Curso:
Métodos de Monte Carlo
Unidad 4, Sesión 8: Números aleatorios

Departamento de Investigación Operativa
Instituto de Computación, Facultad de Ingeniería
Universidad de la República, Montevideo, Uruguay

dictado semestre 1 - 2016

Contenido:

1. Números aleatorios.
2. Fuentes de números aleatorios.
3. Dispositivos de hardware.
4. Generadores software/hardware.
5. Fuentes en Internet.
6. Ejercicio.

Números aleatorios

Una de las bases esenciales para la aplicación de métodos de Monte Carlo radica en el empleo de números obtenidos mediante el sorteo de variables aleatorias independientes de distribución uniforme en $(0, 1)$. Una secuencia de estos números recibe el nombre de secuencia de números aleatorios. De manera alternativa, también recibe el nombre de secuencia de números aleatorios a una secuencia de valores obtenidos de sortear variables aleatorias independientes de distribución discreta entre los números naturales 0 y M.

Resulta entonces necesario estudiar cómo es posible disponer en una computadora de secuencias de esta naturaleza (o de comportamiento suficientemente cercano). Este problema surge en muchos otros contextos además de la aplicación de métodos Monte Carlo, quizás los de mayor impacto económico en la actualidad son las aplicaciones criptográficas, y los juegos (de azar o de otro tipo), pero también en general las simulaciones a eventos discretos, el muestreo de casos representativos, la generación de casos de prueba para testeo de software o para análisis de

tiempos de ejecución de algoritmos, la generación de escenas de realidad virtual en computación gráfica, etc. Dista mucho de ser un tema resuelto de manera totalmente satisfactoria; por el contrario, es objeto de investigación activa y en los últimos años han aparecido soluciones que han mejorado notablemente las posibilidades existentes.

Históricamente, tirar un dado, sortear bolas en una urna (por ejemplo para el juego de lotería), o hacer girar una ruleta han sido algunas de las primeras formas de obtener números aleatorios, suponiendo que estos dispositivos han sido construidos de manera tal de obtener una distribución presumiblemente uniforme.

Ya a comienzos del siglo XX, estos mecanismos eran insuficientes, y otros aparatos mecánicos o electromecánicos fueron diseñados para esta tarea; muy pronto luego de la construcción de las primeras computadoras programables, comenzaron los primeros intentos para emplearlas en la obtención de números aleatorios.

Fuentes de números aleatorios

En la actualidad, las principales formas para obtener secuencias de números aleatorios son:

1. A través de algoritmos determinísticos (software).

2. A través de dispositivos de hardware diseñados específicamente.

3. Mixto, software que emplea información proveniente del hardware estándar de una computadora.

Una cuarta forma es el empleo de números que fueron generados externamente (con alguna de las tres alternativas previas), y están disponibles a través de un dispositivo de almacenamiento o de comunicación. Históricamente, esta última alternativa estaba representada por tablas de números aleatorios publicadas en papel; posteriormente (década del 50) también existieron tablas disponibles no sólo como libros,

sino también en formato de tarjetas perforadas (ver http://en.wikipedia.org/wiki/A_Million_Random_Digits_with_100%2C000_Normal_Deviates - accedido 2016-04-11). Durante muchos años esta alternativa quedó descartada, pero en la década del 90 aparecieron tablas en CD, y en la actualidad existen además varios sitios Web que ofrecen en forma gratuita números aleatorios obtenidos a través de dispositivos de hardware específicos.

El método que resulta más rápido y satisfactorio para las aplicaciones de tipo Monte Carlo es en general el empleo de algoritmos determinísticos, que técnicamente son llamados generadores de números seudo-aleatorios (ya que conceptualmente sería una contradicción que un método determinístico generara números aleatorios). Si bien nos concentraremos en estos métodos, igualmente veremos brevemente alguna información sobre la generación empleando dispositivos de hardware, y sobre el empleo de números generados externamente.

Dispositivos de hardware

Los dispositivos de hardware utilizan en general la medición de alguna fuente de ruido ambiente o generada internamente, que es luego transmitida a una computadora a través de una interfaz estándar (eventualmente tras haber sufrido algún procesamiento por software para mejorar la calidad de los números generados).

Entre las fuentes fundamentales de aleatoriedad física se encuentran las derivadas de la mecánica cuántica a nivel atómico, y las derivadas del ruido termal.

Dado que de acuerdo a las teorías físicas actualmente vigentes, se admite que no es posible prever por ningún método el resultado de un evento a nivel cuántico, estos serían la fuente de aleatoriedad más apropiada. Es el caso por ejemplo de una fuente de radiación ligada a la fisión atómica de un material radioactivo, en la que los eventos son fácilmente detectables con un medidor Geiger. Otro caso es el de fotones atravesando un espejo semi-transparente, que puede transmitir o reflejar los mismos, lo que puede interpretarse como los bits 0 y 1.

Por otro lado, puede ser más sencillo el emplear eventos vinculados a fenómenos térmicos, como por ejemplo ruido térmico de una resistencia, amplificado para crear una fuente de voltaje aleatorio; o estática (ruido de fondo) recibida con una antena de radio.

Otras fuentes empleadas incluyen ruido a nivel de audio o video, aunque no resulta fácil caracterizar o garantizar las propiedades de aleatoriedad de los mismos.

En todos los casos, es posible que exista algún tipo de sesgo (si por ejemplo consideramos los números generados como una secuencia de 1 y 0, estos dos dígitos deberían aparecer de manera equiprobable asintóticamente, pero es muy posible que uno de los dos predomine). Es necesario entonces aplicar algún método para eliminar el sesgo (pero sin que esto implique perder la independencia entre los valores sucesivos).

Una de las formas más sencillas para hacer esto fue propuesta por John von Neumann, y consiste en considerar los bits generados de a pares, tomando una de las siguientes tres acciones: si dos bits sucesivos son iguales, se descartan; si aparece una secuencia 1,0 , se interpreta como el

bit 1; si aparece una secuencia 0,1 se interpreta como 0. Esto elimina sesgo simple (en el que un dígito aparece de manera más frecuente que el otro), al costo de descartar una cantidad muy importante (siempre mucho mayor del 50%) de los bits generados. Existen otros métodos alternativos más sofisticados.

Lectura adicional obligatoria: Discusión sobre varios generadores de números aleatorios por hardware, y problemas encontrados con algunos de ellos, por Robert Davies, reporte publicado en la página `http://www.robertnz.net/true_rng.html` (accedido 2016-04-11).

Referencias adicionales a algunos proveedores comerciales de generadores de números aleatorios por hardware (no es obligatorio visitar todos los sitios, aunque puede resultar interesante ver alguno para conocer otros métodos de generación empleados - los vínculos pueden no existir actualmente, por cambios en las empresas):

- Generador fabricado por Mario Stipcevic, Quantum Random Number Generator: `http://qrbg.irb.hr/`

- Generador fabricado por Rolf Freitag: `http://www.true-random.com/`
- Empresa Protego: `http://www.protego.se`
- Empresa Comscire: `https://comscire.com/about-comscire/`
- Empresa RNGResearch: `http://RNGResearch.com/`
- Empresa IdQuantique: `http://www.idquantique.com/products/quantis.htm`

Una lista de proveedores está también disponible aquí: `http://mindprod.com/jgloss/truerandom.html`.

Una mención aparte merece LavaRnd `http://www.lavarnd.org/index.html`, que ha desarrollado un generador basado en hardware sencillo de obtener (webcams), y prove las instrucciones para montarlo así como software en licencia libre (LGPL) en

forma gratuita: LavaRnd hardware number generador, `http://sourceforge.net/projects/lavarnd/` (último acceso: 2016-04-11).

Números aleatorios disponibles en Internet

Existen algunas organizaciones que de manera experimental o como servicio a la comunidad han puesto disponibles a través de la Web sitios en los cuales es posible obtener números generados por dispositivos de hardware muy particulares, no disponibles en general comercialmente.

- HotBits, `http://www.fourmilab.ch/hotbits/` (último acceso: 2016-04-11), en el sitio Fourmilab creado y mantenido por John Walter. Este sitio permite obtener a través de un formulario de pedido un conjunto de bytes obtenidos mediante un detector de radiación (contador Geiger, ver los detalles en `http://www.fourmilab.ch/hotbits/hardware.html`).

  También se ofrece un paquete Java, llamado randomX, que encapsula algunos generadores estándar, así como una clase para acceder directamente al servidor de HotBits desde un programa.

- Sitio random.org, True Random Number Service

http://www.random.org/ (último acceso: 2016-04-11), creado y mantenido por Mads Haahr, docente en el Distributed Systems Group, Department of Computer Science, University of Dublin, Trinity College (Irlanda), que emplea estática atmosférica (radio) para generar números aleatorios.

- Sitio EntropyPool and Entropy Filter Home Page, http://random.hd.org/index.html (último acceso: 2016-04-11).

También es possible encontrar en algunos casos series específicas de números generadas con antelación y publicadas de manera estática, como por ejemplo la serie de dígitos generada por la RAND Corporation en los años 50, http://www.rand.org/content/dam/rand/pubs/monograph_reports/MR1418/MR1418.digits.txt.zip (último acceso: 2016-04-11), o números de ejemplo generados por el dispositivo de hardware xRNG, de RNGResearch: http://rngresearch.com/download/ (último acceso: 2016-04-11).

Generadores de software basados en el hardware estándar

Hace ya algún tiempo que se realizó la observación que el propio hardware de una computadora era fuente de aleatoriedad (o entropía), a través del estado por ejemplo de las interrupciones de teclado, movimientos del mouse, temporizaciones de la interfaz IDE, etc. Esto motivó la creación en el sistema operativo Linux del dispositivo `/dev/random/`, en el cual es posible acceder a números aleatorios generados con esta información. Sobre esta idea, e incorporando distintas fuentes de aleatoriedad, han surgido otros generadores como los siguientes:

- EGD: The Entropy Gathering Daemon, `http://egd.sourceforge.net/` (último acceso: 2016-04-11)
- PRNGD - Pseudo Random Number Generator Daemon, `http://prngd.sourceforge.net/` (último acceso: 2016-04-11)
- EntropyPool and Entropy Filter Home Page, `http://random.hd.org/` (último acceso: 2016-04-11)

La mayoría de estos no resulta adecuado para simulaciones Monte Carlo debido a que generan números de manera demasiado lenta para acompasarse con una simulación. Sin embargo, una de las propuestas más recientes, HAVEGE (`http://www.irisa.fr/caps/projects/hipsor/`, último acceso: 2016-04-11), emplea información interna muy detallada, entre otra sobre caches internos al procesador, predictores de saltos, paralelismo intrínseco, etc., lo que le permite generar números a tasas muy superiores a las de otras propuestas.

Preguntas para auto-estudio

- ¿Cuáles son las principales fuentes de números aleatorios?
- ¿Qué tipos de problemas se han observado en la práctica con generadores de hardware?
- ¿Qué fuentes de números aleatorios hay disponibles en Internet? De acuerdo a las páginas de las mismas, parecen de uso sencillo? Qué limitaciones tienen? (por ejemplo, de velocidad de respuesta o cantidad de números disponibles).

Ejercicio

Entrega 5
Ejercicio 8.1 : (grupal)

a) Elegir al menos dos fuentes de números aleatorios disponibles en Internet (sitio o tabla con valores). Explicar cómo funcionan, como se accede a los números, y qué características tienen.

b) En base a este análisis, elegir una de las fuentes, fundamentar la selección, y modificar el ejercicio 3.1, parte a (visto en la sesión 3) para que emplee dichos números aleatorios (en lugar de los generados por bibliotecas como hasta el momento). Comparar si la salida obtenida es consistente o no con la obtenida en los experimentos de la parte a del ejercicio 3.1.

Fecha entrega: Ver cronograma y avance del curso