para el análisis del comportamiento de un sistema en condiciones especificadas por el experimentador. Por tanto la simulación es una teoría, si no una metodología de resolución de problemas. Además la simulación es solo uno de varios planteamientos valiosos para resolver problemas que están disponibles para el análisis de sistemas. Pero ¿Cuándo es útil utilizar la simulación? Cuando existan una o más de las siguientes condiciones: 1.- No existe una completa formulación matemática del problema o los métodos analíticos para resolver el modelo

matemático no se han desarrollado aún. Muchos modelos de líneas de espera corresponden a esta categoría. 2.- Los métodos analíticos están disponibles, pero los procedimientos matemáticos son tan complejos y difíciles, que la simulación proporciona un método más simple de solución. 3.- Las soluciones analíticas existen y son posibles, pero están más allá de la habilidad matemática del personal disponible. El costo del diseño, la prueba y la corrida de una simulación debe entonces evaluarse contra el costo de obtener ayuda externa. 4.- Se desea observar el trayecto histórico simulado del proceso sobre un período, además de estimar ciertos parámetros. 5.- La simulación puede ser la única posibilidad, debido a la dificultad para realizar experimentos y observar fenómenos en su entorno real, por ejemplo, estudios de vehículos espaciales en sus vuelos interplanetarios. 6.- Se requiere la aceleración del tiempo para sistemas o procesos que requieren de largo tiempo para realizarse. La simulación proporciona un control sobre el tiempo, debido a que un fenómeno se puede acelerar o retardar según se desee.

PROCESO DE SIMULACION

Antes de especificar los aspectos más importantes que se presentan al formular problemas de simulación, será útil definir esta. La simulación es la utilización de un modelo de sistemas, que tiene las características deseadas de la realidad, a fin de reproducir la esencia de las operaciones reales. También se le ha definido como una representación de la realidad mediante el empleo de un modelo u otro sistema que reaccione de la misma manera que la realidad, en un conjunto de condiciones dadas. Ninguna de estas definiciones incluye todos los requisitos fundamentales de esta, como son, el uso de los modelos matemáticos, las computadoras, los procesos estadísticos o estocásticos, los casos las suposiciones y los cursos de acción alternativos. La definición más general y amplia de esta: una técnica cuantitativa que utiliza un modelo matemático computarizado para representar la toma real de decisiones bajo condiciones de incertidumbre, con objeto de evaluar cursos alternativos de acción con base en hechos y suposiciones.

La simulación es útil en la resolución de problema de negocios cuando no se conocen parcialmente las variables con anticipación y no existe una manera fácil de encontrar estos valores. El problema se parece que al de la secuencia, para la cual no se conoce una fórmula ya elaborada por encontrar el enésimo (o último) término. El único hecho conocido es una regla (relación recursiva) que permite encontrar el siguiente término a partir de los últimos. Básicamente, la única manera de descubrir el enésimo término es aplicando la misma regla una y otra vez hasta llegar a dicho término. La simulación utiliza un método para encontrar estos estados sucesivos en un problema, aplicando repetidamente las reglas bajo las que operan el sistema. Este eslabonamiento sucesivo de un estado particular con otros anteriores es una característica importante de la simulación.

Generalmente, la simulación implica la construcción de un modelo matemático que describa el funcionamiento de sistema en cuanto a eventos y componentes individuales. Además, el sistema se divide en los elementos y las interrelaciones de aquellos elementos de comportamiento previsible, al menos en función de una distribución de probabilidades, para cada uno de los diversos estados del sistema y sus insumos. La simulación es un medio de dividir el proceso de elaboración de modelos en parte componentes más pequeñas y combinarlas en el orden natural o lógico. Lo que permite el análisis en computadoras de los efectos de las interacciones mutuas entre esta. Debido al error estadístico, es imposible garantizar que se encontrará la respuesta óptima, no obstante la respuesta estará por lo menos próxima a la óptima si el problema se simula correctamente. En esencia, el modelo de simulación realiza experimentos sobre los datos de una

muestra más que sobre el universo entero, ya que esto sería demasiado tardado, inconveniente y costoso.

Planificar un proceso de simulación requiere de los siguientes pasos:

- a) Formulación del problema.
- b) Recolección y procesamiento de la información requerida.
- c) Formulación del modelo matemático.
- d) Evaluación de las características de la información procesada.
- e) Formulación de un programa de computadora.
- f) Validación del programa de computadora.
- g) Diseño de experimentos de simulación.
- h) Análisis de resultados y validación de la simulación.

A continuación se resumen las principales características asociadas a cada paso.

FORMULACIÓN DEL PROBLEMA

Generalmente un problema se presenta por síntomas, no por el diagnóstico. Por lo que antes de generar soluciones en un sistema, se deben buscar el mayor número de síntomas.

Según Acoff y Sasieni, las condiciones para que exista el más simple de los problemas son:

Debe existir por lo menos un individuo que se encuentra dentro de un marco de referencia, el cual se puede atribuir el problema del sistema.

El individuo debe tener por lo menos un par de alternativas para resolver su problema, en caso contrario no existe tal problema.

Deben de existir por lo menos, un par de soluciones, una de las cuales debe tener mayor aceptación que la otra en el individuo. En caso contrario, no existe el problema. Esta preferencia está asociada a un cierto objetivo dentro del marco de referencia en donde se encuentra el individuo del sistema.

La selección de cualquiera de las soluciones debe repercutir de manera diferente en los objetivos del sistema, es decir existe una eficiencia y/o efectividad asociada con cada solución. Estas eficiencias y/o efectividades deben ser diferentes, puesto que de lo contrario no existe problema.

Por último el individuo que toma las decisiones ignora las soluciones y/o eficiencia y/o efectividades asociadas con las soluciones del problema.

Si las cinco condiciones anteriores existen, entonces se tiene un problema. Esta situación puede complicarse en los siguientes casos:

El problema recae en un grupo, no en un individuo.

El marco de referencia donde se encuentra el grupo, cambia en forma dinámica.

El numero de alternativas que el grupo puede escoger es bastante grande, pero finito.

El grupo dentro del sistema puede tener objetivos múltiples. Peor aun, no necesariamente estos objetivos son consistentes entre si.

Las alternativas que selecciona el grupo son ejecutadas por otro grupo ajeno, al cual no se le puede considerar como elemento independiente del sistema.

Los efectos de la decisión del grupo pueden sentirse por elementos que aun siendo ajenos al sistema considerando, influyen directa o indirectamente, favorable o desfavorablemente hacia el (político, consumidor, etc.).

Para formular un problema se necesita la siguiente información:

¿Existe un problema?

¿De quien es el problema?.

¿Cual es el marco de referencia del sistema donde se encuentra el problema?

¿Quien o quienes toman las decisiones?

¿Cuales son sus objetivos?

¿Cuales son los componentes controlables del sistema y cuales no lo son?.

¿Cuales son las interrelaciones más importantes del sistema?.

¿Como se emplearan los resultados del proyecto? ¿Por quien? ¿Qué efectos tendrá?

¿Las soluciones tendrán efecto a corto o largo plazo?

¿Podrán los efectos de las soluciones modificarse o cambiarse fácilmente?

¿Cuantos elementos del sistema se afectaran por las soluciones del proyecto? ¿En qué grado?

FORMULAR UN PROBLEMA REQUIERE:

- Identificar las componentes controlables de un sistema.
- Identificar posibles rutas de acción dadas por las componentes, controlables.
- Definir el marco de referencia, dado por las componentes no controlables
- Definir los objetivos que se persiguen y clasificarlos por su orden de importancia.
- Identificar las relaciones importantes entre las diferentes componentes del sistema, este paso equivale a encontrar las restricciones que existen, a la vez que permite más adelante representar estas interrelaciones en forma matemática.

La identificación de la estructura del sistema (componentes, canales, interrelaciones, etc.), se hace a través de un proceso sistemático, que se conoce como diseño de sistemas.

El diseño de sistemas se lleva a cabo de la siguiente manera:

- Se ubica al sistema considerando dentro de sistemas más grandes.
- Se determinan las componentes del sistema.
- Se determinan los canales de comunicación entre las componentes del sistema y de este hacia los elementos de otros sistemas que van a tener influencia directa o indirecta.
- Se determinan de que manera se tiene acceso a la información requerida como se procesa esta y como se transmite entre las diferentes componentes del sistema.

GENERACIÓN DE VALORES DE UNA VARIABLE ALEATORIA

INTRODUCCIÓN.

Métodos más utilizados para generar números aleatorios y pseudoaleatorios con computadora.

Antes de continuar, es necesario establecer la siguiente terminología. El término variable aleatoria se emplea para nombrar una función de valor real, definida sobre un espacio muestral asociado con los resultados de un experimento conceptual, de naturaleza azarosa. El valor numérico resultante de un experimento, de cada una de las variables aleatorias, se llama número aleatorio. Se utilizan letras mayúsculas para denotar las variables aleatorias y minúsculas, para denotar valores de éstas variables aleatorias y minúsculas, para denotar valores de éstas variables, es decir, para los números aleatorios. Por ejemplo, $F(x)$; la función de distribución acumulada para una variable aleatoria X , indica la probabilidad de que X sea menor o igual al particular valor x de la función de probabilidad de la variable aleatoria X , cuando $X = x$.

TECNICAS PARA GENERAR NÚMEROS ALEATORIOS.

Se han venido usando cuatro métodos alternativos para generar las sucesiones de números aleatorios, estos son:

Métodos manuales

Lanzamiento de monedas

Lanzamiento de dados

Barajas

Dispositivos mecánicos

Dispositivos electrónicos

Ventajas: Son aleatorios

Desventajas: No reproducibles

TABLAS DE BIBLIOTECA.

Son números aleatorios que se han publicado; por ejemplo a Million Random Digits, de la Corporación Rand, de los cuales podemos encontrar listas de los en los libros de probabilidad y tablas de matemáticas. Estos números fueron generados por alguno de los métodos de computación analógica, los cuales mencionados a continuación.

Ventaja: Proviene de un fenómeno aleatorio y son reproducibles.

Desventaja: No se obtiene en tiempo real.

MÉTODOS DE COMPUTACIÓN ANALÓGICA

Los métodos de computación analógica dependen de ciertos procesos físicos aleatorios (por ejemplo, el comportamiento de una corriente eléctrica), por lo que se considera que conducen verdaderos números aleatorios.

Ventaja: Aleatorios.

Desventaja: No reproducible.

MÉTODOS DE COMPUTACIÓN DIGITAL

Se distinguen tres métodos para producir números aleatorio cuando se usa la computación digital (computadoras), los cuales son:

PROVISIÓN EXTERNA. Consiste en grabar en la memoria de la computadora, las tablas Randa, a fin de tratar estos números como datos de entrada para un determinado problema.

GENERACIÓN POR MEDIO DE PROCESOS FÍSICOS ALEATORIOS. Consiste en usar algún aditamento especial de la computadora, para registra los resultados de algún proceso aleatorio y además, reduzca estas resultados a sucesiones de dígitos.

GENERACIÓN INTERNA POR MEDIO DE UNA RELACIÓN DE RECURRENCIA. Consiste en generar números pseudoaleatorios por medio de ecuaciones de recurrencia, en las que necesariamente se tiene que dar un valor inicial o semilla, para generar los siguientes valores. Vamos ha centrar nuestra atención en este último método de computación digital, y los describiremos ampliamente.

Ventaja: Son reproducibles.

Desventaja: Son pseudoaleatorios.

CARACTERÍSTICAS DE LOS NÚMEROS PSEUDOALEATORIOS

Uniformemente distribuidos

Estadísticamente independientes

Reproducibles

Sin repetición dentro de una longitud determinada

MÉTODOS QUE UTILIZAN ECUACIONES DE RECURRENCIA PARA GENERAR NÚMEROS PSEUDOALEATORIOS.

Aquí describiremos los métodos de generación de números pseudoaleatorios, usando ecuaciones de recurrencia.

MÉTODOS DE GENERACIÓN DE NUM. PSEUDOALEATORIOS $U(0,1)$.

-Métodos congruenciales "69"

Reglas:

- C debe ser un entero impar, no divisible ni por 3 ni por 5
- a usualmente puede ser cualquier constante sin embargo para asegurar buenos resultados, seleccione a de tal forma que $(a) \bmod 8 = 5$ para una computadora binario a o $(a) \bmod 200 = 21$ para una computadora decimal.
- M debe ser el número entero más grande que la computadora acepte

De acuerdo con Hull y Debell, los mejores resultados par un generador congruencial mixto en una computadora binaria son:

- $a = 8 * c$

Lenguajes de propósito general como.

C, C++.

C es un lenguaje de programación diseñado por Dennis Ritchie, de los Laboratorios Bell, y se instaló en un PDP-11 en 1972; se diseñó para ser el lenguaje de los Sistemas Operativos UNIX1. A su vez, UNIX es un Sistema Operativo desarrollado por Ken Thompson, quién utilizó el lenguaje ensamblador y un lenguaje llamado B para producir las versiones originales de UNIX, en 1970. C se inventó para superar las limitaciones de B. C es un lenguaje maduro de propósitos generales que se desarrolló a partir de estas raíces; su definición aparece en 1978 en el apéndice "C Reference Manual" del libro The C Programming Language, de Brian W. Kernighan y Dennis M. Ritchie (Englewood Cliffs, Nueva Jersey, Prentice-Hall 1978), pero el estándar recomendable más reciente apareció en junio de 1983, en el documento de los Laboratorios Bell titulado The C Programming Language-Reference Manual, escrito por Dennis M. Ritchie

Un programa en C

Generalizando, un programa en C consta de tres secciones. La primera sección es donde van todos los "headers". Estos "headers" son comúnmente los "#define" y los "#include". Como segunda sección se tienen las "funciones". Al igual que Pascal, en C todas las funciones que se van a ocupar en el programa deben ir antes que la función principal (main()). Declarando las funciones a ocupar al principio del programa, se logra que la función principal esté antes que el

resto de las funciones. Ahora, solo se habla de funciones ya que en C no existen los procedimientos.

Y como última sección se tiene a la función principal, llamada main. Cuando se ejecuta el programa, lo primero que se ejecuta es esta función, y de ahí sigue el resto del programa. Los símbolos { y } indican ``begin" y ``end" respectivamente. Si en una función o en un ciclo while, por ejemplo, su contenido es de solamente una línea, no es necesario usar ``llaves" ({ }), en caso contrario es obligación usarlos.

Ejemplo de un programa en C

```
/*Programa ejemplo que despliega el contenido de "ROL" en pantalla*/
#include <stdio.h>
#define ROL "9274002-1"
despliega_rol() {
printf("Mi rol es : \\\%s\\n", ROL);
}
void main() {
despliega_rol();
}
/* Fin programa */
```

Delphi.

En 1994, Borland comenzó a trabajar en una herramienta RAD (Desarrollo Rápido de Aplicaciones, por sus siglas en inglés) a la que asignó el nombre clave de Delphi. Cuando se decidió que la arquitectura del modelo de componentes era la mejor forma de implementar el desarrollo rápido de aplicaciones, entonces fue necesario establecer el lenguaje de programación que sería el corazón del sistema.

En ese momento Borland era el único fabricante de compiladores en masa que comercializaba un compilador de Pascal. Borland era conocida como la compañía que producía las mejores herramientas Pascal. Si usted fue un programador de Pascal, probablemente utilizó el Turbo Pascal de Borland en alguna de sus modalidades. En cierto modo, Borland era el "dueño" de Pascal. Aunque Borland no poseía el lenguaje Pascal en un sentido legal, sin duda pensó que debido a su posición en el mundo de Pascal, podía tomarse libertades considerables en la implementación de nuevas características y mejoras al lenguaje. Además, no había un comité de estándares de Pascal, ni siquiera un estándar escrito que definiera al lenguaje Pascal. Así que Borland creó Delphi utilizando Pascal como el lenguaje base (el nombre clave interno de Borland se quedó y se convirtió en el nombre oficial del producto).

Antes de que existiera Delphi, Borland ya había modificado el lenguaje Pascal en formas positivas. Por ejemplo, Borland ya había extendido a Pascal creando un nuevo lenguaje denominado Object Pascal. Podría decirse que Object Pascal es a Pascal lo que C++ es a C. Object Pascal incorporó clases a Pascal, lanzando así a Pascal al mundo de los lenguajes de OOP (Programación Orientada a Objetos, por sus siglas en inglés). Mientras se desarrollaba Delphi, se agregaron palabras clave y un nuevo comportamiento del lenguaje para abordar el modelo de componentes. Se incorporaron palabras clave como *published* y *property*, así como otras. Esto le permitió a

Borland implementar por completo el poder del modelo de componentes. Al modificar el lenguaje Pascal para adaptarlo al modelo de componentes, Borland pudo implementar el desarrollo rápido de aplicaciones en la forma correcta. En esencia, el lenguaje Object Pascal se modificó de acuerdo a las necesidades cuando surgieron aspectos de diseño durante el desarrollo del entonces desconocido producto de nombre Delphi. El resultado es un lenguaje que funciona de manera fluida con el modelo de componentes.

Aunque la modificación del lenguaje Pascal podría considerarse como un paso capital de Borland, no fue sin que hubiera precedentes. Antes, Microsoft había tomado y modificado el lenguaje BASIC para producir un nuevo lenguaje denominado Visual Basic. Este nuevo lenguaje era casi irreconocible al compararlo con el lenguaje BASIC original que le sirvió de base.

Borland se arriesgó al modificar a Pascal. Después de todo, tenía una base leal de clientes que podrían no tomar a bien las mejoras al lenguaje que habían llegado a conocer y apreciar. Sin embargo, Borland tenía una sólida posición en el mercado de Pascal y continuó con sus planes. El resultado fue, por supuesto, un gran éxito.

No se hay que equivocarse, Object Pascal es un lenguaje de programación poderoso, y no se hace esta declaración a la ligera. Pronto descubrirá que el lenguaje Object Pascal es muy capaz. De hecho, en manos del programador promedio, casi no hay diferencia entre los lenguajes C/C++ y Object Pascal en términos de poder. Object Pascal es único en que es al mismo tiempo poderoso y relativamente fácil de aprender. No se quiere de ninguna manera dejar la impresión de que Pascal no es un lenguaje de programación con todas las características. Con frecuencia, Pascal ha sido tachado como un lenguaje de programación menos que serio. Eso nunca ha sido cierto, y menos cierto con el Object Pascal actual.

Visual

La **Simulación Visual Interactiva**, que puede definirse como aquella que posibilita la creación gráfica de modelos de simulación, permite mostrar por pantalla dinámicamente el sistema simulado, así como la interacción entre el usuario y el programa en ejecución. La interacción implica que o bien se detiene la simulación y solicita información al usuario, o bien que éste puede parar la simulación a su voluntad e interaccionar con el mencionado programa ; esto último se puede realizar "off-line" o "on-line", es decir sin interrumpir la simulación, e introduciendo las variaciones oportunas tanto en los modelos, como en los valores de las variables en el siguiente ciclo de scan del proceso de ejecución del programa en la computadora que para esto debe tener una estructura multitarea que permita este tipo de operaciones. Algunos productos del mercado son: SIMFACTORY DE CACI Inc. , PROMODEL de ProModel Corporation , ARENA de Rockwell Software Inc., WITNESS de ATT & Istel , o FACTOR/AIM de Pritsker Corporation , FIX DEMACS de Intellution (Fisher-Rosemount). Todos ellos son productos orientados primordialmente a la utilización de la simulación para la resolución de problemas en el ámbito de la producción.

Utilizables desde entorno Windows, y ejecutables sobre computadoras personales o sobre plataformas mas potentes como Estaciones de trabajo (Workstations). Estos permiten construir modelos complejos de manera incremental, a partir de la selección de componentes del sistema de entre un repertorio limitado a la extensión de las librerías que contienen unas entidades predefinidas, si bien las ultimas tendencias añaden a estos paquetes editores para crear nuevas plantillas con características a gusto del consumidor, introduciendo además utilidades de todo tipo

incluidas las gestiones de configuración y control de las comunicaciones con un sistema de control real al que se puede conectar el equipo.

Ventajas:

- Sirven para comunicar la esencia del modelo de simulación a los directivos.
- Puede ayudar a corregir errores del programa de simulación, o a mostrar que el modelo no es válido.
- Puede ayudar a entender el comportamiento dinámico del sistema.

Inconvenientes:

- No puede sustituir a un cuidadoso análisis estadístico de los resultados.
- Sólo una parte de la lógica del modelo de simulación puede verse en la animación, y no se puede concluir a partir de ese corto periodo de tiempo que el modelo está bien definido.
- Aumenta el tiempo para desarrollar el programa de simulación.
- Muy lenta la animación en directo.

Otro enfoque se puede derivar de los lenguajes de simulación y de los simuladores es el de los Sistemas Híbridos que combinan la flexibilidad de un lenguaje de simulación con la facilidad de uso de un simulador como lo son el ARENA y el QUEST.

Los simuladores y lenguajes de simulación pueden adoptar uno de los diferentes métodos o estrategias. Existen tres estrategias que son generalmente reconocidas:

- **Enfoque de modelado basado en eventos.** La orientación basada en Eventos (ES) es gobernada por un calendario y ejecución de subrutinas (eventos) que como consecuencia programa la ejecución de otras subrutinas. Los eventos son los instantes de tiempo en los cuales un cambio en el sistema ocurre y coincide con el inicio o terminación de las actividades. Bajo este enfoque segmentos del programa son empleados para definir cada evento en el modelo. Después de inicializado el modelo, las rutinas de ejecución revisan los tiempos de ocurrencia de los eventos y avanzan el reloj de la simulación hacia el tiempo en el cual ocurrirá el próximo evento. Debe existir una subrutina para cada tipo de evento.
- **Enfoque de modelado basado en actividades.** La orientación basada en Seguimiento de Actividades (SA) bajo este enfoque un segmento del programa es empleado para definir cada actividad en la cual las entidades se ven involucradas y las condiciones bajo las cuales la actividad puede realizarse. Dicho segmento incluye una serie de pruebas para determinar si la actividad ha sido iniciada en un punto del tiempo y define las acciones que se deben ejecutar si la actividad ha sido iniciada.
- **Enfoque de modelado basado en procesos.** La orientación basada en Interacción de Procesos (IP) es desarrollada desde el punto de vista de las entidades (transacciones) que fluyen en el sistema. Bajo este enfoque las entidades se clasifican en transacciones o clientes, servidores o recursos (entidades permanentes y entidades temporales). En este enfoque, existen segmentos del programa que son empleados para describir los procesos en los cuales se ven involucradas las entidades.

Sistemas productivos.

La simulación en los sistemas productivos permite elaborar diversas situaciones, a veces difíciles de imaginar en la realidad, tales como una baja en la producción, pérdidas en la distribución de productos, etc., estas técnicas son útiles para desarrollar planes de contingencia y estar preparados para la toma de decisiones en forma rápida y segura.

Por ejemplo, el grupo SIMON de investigaciones en modelos y simulación, aplicando el carácter interdisciplinario de la Dinámica de Sistemas, asociado a labores desarrolladas por CORPOICA en el análisis de procesos pecuarios, desarrolló el proyecto titulado, *Propuesta de un Modelo de Simulación de Sistemas de Producción de Ganadería Bovina para la Investigación Integral. Un Enfoque Sistémico: SIPROB 1.0*, elaborado con el lenguaje de la Dinámica de Sistemas y con el propósito de representar los sistemas de producción bovina, permitiendo su comprensión y la prueba de diferentes tecnologías para facilitar a los ganaderos la toma de decisiones y el aprendizaje sobre sus sistemas de Producción.

En la construcción del modelo se elaboraron tres prototipos. A continuación se presenta una descripción general de uno de ellos:

El prototipo 1 modela los sistemas de producción Bovina con relación a variables demográficas y productivas, estas hacen referencia a los elementos que componen el sistema: Terneros, Levantes y Vacas -en todas sus etapas: Vacías, Preñadas, En días de Descanso y Lactantes-, con factores determinantes de la producción como son la Tasa de Natalidad y la Probabilidad de Muerte. Se presenta el ciclo animal completo desde el nacimiento o introducción al sistema de cada animal ya sea por una compra o por el nacimiento del mismo, hasta su muerte o venta cuando este cumpla su ciclo. En este prototipo no se presentan los limitantes biofísicos ni los económicos.

Calidad.

En los procesos de control de calidad, los modelos de simulación representan una herramienta de mucha utilidad para la obtención de información acerca de los procesos y de la forma de corregir los defectos que se derivan en el manejo de los mismos. Por ejemplo, en el trabajo titulado:

“Medida y simulación Monte Carlo de factores de corrección por calidad de haz”

La calibración de un haz de fotones de megavoltaje de un usuario consiste en la determinación de la dosis absorbida en agua depositada por el haz en condiciones de referencia. Esto se realiza mediante la medida de carga o corriente recogida en una cámara de ionización que ha sido calibrada en un laboratorio de calibración mediante un haz de cobalto 60. Para tener en cuenta la diferencia entre los espectros del haz del usuario y del haz de cobalto del laboratorio de calibración se utiliza un factor de corrección por calidad de haz. La red de laboratorios de calibración no cuenta con haces de calidades iguales a las de los usuarios para determinar directamente estos factores. Éstos se encuentran tabulados en protocolos de dosimetría para la mayor parte de cámaras de ionización comerciales [1]. Para aquellas cámaras no tabuladas, los protocolos describen algoritmos sencillos para estimar estos factores. La incertidumbre asociada a ellos es del orden del 1%. En el presente trabajo, se muestran los resultados de la simulación Monte Carlo de factores de corrección por calidad de haz (para distintas calidades y cámaras de ionización) y su comparación con medidas experimentales relativas a una cámara de ionización calibrada a varias calidades en el laboratorio de calibración alemán (Physikalisch-Technische Bundesanstalt, PTB).

Tanto los resultados experimentales como los de las simulaciones muestran una incertidumbre menor que la de los factores tabulados en los protocolos de dosimetría, siendo compatibles con éstos dentro de incertidumbres (una desviación estándar).

Material y métodos

En este trabajo se determinaron mediante dos métodos factores de corrección por calidad de haz para 9 modelos de cámara de ionización de los principales fabricantes (PTW, Exradin y Wellhöfer).

Simulación Monte Carlo: El factor de corrección por calidad de haz para una cámara de ionización y un haz de calidad Q dados.

$$k_{Q,Q_0} = \frac{\left(\frac{D_w}{\overline{D}_{air}} \right)_Q}{\left(\frac{D_w}{\overline{D}_{air}} \right)_{Q_0}}$$

Donde D_w es la dosis absorbida en agua, determinada a la profundidad de referencia en un tanque de agua y condiciones de irradiación de referencia establecidas en el protocolo de dosimetría TRS-398; Y \overline{D}_{air} es la dosis absorbida en el aire encerrado en la cavidad de una cámara de ionización cuyo punto de referencia está situado en el mismo punto en el cual se determine D_w y en las mismas condiciones de irradiación de referencia establecidas en el protocolo. La ecuación (1) se deduce de la definición alternativa del factor de corrección por calidad de haz que aparece en Andreo 1992 [2] y de la relación entre D_w y \overline{D}_{air} .

Las dosis que intervienen en la ecuación (1) se calcularon mediante simulación Monte Carlo (sistema EGSnrc [3]) propagando en un tanque de agua los espacios de fase correspondientes a la simulación de haces clínicos (haz de cobalto 60 como referencia y haces de megavoltaje de 4 a 25 MV, mediante el código BEAMnrc [4]). Para el cálculo de la dosis en agua se almacenó la dosis en un voxel centrado a la profundidad de referencia en un tanque de agua (código cavrznrc [5]). Para el cálculo de la dosis en la cavidad se empleó una geometría realista de la cámara de ionización proporcionada por el fabricante

Inventarios.

Un ejemplo sería un sistema de inventarios de una fábrica, o bien el sistema de líneas de espera de una fabrica, etc. Estos modelos producen una salida que es en si misma de carácter aleatorio y ésta debe ser tratada únicamente para estimar las características reales del modelo, esta es una de las principales desventajas de este tipo de simulación, Otro ejemplo: un fabricante de comida para perros, requiere el auxilio de una compañía consultora con el objeto de construir un modelo de simulación para su línea de fabricación, la cual produce medio millón de latas al día a una velocidad casi constante. Debido a que cada una de las latas se representó como una entidad separada en el modelo, éste resulto ser demasiado detallado y por ende caro para correrlo, haciéndolo poco útil. Unos meses más tarde, se hizo una reformulación del modelo, tratando al proceso como un flujo continuo. Este nuevo modelo produjo resultados precisos y se ejecuto en una fracción del tiempo necesario por el modelo original.

Económicos.

En este campo la simulación representa un mecanismo de evaluación para la toma de decisiones que afecta a grandes sectores de la población, un ejemplo es el estudio titulado: "Simulación de políticas económicas: los modelos de equilibrio general aplicado". Los modelos de equilibrio general aplicado o computacional se utilizan para la simulación de políticas económicas de diversa índole, como políticas fiscales, comerciales, medioambientales, y otras. Estos modelos han comenzado a plantearse recientemente como problemas de complementariedad mixta, lo que ha permitido solucionar algunos de los problemas planteados en su formulación tradicional como problema de optimización del comportamiento de los agentes. En este trabajo realizamos una presentación y puesta al día de la metodología, junto con una descripción de las bases de datos utilizadas (matrices de contabilidad social), así como una revisión de los modelos de equilibrio general aplicado realizados para la economía.