

Introducción a la Práctica de MC

Alejandro Algora

Dolores Jordan





Que es Monte Carlo



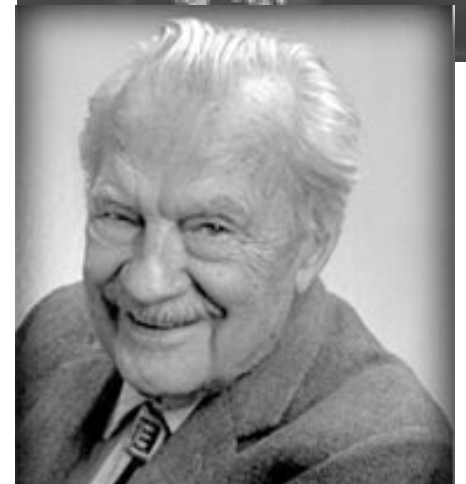
Nombre genérico de una serie de algoritmos computacionales que se basan en el uso de números aleatorios (sampleado repetido para obtener el resultado). Son métodos intrínsecamente estadísticos.



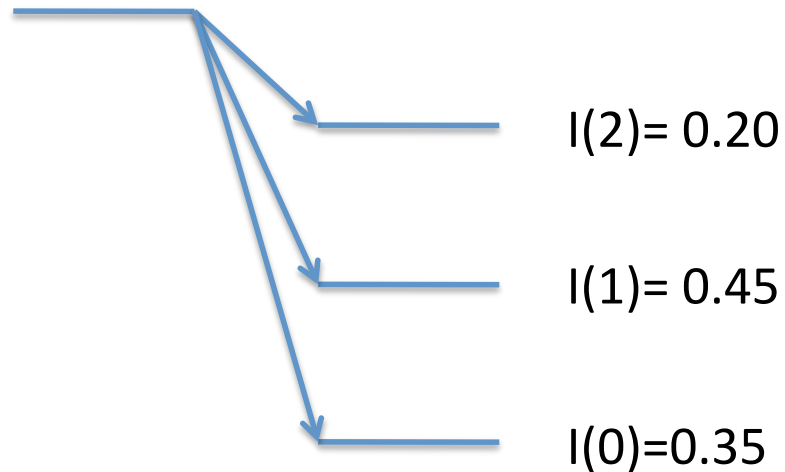
Tiene aplicaciones múltiples en simulaciones de sistemas físicos y matemáticos. Especialmente en sistemas complejos.



Denominación proviene de John von Neumann, Stanislaw Ulam and Nicholas Metropolis (1940) cuando estos trabajaban en el programa Manhattan (el tío de Ulam perdía su dinero frecuentemente en el Casino de Monte Carlo)



Para que sirve el Monte Carlo: un ejemplo



Resultados del "Programa"

$r = 0.745$

$total = 0$

$flag = 0$

1. Iteración ($i=0$)

$total = total + 0.35 = 0.35$, $flag = 0$,
nada ocurre

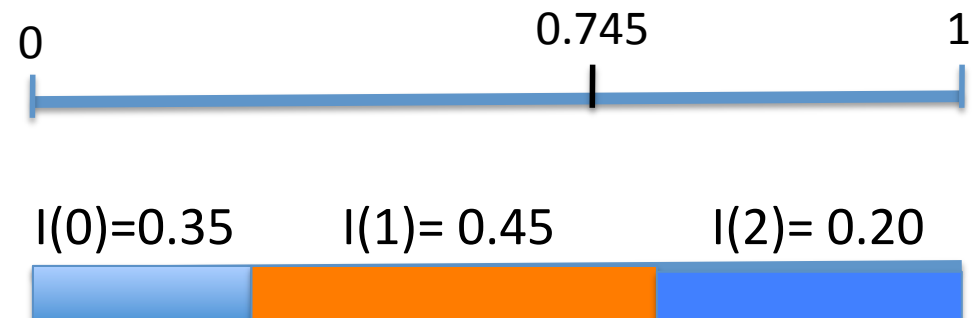
2. Iteración ($i=1$)

$total = total + 0.45 = 0.80$, $flag = 1$,
SELECTED !!!

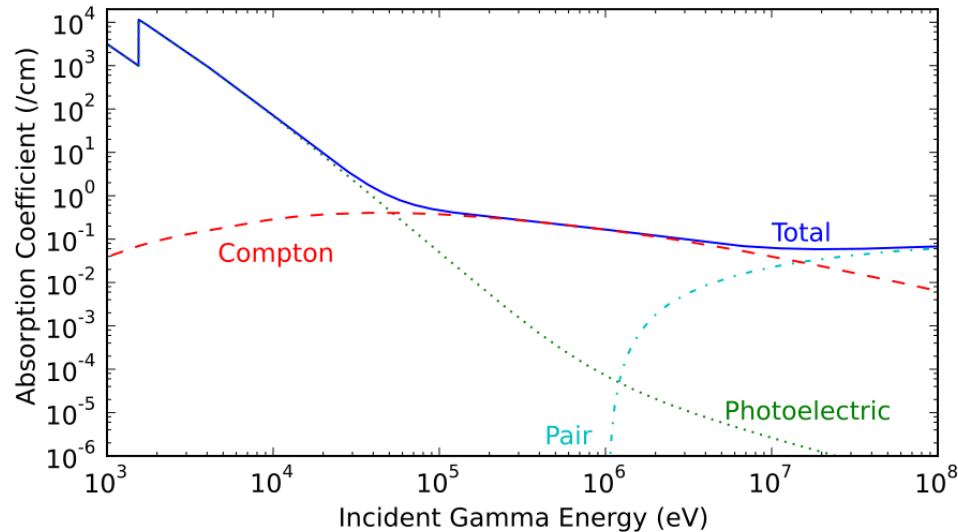
3. Iteración ($i=2$)

$total = total + 0.20 = 1.0$, $flag = 1$,
nada ocurre

```
r = G4UniformRand();
total=0.;
flag=0;
for (G4int i=0; i<3; i++) {
    total=total+I[i];
    If (r<total && flag==0) {
        Selected_level= i;
        flag ++;
    }
}
```



Para que sirve el Monte Carlo: otro ejemplo



$$I = I_0 e^{-\mu x}$$

$$\mu = n\sigma$$

Tres procesos

- Fotoeléctrico
- Compton
- Producción de Pares

Que es GEANT ?

- **GE**neration **ANd** **T**racking (finales del 1970, GEANT1)
- **GE**ometry **ANd** **T**racking (hoy en día)

Es un código para la simulación del paso de partículas a través de la materia, que se usa comúnmente en Física de las Altas Energías y en la Física Nuclear. También en los últimos años ha encontrado múltiples aplicaciones en la Física Medica, Dosimetría, Física del Espacio.

TOOLKIT: usando GEANT, los usuarios deben desarrollar su código, versiones actuales están en C++

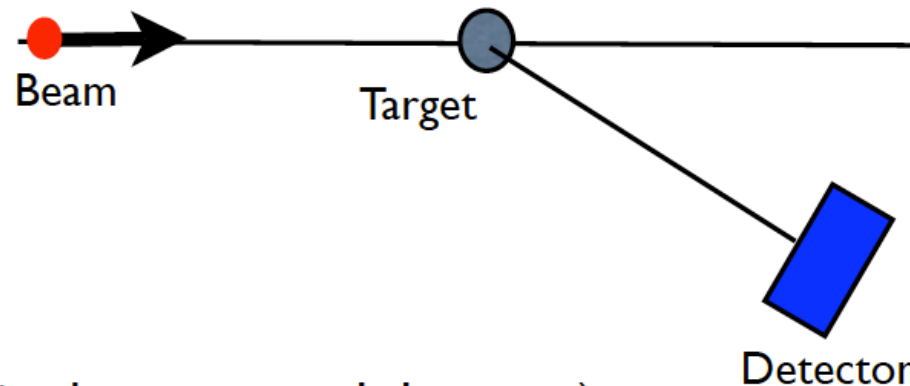
GEANT4 Collaboration <http://www.cern.ch/geant4>

Porque es necesario utilizar el MC

- Optimización de detectores (coste, eficiencia)
- Análisis de experimentos complejos, estudio de modelos físicos

Experimento Básico

- Haz
- Blanco
- Detectores



Para entenderlo se necesita tener un modelo de todos los ingredientes, incluyendo la física, campos, interacciones, etc.

MINIMO que se debe hacer cuando se usa un código de simulación:

- Definir los materiales y la geometría (blanco, detectores, etc)
- Definir las partículas y la física de sus interacciones
- Definir un evento primario y generarlo, y obtener la información relevante

Estructura del Programa (clases mas importantes)

- Descripción de la geometría

G4VUserDetectorConstruction

- Descripción de los procesos físicos

G4VUserPhysicsList

- Generación de eventos

G4UserRunAction

- Tracking (seguimiento) de las partículas (primarias y secundarias)

G4UserSteppingAction

- Recopilación de la Información

G4UserEventAction

RUN → EVENT → TRACK → STEP

RUN → EVENT → TRACK → STEP

RUN

Collección de EVENTOS que comparten el mismo detector y condiciones físicas. Durante el RUN no es posible cambiar la geometría, ni la física.

EVENT

Contiene la información de las partículas primarias (posición, momento, energía). Estos primarios se generan, y se crea un “stack” que contiene toda la información del evento. Este stack se vacía al final.

TRACK

La historia de las partículas. Contiene la información de la partícula que se transporta.

STEP

Cada paso contiene la información relevante al transporte de la partícula (energía depositada en el paso, tiempo de vuelo del paso, etc.). La longitud del paso se determina de una manera dinámica en función del rango de la partícula en el medio.

Explicaciones varias

- All user action classes, methods of which are invoked during “Beam On”, must be constructed in the user’s *main()* and must be set to the RunManager.
- **G4UserRunAction**
 - `BeginOfRunAction(const G4Run*)`
 - Define histograms
 - `EndOfRunAction(const G4Run*)`
 - Store histograms
- **G4UserEventAction**
 - `BeginOfEventAction(const G4Event*)`
 - Event selection
 - Define histograms
 - `EndOfEventAction(const G4Event*)`
 - Analyze the event
- **G4UserStackingAction**
 - `PrepareNewEvent()`
 - Reset priority control
- `ClassifyNewTrack(const G4Track*)`
 - Invoked every time a new track is pushed
 - Classify a new track -- priority control
 - Urgent, Waiting, PostponeToNextEvent, Kill
- `NewStage()`
 - Invoked when the Urgent stack becomes empty
 - Change the classification criteria
 - Event filtering (Event abortion)
- **G4UserTrackingAction**
 - `PreUserTrackingAction(const G4Track*)`
 - Decide trajectory should be stored or not
 - Create user-defined trajectory
 - `PostUserTrackingAction(const G4Track*)`
- **G4UserSteppingAction**
 - `UserSteppingAction(const G4Step*)`
 - Kill / suspend / postpone the track
 - Draw the step (for a track not to be stored by a trajectory)

Cuestiones Prácticas

```
[Escuela2012@pcgamma21 work]$  
[Escuela2012@pcgamma21 work]$  
[Escuela2012@pcgamma21 work]$ ls -l  
total 696  
-rw-r--r-- 1 Escuela2012 Escuela2012 28 Jan 21 10:36 dawn_visualizer.com  
-rw-r--r-- 1 Escuela2012 Escuela2012 28 Jan 18 18:24 dawn_visualizer.com~  
-rw-rw-r-- 1 Escuela2012 Escuela2012 33 Jan 21 09:58 freewrl_visualizer.com  
-rw-r--r-- 1 Escuela2012 Escuela2012 6444 Jan 18 18:24 geant4.9.6_enviroment.sh  
-rw-r--r-- 1 Escuela2012 Escuela2012 6444 Jan 18 18:24 geant4make.sh  
-rw-r--r-- 1 Escuela2012 Escuela2012 115 Jan 18 18:24 root_enviroment.com  
drwxr-xr-x 6 Escuela2012 Escuela2012 4096 Jan 21 10:20 test  
-rw-r--r-- 1 Escuela2012 Escuela2012 675840 Jan 21 11:31 test_electromagnetic.tar  
[Escuela2012@pcgamma21 work]$  
[Escuela2012@pcgamma21 work]$ █  
[Escuela2012@pcgamma12 work]$ cd test  
[Escuela2012@pcgamma12 test]$ ls -ls  
total 144  
 4 drwxr-xr-x. 3 Escuela2012 exp 4096 Jan 17 16:22 bin  
 4 -rw-r--r--. 1 Escuela2012 exp 678 Jan 16 11:31 GNUmakefile  
 4 -rw-r--r--. 1 Escuela2012 exp 1276 Jan 17 10:28 hplot.cpp  
 4 -rw-r--r--. 1 Escuela2012 exp 1276 Apr 26 2012 hplot.cpp~  
 4 drwxr-xr-x. 2 Escuela2012 exp 4096 Jan 17 10:43 include  
 4 -rw-r--r--. 1 Escuela2012 exp 3082 Jan 17 10:22 old_example.mac  
 4 -rw-r--r--. 1 Escuela2012 exp 358 Feb 1 2006 print-track.mac  
 4 -rw-r--r--. 1 Escuela2012 exp 246 Feb 1 2006 spec-eff.mac  
 4 drwxr-xr-x. 2 Escuela2012 exp 4096 Jan 17 10:42 src  
 4 -rw-r--r--. 1 Escuela2012 exp 3631 Jan 16 14:53 test.cc  
 4 -rw-r--r--. 1 Escuela2012 exp 3481 Apr 26 2012 test.cc~  
 4 -rw-r--r--. 1 Escuela2012 exp 53 Jan 17 16:25 test.eff  
84 -rw-r--r--. 1 Escuela2012 exp 85002 Jan 17 16:25 test.vec  
 4 drwxr-xr-x. 3 Escuela2012 exp 4096 Jan 17 16:22 tmp  
 4 -rw-r--r--. 1 Escuela2012 exp 1085 Jan 17 16:23 vis.mac  
 4 -rw-r--r--. 1 Escuela2012 exp 1031 Dec 14 20:38 vis.mac~  
[Escuela2012@pcgamma12 test]$ █
```

Estructura del Programa

Programa principal *test.cc*

Directorios relevantes: *src, include, bin, tmp*

```
[Escuela2012@pcgamma12 test]$ cd src
[Escuela2012@pcgamma12 src]$ ls -ls
total 132
12 -rw-r--r--. 1 Escuela2012 exp 9653 Jan 17 10:02 testDetectorConstruction.cc
12 -rw-r--r--. 1 Escuela2012 exp 9653 Jan 17 10:07 testDetectorConstruction.cc.Ge
 8 -rw-r--r--. 1 Escuela2012 exp 6000 Jan 17 10:31 testDetectorConstruction.cc.He3
 4 -rw-r--r--. 1 Escuela2012 exp 1404 Jan 17 10:30 testEventAction.cc
 8 -rw-r--r--. 1 Escuela2012 exp 6170 Jan 17 10:36 testPhysicsList.cc
20 -rw-r--r--. 1 Escuela2012 exp 19957 May  1 2012 testPhysicsList.cc~
 8 -rw-r--r--. 1 Escuela2012 exp 6170 Jan 17 10:36 testPhysicsList.cc.electromagneticSimple
20 -rw-r--r--. 1 Escuela2012 exp 20328 Jan 16 11:32 testPhysicsList.cc.networking
20 -rw-r--r--. 1 Escuela2012 exp 19847 May  1 2012 testPhysicsList.cc.old
 4 -rw-r--r--. 1 Escuela2012 exp 2841 Jan 17 10:34 testPrimaryGeneratorAction.cc
 4 -rw-r--r--. 1 Escuela2012 exp 2750 Jan 31 2006 testPrimaryGeneratorAction.cc~
 4 -rw-r--r--. 1 Escuela2012 exp 2163 Jan 17 10:38 testRunAction.cc
 4 -rw-r--r--. 1 Escuela2012 exp 1327 Jan 17 10:40 testSteppingAction.cc
 4 -rw-r--r--. 1 Escuela2012 exp 3324 Jan 31 2006 testVisManager.cc.notused
[Escuela2012@pcgamma12 src]$
```

