

# Tema 12

## Métodos Monte Carlo

### 12.1. Introducción

#### 12.1.1. Un poco de historia

El nombre del método hace alusión directa al famoso Casino de Monte-Carlo, en Mónaco, y fue acuñado en los años 40 por científicos trabajando en el proyecto de la bomba nuclear en Los Álamos:

“During World War II physicists at the Los Alamos Scientific Laboratory came to a knotty problem on the behavior of neutrons. How far would neutrons travel through various materials? The question had a vital bearing on shielding and other practical considerations. But it was an extremely complicated one to answer. To explore it by experimental trial and error would have been expensive, time-consuming and hazardous. On the other hand, the problem seemed beyond the reach of theoretical calculations. The physicists had most of the necessary basic data: they knew the average distance a neutron of a given speed would travel in a given substance before it collided with an atomic nucleus, what the probabilities were that a neutron would bounce off instead of being absorbed by the nucleus, how much energy the neutron was likely to lose after a given collision, and so on. However, to sum all of this up in a practicable formula for predicting the outcome of a whole sequence of such events was impossible.

At the crises the mathematicians John von Neumann and Stanislaus Ulam cut the Gordian knot with a remarkably simple stroke. They suggested a solution which in effect amounts to submitting the problem to a roulette wheel. Step by step the probabilities of the separate events are merged into a composite picture which gives an approximate but workable answer to the problem.

The mathematical technique von Neumann and Ulam applied had been known for many years. When it was revived for the secret work at Los Alamos, von Neumann gave it the code name “Monte Carlo”. The Monte Carlo method was so successful on neutron diffusion problems that its popularity later spread. It is now being used in various fields, notably in operations research.”

Extracto sacado de: D.D. McCracken (1955), "The Monte Carlo Method", *Scientific American*, vol. 192, pp 90-95.

Como se menciona en este extracto, en aquella época de desarrollo nuclear era necesario investigar la interacción que experimentaba la radiación ionizante cuando atravesaba un medio material. Fundamentalmente, la cuestión era, ¿qué fracción de la energía de la radiación es absorbida por el medio? Aparte de las aplicaciones industriales, la importancia de esta cuestión era, y sigue siendo, principalmente biológica. Para comprender la importancia de este problema primero debemos saber qué se entiende por radiación ionizante.

Se define radiación ionizante como aquella radiación, bien electromagnética (compuesta de fotones, como los rayos  $\gamma$  o los rayos X)<sup>1</sup>, o bien de partículas, como la radiación  $\alpha$  (núcleos de Helio), la radiación  $\beta^-$  (electrones) o la radiación  $\beta^+$  (positrones), cuyos componentes tienen suficiente energía como para ionizar un átomo o molécula, es decir, que son capaces de arrancar un electrón de su orbital atómico o molecular. Esta capacidad depende únicamente de la energía de la radiación, y no de su flujo o intensidad (número de partículas incidentes por unidad de tiempo y de superficie). La radiación ionizante es emitida durante procesos nucleares (desintegración nuclear de elementos radiactivos, procesos nucleares de fisión y fusión), durante la desaceleración brusca de electrones (proceso conocido como *bremssstrahlung*) y en la reacción de aniquilación positrón-electrón. En general, sólo los fotones y las partículas cargadas pueden interactuar con los electrones de una forma suficientemente intensa como para ionizar la materia. Los neutrones, al no ser partículas con carga, no son capaces de ionizar directamente un átomo aunque sí pueden producir radiaciones ionizantes secundarias cuando interactúan con los núcleos atómicos.

Desde el punto de vista biológico, la capacidad de ionización es, sin lugar a dudas, el aspecto más importante de la interacción radiación-materia. Por esta razón, el término *radiación ionizante* suele aparecer con mucha frecuencia en diversos contextos y ámbitos de la física médica: medicina nuclear, radiodiagnóstico, radioterapia, etc. Cuando un haz de radiación ionizante atraviesa un tejido u otro medio absorbente, parte de la energía del haz incidente es transferida al medio —también se utiliza el término energía depositada. En el caso de tejidos biológicos, después de la transferencia inicial de energía se sucede una cadena de eventos, físicos, químicos y biológicos, por este orden, que finalmente conducen al daño, reversible o irreversible, del tejido. Los efectos de la radiación ionizante sobre los seres vivos pueden ser muy graves. Desde el descubrimiento de los rayos X por Röntgen en 1895, pocos conjuntos de datos físicos han estado tan omnipresentes o han sido tan necesarios y utilizados como los datos sobre la transmisión y absorción de rayos X en materiales biológicos, en materiales de protección contra la radiación y en materiales dosimétricos. Esta importancia puede hacerse extensiva a los diversos tipos de radiación ionizante.

Actualmente, el estudio de la interacción entre la radiación ionizante y la materia es un problema de gran relevancia en muchos campos de la ciencia,<sup>2</sup> y muy en particular en la física médica. La lista es larga e incluye importantes especialidades de la medicina moderna como la medicina nuclear o la radiología, dedicadas fundamentalmente al diagnóstico y tratamiento de enfermedades. Importantes técnicas no invasivas de

<sup>1</sup>De la misma naturaleza que la luz blanca que todos vemos pero con mucha más energía.

<sup>2</sup>Entre las muchas aplicaciones derivadas de esta interacción y su estudio destacamos la dosimetría, la protección radiológica, la microscopía electrónica, diferentes técnicas de espectroscopía de superficies, el diseño de detectores de radiación, etc.

imagen utilizadas en radiología —radiodiagnóstico o radiología diagnóstica— y en medicina nuclear para el diagnóstico están basadas en los mismos principios físicos de la interacción radiación materia. Tal es el caso de la imagen por rayos X, la imagen por resonancia magnética o IRM (del inglés *Magnetic Resonance Imaging*, MRI, o *Nuclear Magnetic Resonance Imaging*, NMRI), la tomografía axial computerizada o TAC (del inglés *Computed tomography*, CT) o la tomografía por emisión de positrones o TEP (del inglés de *Positron Emission Tomography*, PET). Una de las principales aplicaciones médicas de las radiaciones ionizantes es la radioterapia, práctica médica muy utilizada para el tratamiento contra el cáncer en la que se hace uso de los efectos producidos por la radiación sobre las células para destruir tejidos tumorales.

Volviendo al problema que se planteaban los científicos de los años 40, el estudio de la interacción entre la radiación ionizante y la materia, y en particular, el cálculo de la energía depositada en el medio era un problema extremadamente complejo para ser abordado a partir de la teoría del transporte construida sobre la ecuación de transporte de Boltzmann, bien conocida por entonces. Sin embargo, los matemáticos John von Neumann and Stanislaus Ulam propusieron una solución numérica que básicamente se reduce a realizar tiradas con una ruleta de casino, de ahí el nombre de *método de Monte Carlo*. Veamos en qué consistía esa original solución.

Por entonces ya se sabía que cuando un haz de radiación ionizante penetra en un medio material, las partículas que componen la radiación —partículas primarias<sup>3</sup>— experimentan una sucesión de interacciones —o “choques”— con los átomos o moléculas del medio a través de las cuales transfieren su energía, y en las que se pueden producir otras partículas que se denominan *secundarias*. De este modo, el paso de la partícula primaria origina una cascada de partículas secundarias que arrastran parte de su energía dispersándola por el medio hasta que finalmente toda la energía inicial es depositada y no quedan partículas en movimiento. También se conocían muy bien los diversos tipos de interacción que podían tener lugar entre los diferentes tipos de radiación y el medio. Sin embargo, el transporte de la radiación en la materia es un proceso eminentemente aleatorio. Esta aleatoriedad procede fundamentalmente de la naturaleza cuántica de las interacciones y de la propia naturaleza desordenada de muchos medios microscópicos. Esto quiere decir que es imposible predecir el comportamiento de una partícula individual.

A pesar de esto, Von Neumann y Ulam mostraron que el comportamiento promedio de un gran número de partículas podía ser reproducido de forma muy precisa si se conocen los mecanismos de interacción y las leyes de probabilidad que los gobiernan. En aquella época la electrodinámica cuántica de Dirac ya proporcionaba estas leyes, es decir, ya se sabía cuál era la probabilidad, por unidad de camino libre recorrido, de que se produjera una interacción entre la partícula ionizante y el medio, la probabilidad de que ocurriera uno u otro tipo de interacción en concreto, la probabilidad de que la partícula cediera una cierta cantidad de energía en la interacción y de que la dirección de su movimiento cambiara un cierto ángulo, y también la probabilidad de que crease una partícula secundaria con una cierta energía y emitida en una cierta dirección. Es decir, se sabía todo, pero todo obedecía leyes de probabilidad.

La solución de Von Neumann y Ulam fue simular con el ordenador historias de partículas ionizantes utilizando números aleatorios (generados mediante una ruleta de casino, por ejemplo). La historia de cada partícula es vista como una sucesión cronológica de interacciones aleatorias con el medio, en las que la partícula puede

<sup>3</sup>Partículas sin masa, como los fotones, o partículas con masa como electrones o positrones.

sufrir cambios en su dirección, perder energía e incluso producir partículas secundarias. Entre dos interacciones consecutivas, la partícula describe un movimiento libre recorriendo una trayectoria lineal. La longitud del camino libre recorrido entre dos interacciones consecutivas, el tipo de interacción que tiene lugar, y la pérdida de energía y el cambio en la dirección de su movimiento son obtenidos a partir del muestreo mediante números aleatorios de las correspondientes leyes de probabilidad dadas por la teoría. Por ejemplo, supongamos que estamos en un punto del espacio en el que se produce una interacción y tenemos que decidir qué tipo de interacción ocurre; tenemos tres tipos de interacción posible: A, B, C, con la siguiente distribución de probabilidad:  $\text{prob}(A) = 0,5$ ,  $\text{prob}(B) = 0,25$  y  $\text{prob}(C) = 0,25$ . Podemos lanzar nuestra ruleta de 37 números (desde el 0 hasta el 36) y decidir el tipo de interacción que ocurre en función del número  $X$  obtenido del lanzamiento de la ruleta. El algoritmo de elección es el siguiente:

- Si  $X \leq 18$ , entonces ocurre el proceso A.
- Si  $18 < X \leq 27$ , entonces ocurre el proceso B.
- Si  $27 < X \leq 36$ , entonces ocurre el proceso C.

Como todos los números de la ruleta son equiprobables, este algoritmo respeta las probabilidades iniciales, y aunque no podamos predecir cuál será el proceso que ocurrirá en esa interacción, sí que reproduciremos el comportamiento promedio de muchas interacciones. Los valores de las otras variables aleatorias que caracterizan la historia de la partícula ionizante, como el camino libre recorrido entre dos interacciones consecutivas, la energía perdida o el cambio de dirección, también son elegidos de un modo similar realizando nuevos lanzamientos de la ruleta.

De este modo, la simulación de Monte Carlo de la historia de una partícula ionizante se reduce simplemente a la generación secuencial de números aleatorios para muestrear las leyes de probabilidad de las magnitudes relevantes. Este muestreo va trazando paso a paso (“tirada a tirada”) el “camino aleatorio” que describe la partícula al atravesar el medio. La historia de la partícula es seguida hasta que ésta no puede contribuir con ninguna información de interés al problema. En general, esto ocurre cuando la partícula abandona el volumen de estudio o cuando su energía disminuye por debajo de un umbral a partir del cual se puede considerar que, a todos los efectos, la partícula ha sido absorbida por el medio.

A diferencia de los métodos deterministas (no aleatorios), los cuales resuelven la ecuación de transporte para el comportamiento promedio de las partículas, el método de Monte Carlo no resuelve explícitamente esta ecuación. En lugar de ello, simula la historia de partículas individuales y después promedia para obtener los resultados.

En la actualidad, el término Monte Carlo se utiliza para hacer referencia a un tipo general de métodos numéricos que utilizan números aleatorios para resolver problemas a partir de la información estadística que se tiene sobre los mismos. La simulación de Monte Carlo consiste en generar una secuencia de eventos de azar cuya probabilidad de ocurrencia está determinada por la información estadística del problema y cuyo resultado representa un estimador estadístico, sujeto por tanto a incertidumbres estadísticas, de la solución del problema. Por consiguiente, la característica esencial de la simulación de Monte Carlo es el uso de números aleatorios. En las últimas décadas, y gracias a la creciente capacidad de cálculo y memoria de los computadores, los métodos de simulación de Monte Carlo han emergido como una exitosa y poderosa

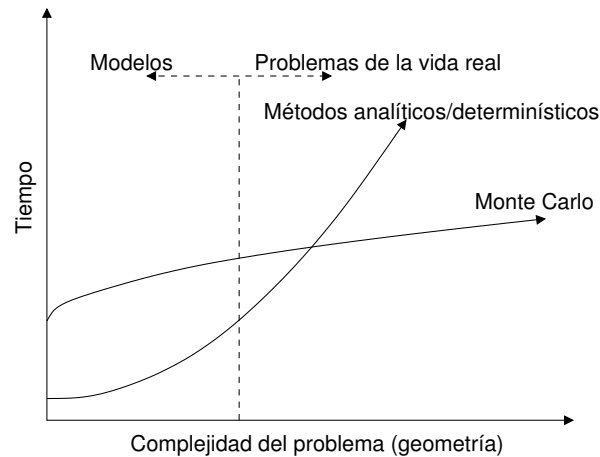


Figura 12.1: Métodos deterministas vs. Monte Carlo. El comportamiento mostrado es cualitativo.

herramienta de investigación para resolver problemas de gran complejidad en diferentes ámbitos de la ciencia, problemas que son prácticamente inabordables desde un punto de vista analítico o aplicando métodos numéricos convencionales.

La principal pega de los Métodos de Monte Carlo radica en su naturaleza aleatoria, por lo que todos los resultados conllevan asociadas unas ciertas incertidumbres estadísticas. El resultado de una simulación es una estimación que debe ser dada dentro de un intervalo de confianza concreto alrededor del valor verdadero. La magnitud del error o incertidumbre estadística asociada al resultado puede ser reducida incrementando el número de historias (simulaciones), lo que conlleva un mayor coste computacional: cuanto mayor es el número de historias, menor es el intervalo de confianza alrededor del comportamiento promedio verdadero de las partículas. Por consiguiente, detrás de cada simulación de Monte Carlo existe un conflicto, sin solución aparente, entre la fiabilidad del resultado y el tiempo de cálculo.

En general, cuando la complejidad del problema crece, los métodos de Monte Carlo resultan ser mucho más eficientes que los métodos deterministas. En términos de complejidad computacional se dice que el orden de los métodos deterministas es superior al del método de Monte Carlo. En la siguiente figura se ha ilustrado este comportamiento.

### 12.1.2. Un ejemplo muy simple

Un ejemplo archiconocido del uso del método de Monte Carlo es el cálculo del número  $\pi$ . Incluso podemos realizar este experimento en casa.

Tomemos un círculo de radio  $R$  (una diana) inscrito dentro de un cuadrado de lado  $2R$  (un tablero). El método de Monte Carlo consiste en ir seleccionando aleatoriamente puntos del cuadrado e ir contando cuántos de ellos han caído dentro del círculo. Continuando con el ejemplo de la diana, el procedimiento es estadísticamente similar a tirar dardos con los ojos cerrados, esto es, “al azar”.

Supongamos que de los  $N$  dardos tirados (o de las  $N$  veces que hemos tirado el mismo dardo),  $n$  han dado en la diana. La estadística de nuestro problema nos dice que la probabilidad  $P$  de que un punto elegido al azar dentro del cuadrado esté dentro

del círculo será:

$$P = \frac{\text{área}_{\text{diana}}}{\text{área}_{\text{cuadrado}}} = \frac{\pi}{4}$$

mientras que nuestro *estimador* de esta probabilidad vendrá dado por  $n/N$ . Despejando obtenemos que nuestra estimación de  $\pi$  será

$$\pi \simeq \frac{4n}{N}$$

A medida que vayamos repitiendo el experimento, nuestro estimador de  $\pi$  se irá aproximando al valor exacto. En efecto, la *ley de los grandes números* establece que

$$\lim_{N \rightarrow \infty} \frac{4n}{N} = \pi$$

Aunque el método de Monte Carlo se utiliza con éxito para resolver problemas que no pueden ser resueltos analíticamente, debemos reconocer que no es el mejor método para estimar  $\pi$ : para calcular cada cifra significativa se necesitan aproximadamente del orden de 10 veces más lanzamientos que los necesarios para calcular la cifra significativa anterior.

## 12.2. Caminantes aleatorios y difusión browniana

Un proceso de difusión consiste en el desplazamiento de masa o energía desde una región del espacio en la que abunda a otra región del espacio en la que existe déficit. Así, la difusión térmica consiste en el paso paulatino de calor de una parte de un cuerpo que está caliente a otra parte que está fría. La difusión de una gota de colorante en un vaso de agua es el reparto paulatino del colorante, desde el punto en que cae la gota, donde la concentración del colorante es máxima, hacia las zonas de agua transparente, con el resultado final de que se colorea la totalidad del agua del vaso.

En 1828, Robert Brown (botánico escocés, 1773-1858) observó bajo el microscopio que, cuando un grano de polen cae en el agua, realiza un movimiento sin dirección definida. A lo largo de este “paseo aleatorio”, el grano de polen visita la totalidad del portamuestras, si se tiene la paciencia de seguirlo (como hizo Brown). Este movimiento de una partícula se llama desde entonces movimiento Browniano. En 1905, Albert Einstein explicó este movimiento a través de un modelo en el que la partícula browniana (el grano de polen) choca con las moléculas del fluido en el que se halla, las cuales se mueven en distintas direcciones debido a la agitación térmica. Cuando se tienen muchas partículas brownianas, estas colisiones hacen que las mismas acaben repartidas por todo el espacio, del mismo modo que sucede en el caso de la difusión. Por eso se habla de difusión browniana.

Como vimos más arriba (al hablar del *teorema central del límite*, sección 11b.2.2), el resultado de la suma de todos estos desplazamientos aleatorios causados por las colisiones térmicas acabará dando una distribución gaussiana de las partículas brownianas alrededor del punto de partida; esta distribución vendrá dada, en función del tiempo  $t$ , para la coordenada  $x$  de las partículas, por

$$p(x, t) = \frac{1}{\sqrt{4\pi Dt}} \exp\left(-\frac{x^2}{4Dt}\right)$$



donde  $\mathcal{D}$  se llama *coeficiente de difusión* (y se asume que todas las partículas tienen coordenada  $x = 0$  en el instante inicial  $t = 0$ ).

Este resultado es muy interesante, puesto que nos enseña que en un proceso difusivo, la varianza de la distribución de las observaciones crece linealmente con el tiempo. En el caso de una partícula que se mueve en un espacio  $d$ -dimensional y su posición viene dada por el vector  $\mathbf{r}$ , se cumple:

$$\langle (\Delta \mathbf{r})^2 \rangle(t) = 2d\mathcal{D}t$$

Esta ecuación deducida por Einstein es la generalización, para un movimiento en  $d$  dimensiones del proceso de difusión Browniana.

### 12.2.1. Difusión browniana en 1-D

Para simular un proceso de difusión browniana en una dimensión (sobre la línea), se recurre al modelo del *caminante aleatorio*<sup>4</sup>. Este caminante avanza o retrocede un paso, según le salga cara o cruz del lanzamiento de una moneda, respectivamente. ¿Cuál es la probabilidad de encontrar al caminante a una distancia  $x$  del punto de partida? La respuesta es que esta probabilidad vendrá dada por una distribución gaussiana. Sin embargo, podemos calcular la trayectoria del caminante con un programa y, así, comprobar nuestra hipótesis.

En el programa que simula un movimiento browniano unidimensional, utilizaremos esta biblioteca de funciones para decidir si la (pseudo)moneda devuelve cara o cruz: si la variable aleatoria uniformemente distribuida en  $[0, 1)$  es mayor o menor que 0,5, respectivamente. El código de este programa se muestra en el listado 12.1.

El programa recibe como argumento de línea de comandos el número de pasos que dará el caminante (guardado en la variable “npts”). Posteriormente, en un bucle, modifica la posición del caminante (variable “xp”, inicialmente tomada como 0); esta posición se mide en “número de pasos desde el origen”, por lo que es suficiente con que sea de tipo entero.

El programa muestra por terminal las posiciones del caminante en los sucesivos tiempos (indicados por la variable del bucle, “n”). En la figura 12.2 se muestra el recorrido del caminante, ¡realmente aleatorio!.

---

**Listado 12.1:** Programa que calcula el recorrido de un caminante aleatorio: su posición en función del número de pasos dados.

---

```

1 #include <stdio.h>
2 #include <stdlib.h>
3 #include "libprobabilidad.h"
4
5 int main(int argc, char** argv)
6 {
7     int npts, n;
8     int xp;
9
10    if( argc<2 )

```

---

<sup>4</sup>Además del nombre “políticamente correcto” de *caminante aleatorio* (*random walker*), también se habla del modelo de *peatón* o de la *hormiga borracha* (*drunken ant*).

```

11     {
12         printf("Uso del programa:\n %s <no. puntos>\n",
13             argv[0]);
14         exit(0);
15     }
16
17     npts=atoi(argv[1]);
18     if(npts<1)
19     {
20         npts=1;
21     }
22
23     xp=0;
24
25     for(n=0; n<npts; n++)
26     {
27         (void)uniforme();
28     }
29
30     for(n=0; n<npts; n++)
31     {
32         if( uniforme()>=0.5 )
33         {
34             xp++;
35         } else {
36             xp--;
37         }
38
39         printf("%d\t%d\n", n, xp);
40     }
41
42     return 0;
43 }

```

---

**Ejercicio 12.1.** Comprobar que nuestro caminante aleatorio cumple la relación de Einstein, y que la varianza de la distribución de probabilidad de encontrarlo en una posición  $x$  crece linealmente con el tiempo. Para ello, hágase uso de una “colectividad” de caminantes. Cada uno de ellos ejecutará su danza según el resultado de una “moneda”: nuestro único generador de números aleatorios.

### Consejos:

- La posición de los “NBROWNIANOS” caminantes debe ser un “array” de posiciones, “int xp[NBROWNIANOS]”. Esto hace que en nuestro código anterior, cualquier operación con “xp” se convierta en un bucle de actualización de los “xp[]”.



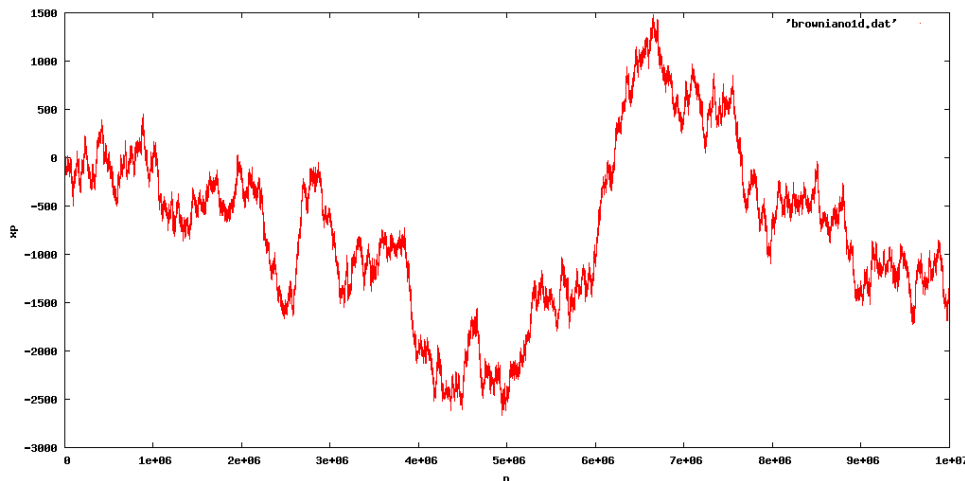


Figura 12.2: Gráfica de la posición de un caminante aleatorio unidimensional en función del tiempo (generado por el programa del listado 12.1).

- Ya no estaremos interesados en las posiciones concretas de todos los caminantes, sino en los valores medios de todas estas posiciones y en su varianza. Esperamos que el valor medio, si realmente cada uno ha “tirado por su lado”, sea cero; pero esto no siempre será exactamente así. Respecto a ese valor medio, calcularemos la media cuadrática de los desplazamientos, para calcular así la varianza.

Representar con “Gnuplot” la evolución temporal de la media  $\bar{x}(n)$  y la varianza  $\Delta x^2(n)$ , en función del tiempo  $n$ , para comprobar la relación de Einstein.

Ajustar por mínimos cuadrados los datos generados a la expresión

$$\Delta x^2(n) = 2d\mathcal{D}n$$

y obtener así el coeficiente de difusión browniana,  $\mathcal{D}$  (recuérdese que, en este apartado,  $d = 1$ ).

### 12.2.2. Difusión browniana en 2-D

Como en el caso de difusión unidimensional, en dos o tres dimensiones ( $d = 2, 3$ , respectivamente), se sigue cumpliendo la relación de Einstein. En este caso, el vector posición de la partícula  $\mathbf{r}$ , relativo al origen del movimiento, y con longitud dada por

$$r = \sqrt{\sum_{i=1}^d x_i^2} \quad \begin{cases} r = \sqrt{x^2 + y^2} & \text{en 2-D} \\ r = \sqrt{x^2 + y^2 + z^2} & \text{en 3-D} \end{cases}$$

figurará en la expresión de la densidad de probabilidad por unidad de volumen (el elemento de volumen en  $d$ -dimensiones es  $dV_d = dx_1 dx_2 \cdots dx_d$ )

$$p(\mathbf{r}, t) dV_d = \frac{1}{(4d\pi\mathcal{D}t)^{d/2}} \exp\left(-\frac{r^2}{4d\mathcal{D}t}\right) dx_1 dx_2 \cdots dx_d$$

Se puede ver que la probabilidad por unidad de distancia al origen del movimiento vendrá dada, en el caso general de un movimiento en un espacio de  $d$  dimensiones, por

$$p_r(r, t)dr = \frac{1}{(4\pi d\mathcal{D}t)^{d/2}} \left\{ \begin{array}{ll} \frac{2^{\frac{d+1}{2}} \pi^{\frac{d-1}{2}}}{(d-2)!!} & d \text{ impar} \\ \frac{2\pi^{d/2}}{(\frac{1}{2}d-1)!} & d \text{ par} \end{array} \right\} \exp\left(-\frac{r^2}{4d\mathcal{D}t}\right) r^{d-1}dr$$

En particular, para  $d = 2$

$$p_r(r, t) = \frac{r}{4\mathcal{D}t} \exp\left(-\frac{r^2}{8\mathcal{D}t}\right)$$

y, para  $d = 3$

$$p_r(r, t) = \frac{\pi}{2} \frac{r^2}{(3\pi\mathcal{D}t)^{3/2}} \exp\left(-\frac{r^2}{12\mathcal{D}t}\right)$$

Lo interesante de esta distribución es que, para  $d > 1$ , y para un tiempo  $t$ , existe una distancia al origen<sup>5</sup> a la que es más probable hallar la partícula browniana, mientras que en el caso 1-D la probabilidad era máxima sólo en las cercanías de dicho origen.

**Ejercicio 12.2.** El objetivo de este ejercicio es generalizar los dos programas del apartado anterior para visualizar una instancia de paseo aleatorio en 2-D (o 3-D) y comprobar la expresión anterior de la distribución de probabilidad.

Para ello, se introducirá una estructura de datos que represente la partícula browniana, definida como:

```
struct Particula {
    int x, y;
};
```

Las coordenadas (obsérvese que son enteras) de esta partícula browniana serán modificadas por una función de agitación térmica, que simulará las colisiones con las moléculas del fluido: desplazará, cada vez que sea llamada, a la partícula a derecha o a izquierda, y arriba o abajo, en función de sendos “lanzamientos de moneda”. Para los lanzamientos de moneda usaremos, como antes, la función “double uniforme()” de nuestra “libprobabilidad.h”. Esta función de agitación térmica se declarará como “void agita(struct Particula \*p)” y modificará la posición de la partícula pasada por referencia.

Representar con “Gnuplot” la trayectoria de un caminante aleatorio bidimensional tras  $10^6$  pasos (o más).

**Ejercicio 12.3.** Para comprobar la correcta distribución de las partículas brownianas en 2-D alrededor del origen de su movimiento, se calculará el histograma de su distribución tras  $N$  pasos de tiempo (un paso de tiempo corresponde a una llamada a la función “void agita()”).

El histograma se calculará para un conjunto de caminantes aleatorios, que se guardarán en un “array” de estructuras “Particula”. Cada una de ellas se “agitara”  $N$  veces. Posteriormente, se calcularán sus distancias al origen y se guardarán en un “array” de valores reales (conviene usar para ello la función “double hypot(double, double)” declarada en <math.h>).

<sup>5</sup>Esta distancia es  $r_{\text{máx}} = \sqrt{2d(d-1)\mathcal{D}t}$ . ¡Calcúlese!

El programa calculará después el histograma de estas distancias. La comprobación se llevará a cabo representando el histograma junto con la correspondiente función de distribución de probabilidad, que será en este caso (en función de  $N$ )

$$p_r(r) = \frac{1}{N} r \exp\left(-\frac{r^2}{2N}\right)$$

(justifíquese esta expresión).

**Nota.** Obsérvese que, si el histograma abarca el intervalo  $[0, r_{\text{máx}}]$ , y tiene un número de cajas  $n_{\text{histo}}$ , entonces la función de densidad de probabilidad vendrá dada por

$$p_r(r_i) = \frac{h_i}{N_{\text{caminantes}} \times \delta r}$$

donde  $h_i$  es el número de caminantes en la  $i$ -ésima caja, situada alrededor de  $r_i$ ,  $N_{\text{caminantes}}$  el número total de caminantes en la estadística, y  $\delta r = r_{\text{máx}}/n_{\text{histo}}$  el tamaño de las cajas del histograma.

## 12.3. Integración Monte Carlo

Hasta ahora se han utilizado los generadores de números aleatorios para simular procesos de naturaleza estocástica o probabilista. Sin embargo, en muchas áreas de la física se emplean estos métodos como métodos para aproximar numéricamente soluciones de problemas no aleatorios, pero de muy difícil solución. Este es el ejemplo del cálculo de volúmenes o hipervolúmenes o, en general, integrales múltiples con un elevado grado de dimensionalidad. Por ejemplo, una integral múltiple de la forma

$$\int_{x_a}^{x_b} dx \int_{y_a}^{y_b} dy \int_{z_a}^{z_b} dz \int_{u_a}^{u_b} du \int_{v_a}^{v_b} dv \int_{w_a}^{w_b} dw f(x, y, z, u, v, w)$$

es muy costosa de evaluar. Aunque parezca excesivamente complicada, este tipo de integrales múltiples (y en más variables) son frecuentes en teoría de campos, tanto clásicos como cuánticos, y en física estadística (funciones de correlación).

Recordando el significado geométrico de la integración, que no es más que el de la suma de valores dados por una función en los puntos del dominio de integración, se puede llevar a cabo un “muestreo” de puntos distribuidos en este dominio y, simplemente, sumar los valores de la función  $f(x, y, z, u, v, w)$  en ellos, ponderando la suma con la “concentración” de los puntos elegidos en el dominio. Como corolario al *teorema de los grandes números*,<sup>6</sup> el resultado de esta suma converge, según crece el tamaño de la muestra, al valor de la integral.

### 12.3.1. Cálculo de volúmenes

El ejemplo más sencillo de cálculo de integrales por Monte Carlo es el de la determinación de un área, volumen o, en general, un hipervolumen, de una región de

<sup>6</sup>El teorema de los grandes números para variables aleatorias continuas dice, básicamente, que la esperanza de una función de estas variables en el dominio de variación de las mismas, es aproximada por el valor medio de los valores de dicha función muestreados aleatoriamente, y la aproximación es mejor según el número de valores tomados crece.

un espacio  $\mathbb{R}^d$  determinada por ciertas condiciones. Como ejemplo, veremos el de la determinación del volumen de una esfera de radio unidad. Esta esfera de radio unidad, centrada en el origen de coordenadas  $(0, 0, 0)$ , se halla inscrita en el cubo  $[-1, 1] \times [-1, 1] \times [-1, 1]$ .

La idea tras el método Monte Carlo es sencilla<sup>7</sup>. La probabilidad de que un punto, elegido al azar, caiga dentro de la esfera, viene dada por  $V_{\text{esfera}}/V_{\text{cubo}}$ , por lo tanto, si elegimos  $N$  puntos al azar dentro del cubo, es esperable que  $N V_{\text{esfera}}/V_{\text{cubo}}$  caigan dentro de la esfera. El volumen de la esfera, si se observan  $N_{\text{esfera}}$  puntos en su interior, será entonces  $V_{\text{esfera}} = V_{\text{cubo}} N_{\text{esfera}}/N$ .

El programa listado en 12.2 calcula el volumen de la esfera de radio  $R = 1$ . La pertenencia de un punto a esta esfera se determina mediante la función “int dentro(double x, double y, double z)”, que retorna un resultado igual a 0 (FALSO en C) en caso de que el punto no esté dentro y un valor distinto de cero (VERDADERO en C) en caso de que el punto esté dentro de ella.

Al programa se le indica por línea de comando el número de muestras aleatorias que deberá tomar para el cálculo. El volumen se calcula como el producto del “volumen” del cubo dentro del cual se toman los puntos por el cociente entre las “cuentas” de puntos dentro de la esfera respecto del total de muestras tomadas (“nmuestras”).

---

**Listado 12.2:** Programa que calcula por un método de Monte Carlo el volumen de una esfera.

---

```

1 #include <stdio.h>
2 #include <stdlib.h>
3 #include "libprobabilidad.h"
4
5 int dentro(double x, double y, double z)
6 {
7     double r2;
8
9     r2=x*x+y*y+z*z;
10
11     return (r2<1.0);
12 }
13
14 #define X_MIN -1.0
15 #define X_MAX +1.0
16
17 #define Y_MIN -1.0
18 #define Y_MAX +1.0
19
20 #define Z_MIN -1.0
21 #define Z_MAX +1.0
22
23 int main(int argc, char** argv)
24 {
25     long int nmuestras, n;
```

---

<sup>7</sup>Ya la vimos en el ejemplo para estimar el valor de  $\pi$  en la página ??.

```

26     long int cuentas;
27     double rx, ry, rz;
28     double volumen, fraccion, integral;
29
30     if( argc<2 )
31     {
32         printf("Uso %s <no. muestras>\n",
33             argv[0]);
34         exit(0);
35     }
36
37     nmuestras=atol(argv[1]);
38     if( nmuestras<=0 )
39     {
40         nmuestras=10000;
41     }
42
43     volumen= (X_MAX-X_MIN)
44             *(Y_MAX-Y_MIN)
45             *(Z_MAX-Z_MIN);
46
47     cuentas=0;
48     for(n=0; n<nmuestras; n++)
49     {
50         rx=X_MIN+(X_MAX-X_MIN)*uniforme();
51         ry=Y_MIN+(Y_MAX-Y_MIN)*uniforme();
52         rz=Z_MIN+(Z_MAX-Z_MIN)*uniforme();
53
54         if( dentro(rx,ry,rz) )
55         {
56             cuentas++;
57         }
58     }
59
60     fraccion=(double)cuentas/nmuestras;
61
62     integral=fraccion*volumen;
63
64     printf("Volumen estimado de la esfera = %g\n",
65         integral);
66
67     return 0;
68 }

```

---

Este método Monte Carlo se llama de muestreo por rechazo, porque sólo considera los puntos que cumplen la restricción

$$x^2 + y^2 + z^2 < R^2$$

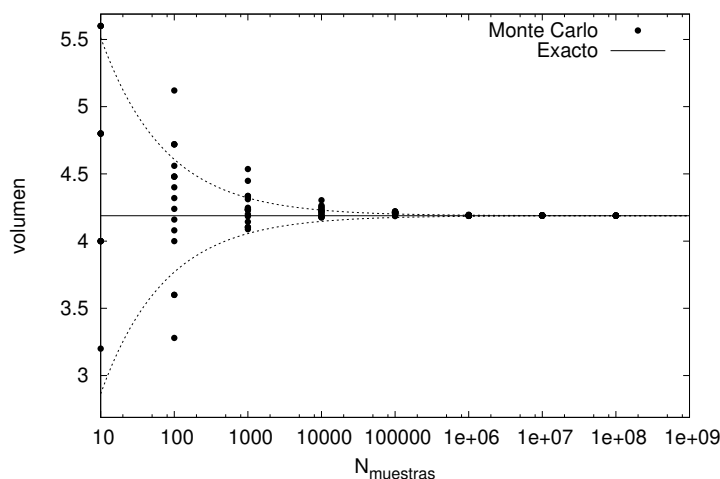


Figura 12.3: Resultados del cálculo del volumen de una esfera,  $V_{N_{\text{muestras}}}$ , en función del número  $N_{\text{muestras}}$  de muestras Monte Carlo. Los puntos corresponden a diferentes semillas del generador de números aleatorios. Se muestran también (con líneas de trazos) el corredor de errores teóricos correspondientes a una desviación típica de la estimación del volumen (el valor exacto del volumen es  $V = 4,1888$ ).

y rechaza los demás (un número realmente grande, la mitad de ellos prácticamente). No es un método muy efectivo desde el punto de vista numérico, pero ilustra de forma sencilla la idea detrás de este tipo de métodos aleatorios.

La figura 12.3 muestra el comportamiento del método en función del número de muestras empleadas, y lo comparac con el volumen exacto de la esfera.

**Nota:** Es posible demostrar que el error (la desviación típica) que afecta al método cuando se usan  $N_{\text{muestras}}$  muestras viene dado por

$$\Delta V = \frac{V}{\sqrt{N_{\text{muestras}}}}$$

Ello quiere decir que para reducir el error a la décima parte, habrá que emplear cien veces más puntos.

**Ejercicio 12.4.** Calcular el volumen de un cilindro de radio unidad y altura unidad, y representar la figura equivalente a la figura 12.3.

### 12.3.2. Cálculo de momentos de inercia de una esfera

El cálculo de una integral de volumen por Monte Carlo no se diferencia mucho del cálculo del propio volumen: basta remplazar el conteo de los puntos en el dominio considerado por la suma de los valores de la función en dichos puntos.

**Ejercicio 12.5.** Modificar el programa anterior para el cálculo del volumen de una esfera de modo que se calcule su momento de inercia respecto al eje Z. Recuerdese que este momento de inercia se calcula como

$$I_z = \int dV \rho(x, y, z) (x^2 + y^2)$$



donde  $\rho(x, y, z)$  es la densidad local en torno al punto  $(x, y, z)$  [**Nota:** aquí se asumirá densidad uniforme  $\rho = 1$ .]

Calcular el momento de inercia para diferentes valores de  $N_{\text{muestras}}$ , desde 10 hasta  $10^6$  (incrementando, cada vez, en un factor 10). Para cada valor de  $N_{\text{muestras}}$  utilizar 10 semillas distintas del generador de números aleatorios.

Comparar el resultado obtenido con el valor exacto para el momento de inercia de una esfera de masa  $M$  respecto a un eje que pasa por su centro de masas:

$$I_z = \frac{2}{5}MR^2$$

o, en términos de su densidad  $\rho$ :

$$I_z = \frac{8}{15}\pi\rho R^5$$

Representar el error relativo cometido respecto a este valor exacto en función del número de muestras y comentar el tipo de función obtenida.

**Nota:** Úsele una escala doble logarítmica. Para ello, en Gnuplot, introdúzcase la instrucción “set logscale xy” antes del comando “plot”.