Contents

“‘markdown

Clase 4: Generación de Números Aleatorios y Distribuciones de Probabilidad en SED

Objetivos de la clase:

- Comprender la importancia de la generación de números aleatorios en la simulación de eventos discretos.
- Aprender a utilizar generadores de números pseudoaleatorios (GNPAs) y evaluar su calidad.
- Conocer y aplicar distribuciones de probabilidad comunes en la modelación de sistemas con SED (exponencial, uniforme, normal, Poisson).
- Implementar métodos para generar variables aleatorias a partir de distribuciones de probabilidad.

Contenido Teórico Detallado:

1. La Importancia de la Aleatoriedad en SED:

En la Simulación de Eventos Discretos, la aleatoriedad juega un papel crucial para modelar la incertidumbre y la variabilidad inherente en muchos sistemas del mundo real. La aleatoriedad se utiliza para:

- **Tiempos de llegada:** Modelar el tiempo entre la llegada de entidades al sistema (p.ej., clientes a una tienda, paquetes a un servidor).
- **Tiempos de servicio:** Representar la duración del servicio que un recurso presta a una entidad (p.ej., el tiempo que un cajero atiende a un cliente, el tiempo que un servidor procesa un paquete).
- **Toma de decisiones:** Simular decisiones aleatorias dentro del sistema (p.ej., la probabilidad de que un cliente abandone la cola, la ruta que toma un paquete en una red).

2. Generadores de Números Pseudoaleatorios (GNPAs):

- **Definición:** Los GNPAs son algoritmos deterministas que producen secuencias de números que aparentan ser aleatorios. Es fundamental comprender que no son verdaderamente aleatorios, sino pseudoaleatorios.
- **Propiedades Deseables:**
  - **Uniformidad:** Los números generados deben estar distribuidos uniformemente en el intervalo (0, 1).
  - **Independencia:** Los números generados deben ser estadísticamente independientes entre sí.
  - **Periodo Largo:** El GNPA debe tener un periodo largo antes de que la secuencia de números se repita.
  - **Eficiencia:** El algoritmo debe ser computacionalmente eficiente.
- **Ejemplo Común: Generador Congruencial Lineal (GCL):**
  - Fórmula: $X_{i+1} = (aX_i + c) \bmod m$
    - X_i: El número aleatorio actual.
    - X_{i+1}: El siguiente número aleatorio en la secuencia.
    - a: Multiplicador.
    - c: Incremento.
    - m: Módulo.
  - Selección de parámetros: La elección adecuada de a, c y m es crucial para la calidad del GNPA. Valores mal elegidos pueden llevar a periodos cortos y patrones no aleatorios. Investigar valores sugeridos en la literatura (p.ej., Knuth).

- **Pruebas de Aleatoriedad:** Existen diversas pruebas estadísticas (p.ej., prueba de Kolmogorov-Smirnov, prueba de chi-cuadrado) para evaluar la calidad de los GNPAs y verificar si cumplen con las propiedades de uniformidad e independencia.

3. Distribuciones de Probabilidad Comunes en SED:

- **Distribución Exponencial:**
  - Uso: Modelar el tiempo entre eventos (p.ej., tiempo entre llegadas en un sistema de colas, tiempo de falla de un componente).
  - Parámetro: (tasa de ocurrencia).
  - Función de Densidad de Probabilidad (PDF): f(x) = e^{-x}, para x 0
  - Generación: X = - (1/) * ln(U), donde U es un número aleatorio uniforme en (0, 1).
- **Distribución Uniforme:**
  - Uso: Modelar situaciones donde todos los valores dentro de un rango tienen la misma probabilidad (p.ej., un tiempo de servicio que varía entre un mínimo y un máximo).
  - Parámetros: a (valor mínimo) y b (valor máximo).
  - Función de Densidad de Probabilidad (PDF): f(x) = 1/(b-a), para a x b
  - Generación: X = a + (b - a) * U, donde U es un número aleatorio uniforme en (0, 1).
- **Distribución Normal (Gaussiana):**
  - Uso: Modelar fenómenos que se agrupan alrededor de un valor promedio (p.ej., altura de personas, errores de medición).
  - Parámetros: (media) y (desviación estándar).
  - Generación: Se utilizan métodos como la Transformada de Box-Muller o el Teorema del Límite Central (sumar varios números aleatorios uniformes para aproximar una distribución normal). La implementación directa es compleja, por lo que generalmente se utilizan bibliotecas especializadas.
- **Distribución de Poisson:**
  - Uso: Modelar el número de eventos que ocurren en un intervalo de tiempo o espacio fijo (p.ej., número de clientes que llegan a una tienda en una hora, número de errores en una página web).
  - Parámetro: (tasa promedio de eventos).
  - Función de Masa de Probabilidad (PMF): P(X = k) = (e^{-} * ^{k}) / k!, donde k es el número de eventos.
  - Generación: Se puede generar utilizando la relación entre la distribución de Poisson y la exponencial. Generar tiempos entre eventos exponenciales hasta que la suma exceda el intervalo de tiempo deseado. El número de eventos será uno menos que el número de tiempos exponenciales generados.

4. Métodos para Generar Variables Aleatorias:

- **Transformada Inversa:** Si se conoce la función de distribución acumulativa (CDF) F(x) de una distribución, se puede generar una variable aleatoria X resolviendo la ecuación U = F(X) para X, donde U es un número aleatorio uniforme en (0, 1). Este método es aplicable a la distribución exponencial y uniforme.
- **Aceptación-Rechazo:** Se utiliza cuando la transformada inversa es difícil de aplicar. Se propone una distribución "envolvente" más simple y se generan valores de ambas distribuciones. Se aceptan los valores de la distribución deseada si cumplen una condición basada en la relación entre las PDFs.
- **Uso de Bibliotecas:** La mayoría de los lenguajes de programación y simuladores ofrecen bibliotecas con funciones para generar variables aleatorias a partir de distribuciones comunes. Es importante entender cómo funcionan estas funciones y cómo configurarlas correctamente.

Ejemplos y Casos de Estudio:

1. **Simulación de un Sistema de Colas M/M/1:**
   - Las llegadas siguen una distribución de Poisson con tasa .
   - Los tiempos de servicio siguen una distribución exponencial con tasa .
   - Implementar un modelo SED para simular este sistema. Generar tiempos de llegada y tiempos de servicio utilizando las distribuciones correspondientes. Calcular el tiempo de espera promedio de los clientes y la utilización del servidor.

2. **Simulación de una Red de Computadoras:**
   - El tamaño de los paquetes sigue una distribución normal.
   - El tiempo de procesamiento en un router sigue una distribución uniforme.
   - Modelar el tráfico en la red y evaluar el retardo promedio de los paquetes.

Problemas Prácticos y Ejercicios con Soluciones:

1. **Implementación de un GNPA:** Implementar un Generador Congruencial Lineal (GCL) en un lenguaje de programación de su elección. Experimente con diferentes valores de a, c y m. Verificar la uniformidad de los números generados usando un histograma.
   - **Solución (Ejemplo en Python):**

“‘python def gcl(semilla, a, c, m, cantidad): numeros = [] x = semilla for _ in range(cantidad): x = (a * x + c) % m numeros.append(x / m) # Normalizar al rango (0, 1) return numeros

Ejemplo de uso

semilla = 12345 a = 1664525 c = 1013904223 m = 2**32 cantidad = 10000 numeros_aleatorios = gcl(semilla, a, c, m, cantidad)

import matplotlib.pyplot as plt plt.hist(numeros_aleatorios, bins=50) plt.title(”Histograma de números aleatorios generados por GCL”) plt.xlabel(”Valor”) plt.ylabel(”Frecuencia”) plt.show() “‘

1. **Generación de Variables Aleatorias Exponenciales:** Escribir una función que genere variables aleatorias a partir de una distribución exponencial utilizando el método de la transformada inversa.
   - **Solución (Ejemplo en Python):**

“‘python import math import random

def generar_exponencial(tasa): u = random.random() # Generar un número aleatorio uniforme en (0, 1) x = - (1 / tasa) * math.log(u) return x

Ejemplo de uso

tasa = 0.5 cantidad = 1000 variables_exponenciales = [generar_exponencial(tasa) for _ in range(cantidad)]

plt.hist(variables_exponenciales, bins=50) plt.title(”Histograma de variables aleatorias exponenciales”) plt.xlabel(”Valor”) plt.ylabel(”Frecuencia”) plt.show() “‘

1. **Simulación de un Cajero Automático:** Simular el funcionamiento de un cajero automático. Los clientes llegan siguiendo una distribución de Poisson con una tasa de 10 clientes por hora. El tiempo de servicio del cajero sigue una distribución exponencial con una media de 3 minutos. Simular el sistema durante 8 horas y calcular el tiempo de espera promedio de los clientes.
   - (Solución: Este problema requiere la implementación completa de un modelo SED, definiendo eventos de llegada, inicio de servicio y fin de servicio, y manteniendo estadísticas sobre el tiempo de espera. Se anima a los estudiantes a utilizar un lenguaje de simulación o una biblioteca SED para resolver este problema.)

Materiales Complementarios Recomendados:

- **Libros:**
  - ”Simulation Modeling and Analysis” by Averill M. Law
  - ”Discrete-Event System Simulation” by Jerry Banks, John S. Carson II, Barry L. Nelson, David M. Nicol
- **Artículos:** Buscar artículos sobre pruebas de aleatoriedad para GNPAs.
- **Recursos en línea:** Documentación de bibliotecas de simulación (p.ej., SimPy, AnyLogic).

“‘