

## Simulaciones de Monte Carlo – PENELOPE

### 0. Instrucciones de conexión y acceso

Para el acceso al clúster hay que recordar su dirección, el puerto, el usuario y la clave. El usuario es el que cada uno recibió durante las prácticas (a101, a102, etc.; cada alumno tiene una clave). Desde la aplicación `ssh` se conecta a la máquina:

```
host: fiswulf.fis.ucm.es
user: alXX (el que sea)
password: la que se proporciona en clase
puerto (importante): 22
```

\*Si obtenéis el mensaje de error: "Unable to negotiate with 147.96.27.247 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss", la solución es añadir la opción: "-o HostKeyAlgorithms=+ssh-dss"

Recordad que hay que tener un cliente de `x11` instalado y encendido (`Xming` por ejemplo para Windows, `XQuartz` en MacOS), o hacer la conexión desde un Linux en modo gráfico. Y habilitar los túneles en `ssh` (activar la opción `X11`).

Una vez se ha terminado el *login*, hay que ir al directorio del master:

```
cd Master_24-25
```

y al directorio de esta práctica de simulación:

```
cd practica_MonteCarlo
```

Dentro de esa carpeta tenemos dos subcarpetas principales:

```
geo
mat
```

Para generar las tablas de materiales, hay que bajar al directorio `mat`:

```
cd mat
```

Para volver a los directorios superiores:

```
cd ..
```

→ Para editar algunos ficheros (input de `penmain` por ejemplo, para cambiar los ficheros de materiales) se puede usar el editor que uno prefiera: `gedit`, `vi`, `emacs`, ...

→ Si en algún momento se daña algún fichero y no se sabe volver atrás, se pueden volver a copiar los ficheros originales de las simulaciones. Se encuentran en el directorio `/home/Master_24-25` y se pueden copiar con el comando `cp` con la secuencia siguiente:

Una vez hecho el *login*, ir al directorio raíz del usuario:

```
cd
```

Y desde ahí entramos en la carpeta `Master_24-25`:

```
cd Master_24-25
```

Y ahí hacemos:

```
cp -rav /home/Master_24-25/practica_MonteCarlo .
```

Ojo al '.'

## Simulaciones de Monte Carlo – PENELOPE

### 1. Introducción

**Penmain** es un programa que usa paquetes de **PENELOPE**<sup>1</sup> para realizar simulaciones Monte Carlo de propagación de diferentes tipos de partículas (fotones, electrones y positrones) a lo largo de medios materiales. La principal diferencia de este programa con otros similares es que este se centra en regiones del espacio concretas, definidas por el usuario previamente, que se denominan cuerpos (*body*). Por esta razón Penmain es muy útil para el estudio del comportamiento de detectores frente a diferentes tipos e intensidades de radiación.

Toda la información que se pretende obtener de cada simulación debe de estar descrita de forma correcta en el input sobre el cual trabajará el programa. Para que el programa empiece a funcionar es necesario definir con anterioridad la forma y el espectro de emisión de la fuente del haz; la geometría del espacio sobre el que se propagará el haz, y los materiales de los que estará compuesta la geometría.

### 2. Input

El input del programa es un **archivo con extensión .in** donde se define la fuente, la geometría, los materiales de los que está compuesta y la información que deseamos obtener de cada detector.

- a) **Fuente:** la fuente se define directamente en el input del programa. Podemos definir tanto la posición como la forma de la fuente. La posición se define mediante las coordenadas x,y,z donde la queramos colocar tras el comando SPOSIT. Podemos dotar a la fuente de forma de caja mediante el comando SBOX seguido de las dimensiones que la queramos dar. También podemos elegir el tipo de partículas que se genere así como la energía que estas tengan. Además, podemos definir la forma cónica del haz que emite la fuente. Para ello usamos el comando SCONE seguido de los ángulos de dirección ( $\varphi$ ,  $\theta$ ) y del ángulo de apertura del cono ( $\alpha$ ).

```
.
>>>>>>> Source definition.
SKPAR 2          [Primary particles: 1=electron, 2=photon, 3=positron]
SPECTR 1e6 1.0e0 [E bin: lower-end and total probability]
SPECTR 1e6 -1.0e0 [E bin: lower-end and total probability]
SPOSIT 0 0 0      [Coordinates of the source]
SCONE 0 0 0       [Conical beam; angles in deg]
.
```

- b) **Material:** los materiales de los que están hechos los detectores que se quieren estudiar deben de estar definidos en archivos externos con extensión **.mat**. Los archivos de los materiales que usaremos están ya creados en la carpeta **mat/**. Para denotar un material en concreto deberemos escribir la siguiente línea en el input.

```
>>>>>>> Material data and simulation parameters.
MFNAME mat/BGO.mat [Material file, up to 20 chars]
MSIMPA 1.0e5 1.0e3 1.0e5 0.1 0.1 1e4 1e3 [EABS(1:3),C1,C2,WCC,WCR]
```

- c) **Geometría:** la geometría no se escribe directamente en el input si no que se define en un archivo a parte de extensión **.geo**. Escribir una geometría directamente en un archivo **.geo** es complicado y por eso el GFN-UCM tiene un programa que se encarga de transformar un

<sup>1</sup> A Code System for Monte-Carlo Simulation of Electron and Photon Transport

archivo .in de fácil comprensión y escritura en otro .geo. El archivo .geo que usaremos en la práctica es la de un cubo de 1 cm x 1 cm x 1 cm cuyo centro está situado en (0,0,1). Este archivo se encuentra en la carpeta geo/. El material del que estará formado este cubo será el elegido como material en **b)**).

```
.
>>>>>>> Geometry definition file.
GEOMFN geo/cubol.geo [Geometry file, up to 20 chars]
.
```

**d) Elección de Outputs:** este programa crea algunos ficheros por defecto que dan información general sobre cómo y qué ha ocurrido durante la simulación (penmain.dat). Pero los outputs interesantes de este programa son aquellos que deben de ser definidos en el input y son principalmente los 2 siguientes:

- **Impact detectors:** este output consiste en el espectro inicial de energías de las partículas que luego han interactuado con el detector, es decir, lo que se representa en este espectro es la energía inicial de las partículas que interactúan con el detector, sin importar cuánta energía depositen o mediante que efecto lo hagan. Para obtener este output debemos de escribir el comando **IMPDET** seguido de la ventana de energía ( $E_o, E_f$ ), el número de bins en que se divide el espectro, un 1 o un 0 según se quiera crear o no espacio de fase, y por último, un 0, 1 o 2 según si se quiere que las partículas se paren al llegar al material, si se quiere que lo atraviesen o si se quiere que lo atraviesen y además se calcule la fluencia. Además, hace falta decir que detector queremos usar, en este caso como sólo hay uno bastará con poner en la siguiente línea **IDBODY 1**. El archivo que obtenemos por defecto tendrá el nombre de “spc-impdet-01.dat”.
- **Energy deposition:** este output es el espectro de la energía depositada en el detector. Para obtenerlo tenemos que escribir en el input el comando **ENDETC** seguido de la ventana de energía ( $E_o, E_f$ ) y el número de bins en que se divide el espectro. De igual forma que para el anterior debemos definir que detector es, en nuestro caso solo hay uno tendremos que poner **EDBODY 1**. El archivo que obtenemos por defecto tendrá el nombre de “spc-enddet-01.dat”.

```
.
>>>>>>> Impact detectors (up to 25 different detectors).
IPSF=0; no psf is created.
IPSF=1; the psf is created.
IDCUT=0; tracking is discontinued at the detector entrance.
IDCUT=1; the detector does not affect the tracking.
IDCUT=2; the detector does not affect tracking, the energy
          distribution of particle fluence (averaged over the
          volume of the detector) is calculated.
IMPDET 0.0 1.2e6 160 0 2 [E-window, no. of bins, IPSF, IDCUT]
IDBODY 1 [Active body; one line for each body]
.
>>>>>>> Energy deposition detectors (up to 25).
ENDETC 1e3 1.2e6 100 [Energy window and number of bins]
EDBODY 1 [Active body; one line for each body]
.
```

**e) Número de eventos y tiempo máximo de trabajo:** para definir el número de historias que se generan hay que utilizar el comando **NSIMSH** seguido del número de eventos que se quieran simular. Para el tiempo máximo de trabajo se suele usar un número muy grande para que no interrumpa la simulación. En nuestro caso 2E9 segundos.

```
.
>>>>>>> Job properties
NSIMSH 1e5 [Desired number of simulated showers]
TIME 2e9 [Allotted simulation time, in sec]
END [Ends the reading of input data]
.
```

### 3. Definición de la geometría

Escribir una geometría directamente en un archivo **.geo** es complicado y por ello el grupo tiene un programa que se encarga de transformar un archivo **.in** de fácil comprensión y escritura en otro **.geo**. Este programa se encuentra en el directorio **geo/** y su nombre es **obj-geo**. El comando que se usa para convertir un fichero **.in** en uno **.geo** mediante este programa es

**`./obj-geo detectores.in nombre.geo.`**

En la imagen se ve un archivo **.in** típico en el que se define una geometría para posteriormente pasarla a un archivo de extensión **.geo**. En la imagen se aprecian diferentes líneas con una serie de letras y números. Cada línea define uno de los cuerpos que forman el sistema. La primera letra se corresponde con el tipo de sistema de coordenadas (rectangular **``R``**, cilíndricas **``C``** o esféricas **``S``**); el primer número de la línea hace referencia a la etiqueta numérica para referirse a este cuerpo en el input; los siguientes tres números son el centro del cuerpo en coordenadas (x, y, z) y los tres siguientes son dependientes del tipo de coordenadas: son los lados del rectángulo en caso de coordenadas cartesianas; el radio interior, exterior y altura del cilindro en coordenadas cilíndricas; y el radio interior y exterior de la esfera en coordenadas esféricas (el tercer número en este caso no tiene efecto alguno).

```
!TYPE MATERIAL X_CENTER Y_CENTER Z_CENTER R1 R2 HEIGHT[cm] PH_INC TH_INC[DEG]
R 1 0.0 0.0 4.0 2.0 2.0 2.0 0 0
R 2 0.0 4.0 0.0 2.0 2.0 2.0 0 0
R 3 4.0 0.0 0.0 2.0 2.0 2.0 0 0
!WRITE ONE OBJECT PER LINE. FILE END WITH A STARTING CHARACTER NOT VALID FOR
TYPE
```

En el ejemplo anterior se han creado tres cubos de 2 cm de lado centrados a 4 cm del origen de coordenadas, cada uno en uno de los ejes principales.

Por otra parte, el resto del espacio que no se define con ningún cuerpo el programa lo tratará por defecto como si hubiese sido definido como vacío.

### 4. Ejecución del programa

Para ejecutar el programa desde la terminal hay que introducir el siguiente comando

**`./penmain.exe < input.in`**

### 5. Gnuplot

**Gnuplot** es un programa gráfico de software libre. Usaremos este programa para dibujar los outputs ya que tenemos scripts preparados para cada uno de los diferentes outputs que podemos obtener.

Tenemos dos ficheros de salida principales:

➔ **spc-impdet-01.dat**: para visualizarlo, estando en el mismo directorio ambos archivos, ejecutamos el comando:

**`gnuplot spc-impdet.gnu`**

➔ **spc-endsdet-01.dat**: ejecutamos

**`gnuplot spc-endsdet.gnu`**

para representar este fichero gráficamente.

Para visualizar los ficheros \*.ps o \*.pdf generados se puede usar evince:

**`evince impdet.pdf`**

## 6. Informe

Para el informe de la práctica hay que hacer, como mínimo, lo siguiente:

1. Crear un archivo **.mat** para uno de los siguientes materiales:

- a)  $\text{CeBr}_3$
- b) NaI
- c) BGO
- d) CeGAG
- e) Pr:LuAg
- f) Agua
- g) Algún Plástico

Para crear este archivo **.mat** hay que introducir el comando

**`./material`**

desde el directorio **./mat** (es decir, **cd mat** desde el directorio practica\_MonteCarlo). Nos pedirá cierta información, que será la siguiente:

- ➔ Primero hay que **elegir la opción 1** para crear un material nuevo desde el teclado del ordenador.
- ➔ A continuación, nos pedirá introducir el nombre del material. Tras esto elegimos crearlo a raíz de la fórmula estequiométrica (**opción 1**).
- ➔ Lo siguiente es introducir el **número atómico y el coeficiente estequiométrico** de cada uno de los materiales.
- ➔ El próximo paso nos permite modificar las energías de ionización, pero lo dejamos como está eligiendo la **opción 2**.
- ➔ A continuación, introducimos la **densidad del material**.
- ➔ Para terminar solo queda introducir el nombre del archivo con su extensión por ejemplo **“nombre.mat”**.

2. Crear la tabla del material generado en el apartado anterior y representar las secciones eficaces en una gráfica como se hizo en clase. Comentar los resultados. Para crear las tablas hay que escribir el siguiente comando en el directorio **.mat**:

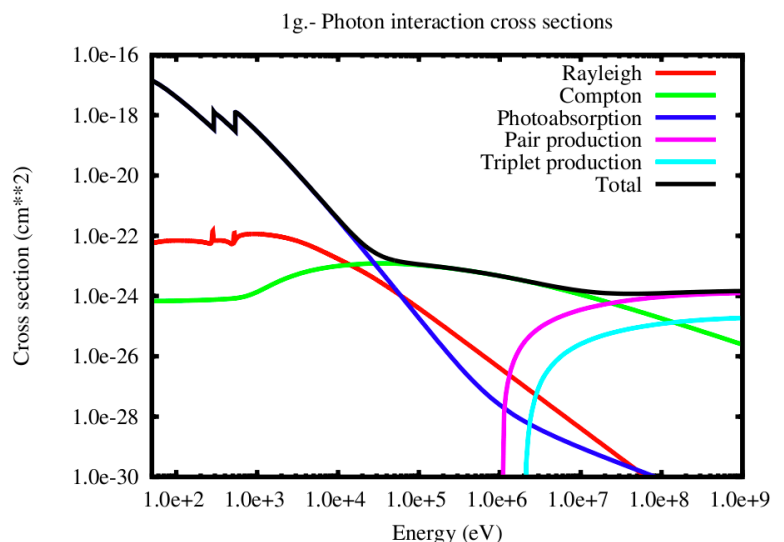
**`./tables`**

Al ejecutarlo preguntará por el nombre del fichero de materiales (el **.mat**) que queramos procesar.

3. Ejecutar comando:

**`gnuplot tables.gnu`**

que generará figuras en un fichero PDF (**tables.pdf**) con los coeficientes de atenuación másica de fotones para ese material como la mostrada a la derecha. Poner estas figuras de cada material en el informe. Es necesario indicar el material del que se trate en cada figura.



Comentar los contenidos de los ficheros que crea el programa **tables**:

eclass1.tab emfps.tab exs.tab gmfp.tab gxs.tab pmfprange.tab pstp.tab radyields.tab  
emfprange.tab estp.tab gmacs.tab graxsmac.tab pclass1.tab pmfps.tab pxs.tab

4. Realizar la simulación para el material BGO ya generado y para el material que generáis vosotros con diferentes energías y comparar los resultados de las simulaciones según el material y la energía. Comparar, por ejemplo, cuántos fotones interaccionan con el material por fotoeléctrico y por Compton. Realizar las figuras comparando los espectros de energía de los fotones detectados.

5. Para ello ejecutar los scripts siguientes:

**gnuplot spc-impdet.gnu** → espectros de partículas impactando al detector.

**gnuplot spc-enddet.gnu** → espectros de energía depositada en el detector.

Puede hacer falta teclear alguna tecla (**enter** por ejemplo) varias veces cuando la pantalla se quede congelada, hasta que termine el script y vuelva a aparecer el cursor.

Observar los ficheros **.pdf** que se han creado al correr este comando. Renombrarlos con el nombre del material correspondiente. Utilizar las figuras de estos PDFs para el informe.

6. Obtener la eficiencia intrínseca de fotopico de los dos materiales (BGO y el nuevo material elegido por vosotros) para radiación gamma a diferentes energías. La cantidad de estadística en el fotopico del espectro de energía depositada en el detector respecto al total es la eficiencia de fotopico buscada.

Hacer la simulación para los 2 materiales y, al menos, para 3 energías de fotones diferentes en el rango 1 keV – 10 MeV y obtener la eficiencia intrínseca para ambos materiales. Representar gráficamente los valores obtenidos para la eficiencia y discutir los resultados obtenidos. Recomendación: representar la eficiencia en %.