

Capítulo 2 .

Fundamentos Básicos de Monte Carlo N-Particle.

2.1 Historia.

El método de Monte Carlo debe su nombre a la ciudad de Montecarlo en Mónaco donde se juega la ruleta, el juego de azar que genera números aleatorios. Este método surge formalmente en el año 1944, sin embargo, ya existían prototipos y procesos anteriores que se basaban en los mismos principios.

La utilización del método de Monte Carlo para fines de investigación comenzó con el desarrollo de la bomba atómica en la Segunda Guerra Mundial. Durante el proyecto Manhattan, los científicos Von Neumann y Ulam perfeccionaron la técnica y la aplicaron a problemas de cálculo de difusión de neutrones en un material. En 1948, Fermi, Metropolis y Ulam calcularon los eigenvalores de la ecuación de Schroedinger recurriendo a Monte Carlo. El uso de las computadoras proporciona una herramienta básica para realizar cálculos y estimaciones más complejas. El código MCNP, Monte Carlo N-Particle, desarrollado en el laboratorio de Los Alamos es el programa de computadora más completo basado en este método. Hoy en día el software MCNP se utiliza en desarrollo de reactores nucleares, Cromodinámica Cuántica, Radioterapia, comportamiento de radiación en la atmósfera terrestre, flujos de tráfico, evolución estelar, cálculos y predicciones económicas, búsqueda de petróleo, entre otros. El principal programador de MCNP fue Dr. Thomas N.K. Godfrey durante los años 1975-1989; sin embargo, hasta la fecha el código incorpora el trabajo de al menos 400 personas-año [16].

2.2 Método de Monte Carlo.

Las soluciones numéricas de un sistema físico se basan en un modelo matemático a partir del cuál se obtienen y resuelven las ecuaciones integro-diferenciales que describen un estado de dicho sistema. No obstante, existen problemas muy complejos, como las interacciones nucleares, que no pueden ser resueltos empleando modelos determinísticos. Con Monte Carlo, los procesos físicos son simulados teóricamente sin necesidad de resolver completamente las ecuaciones del sistema. Sin embargo es necesario conocer las funciones de densidad de probabilidad (pdf por sus siglas en inglés) que describen el comportamiento del sistema. Se puede estimar resultados incluso si el problema no tiene un contexto probabilístico.

El código MCNP construye un modelo estocástico, que basándose en las funciones de densidad modela secuencialmente eventos individuales de una variable aleatoria. Es decir, teóricamente se siguen todos los eventos o interacciones que sufre cada partícula desde su origen hasta que alcanza una condición terminal (absorción, escape, energía de corte, etc.). Lo mismo se aplica para todas las partículas creadas en el proceso. Para cualquier evento, el MCNP genera un número aleatorio fundamentándose en las pdf's, que definirá el tipo de interacción y otros parámetros. Posteriormente, se calcula el valor esperado de todos los eventos simulados. El valor esperado de una o varias variables aleatorias es equivalente al valor de una cantidad física del sistema estudiado.

Una historia comienza calculando mediante procesos aleatorios la probabilidad de que se creó una partícula. De manera análoga se obtiene la energía, la posición y la dirección inicial de la trayectoria de dicha partícula, tomando en cuenta que estas son independientes entre sí. Posteriormente se simula la distancia libre media que recorrerá antes de interactuar y el tipo de colisión que sufrirá. La energía y dirección de las partículas dispersadas y de las partículas secundarias son variables aleatorias que también se calculan en la simulación. Este proceso continua para cada una de las partículas que se van generando hasta que se alcanza una condición terminal. En la Figura 2.1 se muestra una historia aleatoria de eventos que sufre un fotón al incidir en un material.

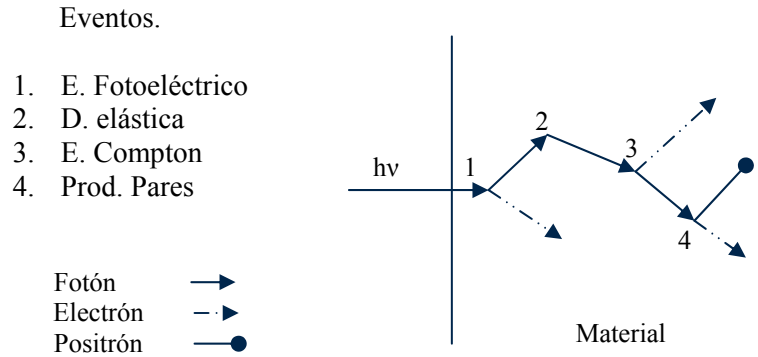


Figura 2.1: Simulación de una secuencia de eventos.

2.3 Generación de números aleatorios.

La generación de una buena secuencia de números aleatorios es la base probabilística del método de Monte Carlo. Cada número aleatorio debe ser totalmente independiente de los otros números de la secuencia. Además, dos generadores aleatorios independientes deben proporcionar estadísticamente el mismo valor promedio de salida. Una manera de obtener estos números es mediante un proceso físico aleatorio, como puede ser el tiempo entre la detección de dos partículas cargadas provenientes de una fuente radioactiva que se registra en un contador Geiger. La desventaja de este proceso es que la generación de números aleatorios es muy lenta comparada con los requerimientos del código MCNP, y además se requiere de mucho espacio de memoria. Asimismo esta secuencia no puede ser duplicada, por lo que no será posible corroborar los resultados obtenidos de la simulación [10].

Se prefiere crear una secuencia de números pseudo aleatorios, es decir, números generados de manera determinística que superan aproximadamente 12 pruebas de aleatoriedad [11]. Las pruebas que deben verificar son de carácter empírico, cuando la computadora realiza evaluaciones estadísticas con la secuencia generada; y de carácter teórico, basándose en métodos numéricos. La llamada “Prueba espectral” (*Spectral Test* en inglés) es hoy en día la prueba aleatoria más importante y está basada en Transformadas de Fourier. Las secuencias pseudo aleatorias tienen un ciclo de repetición por su carácter

determinístico, pero se busca que su período sea muy grande. El procedimiento para componer una lista de números pseudo aleatoria debe tener una teoría de respaldo; no puede simplemente inventarse un método aleatorio, ya que la secuencia generada entonces tiende a abarcar un ciclo de repetición muy pequeño. Por otra parte, además de las secuencias pseudo aleatorias están las cantidades probabilísticas relacionadas con los procesos físicos que intervienen en una simulación.

Durante años, los científicos han calculado teórica y experimentalmente las secciones eficaces y las funciones de densidad de probabilidad para distintos eventos. Toda esta información es parte de la base de datos del código MCNP; y se divide en diferentes tablas de datos nucleares: interacción de neutrones de energía continua, interacción de neutrones con reacción discreta, interacción de fotones, secciones eficaces de dosimetría de neutrones, neutrones térmicos, interacciones de electrones e interacciones neutrón/neutrón, neutrón/fotón, fotón/fotón y partículas cargadas que se comportan como neutrones. Las principales bibliotecas son editadas en Brookhaven National Laboratory (llamada ENDF – Evaluated Nuclear Data File) y en Lawrence Livermore Laboratory. En esta tesis se utiliza la biblioteca *ell* para electrones y *mcplib1* para fotones.

2.3.1 *Distribución uniforme.*

Von Neumann fue el primero en proponer un procedimiento para obtener números pseudo aleatorios¹ que corresponden a una distribución uniforme. Su método consistía en tomar un número raíz de 10 dígitos, x_i , y elevarlo al cuadrado; los 10 dígitos intermedios de x_i^2 serán el siguiente número aleatorio en la lista, x_{i+1} . Se vuelve a repetir el procedimiento tomando como raíz el número x_{i+1} . Sin embargo, este método tiene un ciclo de repetición muy corto, y además si la raíz llega a ser cero solo se generan puros ceros en la secuencia ($0^2=0$).

¹ A partir de aquí llamaremos a los números pseudo aleatorios simplemente números aleatorios.

Actualmente se utiliza el método denominado *Congruencia Lineal* para obtener una secuencia aleatoria a partir de una distribución uniforme [11]. La fórmula recursiva que genera la secuencia es:

$$x_{n+1} = (ax_n + c) \bmod m \quad (2.1)$$

donde a es el multiplicador, c es el incremento, m el módulo y x_0 el valor inicial. Todos estos parámetros deben ser mayores que cero y cumplir cada uno con ciertas reglas para asegurar una secuencia aleatoria. Si los parámetros son escogidos adecuadamente, el período máximo de repetición que puede alcanzar la secuencia es igual a m . Existen variaciones de este método que también son muy utilizadas.

2.3.2 Distribuciones continuas.

En aplicaciones más concretas, se requiere generar números aleatorios que correspondan a diferentes distribuciones continuas no uniformes. En principio, cualquier número aleatorio característico de una distribución continua se puede obtener a partir de la generación de secuencias aleatorias uniformes; no obstante, existen métodos dependiendo del tipo de distribución que hacen más eficiente el proceso. El método general para cualquier distribución continua se basa en las funciones acumuladas de distribución de probabilidad (cdf por sus siglas en inglés)², $F(x)$. Sabemos [13] entonces que:

$$y = F(x) \Leftrightarrow x = F^{-1}(y) \quad (2.2)$$

El problema ahora se reduce a encontrar $F^{-1}(y)$; pero si esta existe y es fácil de encontrar, se puede obtener una variable aleatoria con una distribución continua, X , a partir de una variable aleatoria de distribución uniforme Y , tal que:

$$X = F^{-1}(Y) \quad (2.3)$$

² $F(x) = \int_{-\infty}^x f(x') dx'$, donde $f(x)$ es una pdf. Es decir, es la suma de todas las probabilidades asociadas a valores menores o iguales a un número x . $F(x)$ incrementa monótonicamente en el rango $[0,1]$. También se le conoce como función de distribución de probabilidad. [12]

2.3.2.1 Método de Rechazo.

El *Método de Rechazo* también es utilizado para obtener números aleatorios de una distribución continua. Para explicar los conceptos que intervienen en esta técnica, se describirá el ejemplo de la trayectoria libre media. La trayectoria libre media λ , es el valor esperado de la distancia R entre dos colisiones que sufre una partícula en el sistema. Se sabe que la función de densidad de probabilidad de R es:

$$p(R) = \sigma e^{-\sigma R} \quad (2.4)$$

donde σ es la sección eficaz. Entonces:

$$\lambda = \langle R \rangle = \int_a^b R \cdot p(R) dR = \frac{1}{\sigma} \quad (2.5)$$

Esta relación tiene sentido físico y se puede comprobar experimentalmente.

El llamado método de rechazo se utiliza por MCNP cuando se desea encontrar por ejemplo las R_n de 10 000 000 partículas que correspondan a dicha función de densidad $p(R)$. Este procedimiento genera primero dos números aleatorios ξ_1, ξ_2 entre 0 y 1. Después se obtiene un valor aleatorio de la distancia tal que $a < R < b$, interpolando con:

$$R = a + (b - a)\xi_1 \quad (2.6)$$

Finalmente se rechaza el valor de R si $\xi_2 > p(R)$ ya que no corresponde a dicha función de densidad de probabilidad; y se acepta y guarda para futuros cálculos si $\xi_2 < p(R)$.

2.3.2.2 Distribución normal.

El *Método Polar* es la técnica más utilizada para encontrar números aleatorios a partir de una distribución normal ($\sigma=1, \mu=0$). El procedimiento es [11]:

1. Generar dos números aleatorios uniformes (U_1, U_2) en el rango $[0,1]$.
2. Obtener $V_1=2U_1-1$ y $V_2=2U_2-1$, para conseguir dos números uniformemente distribuidos en $[-1,1]$.
3. Calcular $S = V_1^2 + V_2^2$

4. Si $S \geq 1$, regresar al paso 1. Si $S < 1$, calcular los números aleatorios normalmente distribuidos, X_1 y X_2 , a partir de:

$$X_1 = V_1 \sqrt{\frac{-2 \ln S}{S}} \quad \text{y} \quad X_2 = V_2 \sqrt{\frac{-2 \ln S}{S}} \quad (2.7)$$

Si la distribución normal tiene una media μ y una desviación estándar σ , entonces la variable aleatoria Y se obtiene con $Y_n = \mu + \sigma X_n$.

Knuth [11] explica que la teoría detrás de este método consiste en encontrar las coordenadas de un punto aleatorio uniformemente distribuido, (V_1, V_2) , dentro de un círculo unitario S . Convirtiendo a coordenadas polares, $V_1 = R \cos \Theta$ y $V_2 = R \sin \Theta$, obtenemos $S = R^2$, $X_1 = R' \cos \Theta$ y $X_2 = R' \sin \Theta$, donde $R' = \sqrt{-2 \ln S}$. Θ está uniformemente distribuida entre 0 y 2π ; y la probabilidad de que $R' \leq r$ es igual a la probabilidad de que $S \geq e^{-r^2/2}$, que al tratarse de un círculo unitario se tiene que $1 - e^{-r^2/2} \geq 0$. Por consiguiente la probabilidad de que R' se encuentre entre r y $r+dr$ es la derivada de $1 - e^{-r^2/2}$, es decir, $re^{-r^2/2} dr$; de la misma manera, la probabilidad de que Θ se encuentre entre θ y $\theta+d\theta$ es $\frac{1}{2\pi} d\theta$. Regresando a coordenadas cartesianas, tenemos:

$$F(X_1 \leq x_1, X_2 \leq x_2) = \left(\sqrt{\frac{1}{2\pi}} \int_{-\infty}^{x_1} e^{-x^2/2} dx \right) \left(\sqrt{\frac{1}{2\pi}} \int_{-\infty}^{x_2} e^{-y^2/2} dy \right) \quad (2.8)$$

donde se puede observar que X_1 y X_2 corresponden a una distribución normal y además son independientes.

2.3.2.3 Distribución exponencial.

El tiempo transcurrido entre la emisión de dos partículas por una fuente radioactiva tiene una distribución exponencial con media μ , si en promedio una partícula es emitida cada μ segundos. El proceso se describe por $F(x) = 1 - e^{-x/\mu}$. Aunque existen otros métodos

para obtener números aleatorios con esta distribución, el denominado *Método Logarítmico* es el más utilizado por su simplicidad [11]. Si $y = F(x)$, entonces $x = F^{-1}(y) = -\mu \ln(1-y)$. Generando un número aleatorio uniformemente distribuido, U , se obtiene una variable aleatoria X con una distribución exponencial a partir de:

$$X = -\mu \ln(1-U) \quad (2.9)$$

2.3.3 Vectores aleatorios.

Para completar la simulación de todo sistema físico, es necesario saber la orientación de una partícula después de que sufre una interacción. En el método más sencillo para seleccionar una orientación aleatoria en tres dimensiones, se debe primero generar un número aleatorio uniforme U en el rango $[0, 1]$. U se escala para obtener un número aleatorio φ que este uniformemente distribuido entre $[0, 2\pi]$, es decir, $\varphi = 2\pi U$. De manera similar se obtiene un número aleatorio ω distribuido uniformemente entre -1 y 1 . Las coordenadas del vector de orientación son [14]:

$$\begin{aligned} x &= \sqrt{1-\omega^2} \cos \varphi \\ y &= \sqrt{1-\omega^2} \sin \varphi \\ z &= \omega \end{aligned} \quad (2.10)$$

2.4 Error Relativo.

Existe una diferencia entre el valor esperado verdadero de una función f y el valor de la misma función cuando solo se toma un muestreo de N variables estadísticas independientes (Número de partículas a simular) [15]. Para saber si nuestra simulación arroja un resultado confiable, es importante tomar en cuenta el error relativo R . Este dato representa la precisión estadística y es calculado por el programa a partir cada historia aleatoria.

El error esperado en $\langle f \rangle_N$ se estima con la desviación estándar, ya que:

$$\delta_N^2 = \left\langle (f(x_1, x_2, \dots, x_N) - f(x_1, x_2, \dots, \infty))^2 \right\rangle \quad (2.11)$$

Sin embargo como no se tiene un número infinito de variables, el error se estima considerando cada selección aleatoria como un estimador independiente de $\langle f \rangle_N$; así:

$$\delta_N^2 = \frac{1}{N} \sum_{i=1}^N [f(x_i) - \langle f \rangle_N]^2 \quad (2.12)$$

$$= \langle h^2 \rangle_N - \langle h \rangle_N^2 \quad (2.13)$$

La suma de las desviaciones estándar, ε_{sum}^2 , es N veces el valor de δ_N^2 . Por consiguiente el error de expectación es:

$$\varepsilon_{sum} = \sqrt{N} \delta_N \quad (2.14)$$

El error relativo es la media de dicho error de expectación, de tal manera que:

$$R = \frac{\varepsilon_{sum}}{N} = \frac{\delta_N}{\sqrt{N}} \quad (2.15)$$

Es importante recalcar que R es proporcional a $\frac{1}{\sqrt{N}}$, es decir, mientras mayor sea el número de partículas simuladas se podrá obtener un resultado más preciso. Para reducir el error relativo en una simulación se puede aumentar N o reducir el valor de δ_N . Sin embargo, el presupuesto limita el incremento que se puede tener en N por el tiempo de computadora que requiere. Por esta razón, existen técnicas de reducción de varianza en el código MCNP que se basan en disminuir el valor de δ_N [16].

La Tabla 2.1 resume el significado del valor de R.

Rango de R	Calidad de la simulación
0.5 a 1.0	Sin sentido
0.2 a 0.5	Mala
0.1 a 0.2	Cuestionables
< 0.10	Generalmente aceptables excepto para detectores puntuales
< 0.05	Generalmente aceptables para detectores puntuales

Tabla 2.1: Error Relativo en MCNP [16].

2.5 Simulación.

Para realizar una simulación, se debe crear un archivo de entrada llamado *inp* (Ver Apéndice 1). Este archivo contiene de manera estructurada información sobre la geometría, los materiales utilizados, las secciones eficaces a utilizar, la localización, características y tipo de la fuente (electrones, fotones o neutrones), los *tallies* y cualquier técnica de reducción de varianza. Los *tallies* o conteos son instrucciones que le dicen al programa el tipo de datos que se desea calcular; por ejemplo, la corriente y el flujo de algún tipo de partículas resultantes, o la deposición y distribución de energía.

El programa MCNP leerá las instrucciones del archivo *inp*, realizará la simulación y creará un nuevo archivo llamado *out* (Ver Apéndice 2). El *out* incluirá los resultados generados por cada *tally*, los errores producidos y algunas tablas que resumen el proceso de simulación.