

Universidad del Valle de Guatemala
FACULTAD DE CIENCIAS Y HUMANIDADES
DEPARTAMENTO DE FÍSICA



**Simulación de la fuente radioactiva de Cs-137 del
Laboratorio Secundario de Calibración Dosimétrica
de Guatemala, para la medición de dosis, utilizando
GAMOS.**

**TRABAJO DE GRADUACIÓN PRESENTADO POR
MAX ANTONIO MÉNDEZ GÓMEZ
PARA OPTAR AL GRADO ACADÉMICO
DE LICENCIADO EN FÍSICA**

**GUATEMALA
2019**

**Simulación de la fuente radioactiva de Cs-137 del
Laboratorio Secundario de Calibración Dosimétrica
de Guatemala, para la medición de dosis, utilizando
GAMOS.**

Universidad del Valle de Guatemala
FACULTAD DE CIENCIAS Y HUMANIDADES
DEPARTAMENTO DE FÍSICA



**Simulación de la fuente radioactiva de Cs-137 del
Laboratorio Secundario de Calibración Dosimétrica
de Guatemala, para la medición de dosis, utilizando
GAMOS.**


**TRABAJO DE GRADUACIÓN PRESENTADO POR
MAX ANTONIO MÉNDEZ GÓMEZ
PARA OPTAR AL GRADO ACADÉMICO
DE LICENCIADO EN FÍSICA**

**GUATEMALA
2019**

Vo.Bo.:

(f) 
Lic. Edgar Andrés Monterroso

Tribunal Examinador:

(f) 
Lic. Edgar Andrés Monterroso

(f) 
MSc. Zaida del Rosario Urrutia del Cid

(f) 
Ing. Luis Mijangos

Guatemala 05 de diciembre de 2019

Prefacio

En los años de universidad que he cursado la carrera de física en la Universidad del Valle de Guatemala, me he dado cuenta que el papel aguanta con todo, la teoría suele ser elegante y bonita, pero verdadera belleza surge cuando es posible replicarla experimentalmente y comprobarla. Agradezco primeramente a Dios por la oportunidad de terminar mi carrera y por su cuidado y protección en todos estos años. Luego agradezco a la Universidad del Valle de Guatemala, por la abrirme las puertas y darme la oportunidad de superarme, conocer nuevas personas, obtener nuevos conocimientos y crecer integralmente como persona. A la Licda. Zaida Urrutia quien me apoyó todos estos años y siempre dió importancia a mis principios. A Licda. Olguita Castellanos quien ha sido un gran apoyo por parte del departamento, siempre me ha brindado atención especial sin importar la hora y nunca ha dejado de trabajar duro para facilitar tramites en la universidad.

Agradezco a mi asesor Lic. Edgar Andres Monterroso, porque a pesar de la distancia y tantas ocupaciones estuvo ahí para ayudarme a sacar este trabajo y el PDh Pedro Aguilar, quien también me brindó su apoyo y tiempo a pesar de no tener ningún compromiso con mi trabajo de graduación.

Agradezco a cada uno de los profesores que con mucha paciencia y dedicación estuvieron ahí para enseñarme lecciones importantes, en especial al Lic. José Carlos Chiquín, quien me enseñó a amar la ciencia y fomentar el deseo de seguir aprendiendo y descubriendo como funciona el mundo. A Lic. Eugenio Aristondo quien me enseñó que una teoría bien comprendida es más valiosa que mil ejercicios resueltos. A MSc. Irene Aguilar, quien me enseñó a ver más allá de lo que ven nuestros ojos y preguntarse porqué suceden las cosas, como suceden y para que suceden. A el Lic. Ronald Curtis, quien me enseñó que el orden es la clave de un buen resultado y a nunca pasar por alto los detalles. A Lic. Silvio Urizar, quien me enseñó que un profesor puede ser también un buen amigo.

Agradezco a mis padres: Rudy Méndez y Vianka Gómez, por su apoyo incondicional en todos estos años, el apoyo, los ánimos, la atención que me han dado y porque siempre han estado ahí cuando los necesito. A mi hermano David Méndez quien siempre ha estado ahí para apoyarme. A mi novia Angélica González quien ha sido un gran apoyo también, siempre está dandome ánimos y nunca me ha dejado en las malas.

Agradezco a mis amigos en especial a mi mejor amiga Lizy Pérez, por su apoyo y su amistad todo

este tiempo. A mis compañeros físicos que cursaron estos cuatro años junto a mi en la universidad, Andres Ixcoy, Paolo Marsicovetere, Pedro Valdez, Rodrigo Albizures y Marcos Gutierrez, por su amistad, soporte, compañía y buenos momentos que me han hecho pasar.

Índice de contenido

Lista de figuras	vii
Lista de cuadros	ix
Lista de símbolos	xi
Resumen	xiii
1 Introducción	1
2 Objetivos	3
3 Conceptos fundamentales	5
3.1 Radiación ionizante	5
3.2 Interacción de la radiación ionizante con la materia	6
3.3 Atenuación exponencial	7
3.3.1 Atenuación exponencial simple	7
3.3.2 Atenuación exponencial para modos plurales de absorción	8
3.4 Geometría de haz estrecho	9
3.5 Decaimiento radioactivo	11
3.5.1 Constantes de decaimiento parcial	11
3.5.2 Actividad	12
3.5.3 Mitad de vida radioactiva y vida media	12
3.6 Decaimiento beta	13
3.7 Interacción de los rayos γ y β con la materia	14
3.7.1 Efecto Compton	15
3.7.2 Efecto fotoeléctrico	17
3.7.3 Producción de pares	18
3.8 Física computacional	19
3.8.1 Técnica de Monte Carlo	20
4 GAMOS	21
4.1 Introducción a GAMOS	21
4.1.1 Estructura de GAMOS	21

4.1.2	Concepto plug-in	22
4.1.3	Geant4	22
4.1.4	Tecnicismos relacionados a la arquitectura de GAMOS	24
4.1.5	Interfaz gráfica de GAMOS	24
4.1.6	Interfaz gráfica de GAMOS en Linux (Ubuntu)	25
4.2	Geometría	27
4.2.1	Geometría desde un archivo de texto	27
4.2.2	Unidades	30
4.2.3	Árbol de volúmenes	30
4.3	Generador	31
4.3.1	Fuentes de partículas	31
4.3.2	Distribuciones	32
4.4	Acciones de usuario	35
4.4.1	Filtros	35
4.4.2	Contadores	38
4.5	Física electromagnética de GAMOS	40
4.5.1	Producción de cortes	41
4.6	Verbosidad	42
4.7	Optimización del CPU	43
4.7.1	Límites del usuario	43
4.7.2	Eliminando partículas	44
5	Simulación	45
5.1	Caracterización de la geometría	45
5.1.1	Materiales	45
5.1.2	Matriz de rotación	46
5.1.3	Volumen madre o primario	47
5.1.4	Paredes de concreto	47
5.1.5	Fantoma	48
5.1.6	Volumen de agua del fantoma y voxels	50
5.2	Archivo principal	51
5.2.1	Verbosidad	51
5.2.2	Geometría desde un archivo de texto	51
5.2.3	Comandos generales	52
5.2.4	Caracterización de la fuente	53
5.2.5	Árbol de volúmenes	55
5.2.6	Filtros y límites de usuario para optimizar CPU	55

5.2.7	Dosis en el fantoma	56
5.2.8	Visualización 3D	57
5.2.9	Cantidad de partículas a correr	58
6	Resultados	59
6.1	Distribuciones de dosis en el fantoma	59
6.2	Optimización del tiempo de cómputo	60
6.3	Dosis y energía depositada	61
6.4	Partículas irradiadas	61
6.5	Comparación con los datos de LSCD	62
7	Conclusiones	65
9	Anexos	69
9.1	Lista de materiales para producción de cortes	69
9.2	Lista de energía depositada por voxel en el fantoma dados por la herramienta <i>GmG4PSEnergyDeposit</i>	70
9.3	Datos de dosis dados por la herramienta <i>GmG4PSDoseDeposit</i>	70
9.4	Árbol de volúmenes	71
9.5	Arcihvo de entrada utilizado <i>Prueba.in</i>	72
9.6	Archivo de entrada con la geometría <i>ContruyendoGeometria.geom</i>	74
9.7	Cortona3D	75
9.8	Informe dosimetría personal extendido por LABORATORIOS TÉCNICOS APLI- CACIONES NUCLEARES	75

Lista de figuras

3.1	Atenuación exponencial simple.	8
3.2	Geometría de haz estrecho. El diámetro del haz de fotones o neutrones	10
3.3	Importancia relativa de los tres tipos de interacciones de rayos γ	15
3.4	Cinemática del efecto Compton. Un fotón con energía cuántica $h\nu$	16
3.5	Cinemática del efecto fotoeléctrico. Un fotón con energía cuántica $h\nu$	17
3.6	Producción de pares en un campo de fuerza de Coulomb de un núcleo atómico. . .	19
4.1	Estructura de las clases de Geant4.	23
4.2	Interfaz gráfica GAMOS	25
4.3	Interfaz gráfica GAMOS en Linux	26
5.1	Geometría de las paredes de concreto LSCD	48
5.2	Geometría de la caja del fantoma de acrílico	49
5.3	Geometría del fantoma con caja de acrílico, dividida por voxels de agua	51
5.4	Simulación de 3000 partículas generadas por el generador GAMOS	54
5.5	Comparación entre simulación con filtros y límites de usuario	56
6.1	Distribución de dosis en el fantoma en 1D	59
6.2	Distribución de dosis en el fantoma en 2D	59
6.3	Curva de optimización de tiempo de cómputo	60

Lista de tablas

6.1	Resultados de energía y dosis depositada	61
6.2	Comparación entre la dosis absorbida por el fantoma según el LSCD y la dosis absorbida por el fantoma según el simulador de GAMOS	62

Lista de símbolos

h	Constante de Planck
c	Velocidad de la luz en el vacío
λ	Longitud de onda
σ	Constante de decaimiento Radioactivo
τ	Vida Media
$\tau_{1/2}$	Mitad de vida
ν	Frecuencia

Resumen

El objetivo de este trabajo de graduación es realizar una simulación en el LSCD. Con esta simulación se busca replicar estos procesos con el fin de minimizar el tiempo necesario para obtener resultados. Se realizó la simulación utilizando la interfaz GAMOS (una interfaz que facilita el uso de Geant4 por medio de un lenguaje de script, que compila de forma rápida y sencilla las herramientas de Geant4) que es una interfaz de Geant4, una herramienta elaborada por el CIEMAT para la simulación de interacción de partículas con la materia basado en las técnicas de Monte Carlo. Al finalizar la simulación se espera obtener la dosis absorbida por un fantoma (espectro de losa de polimetilmetacrilato (PMMA) de tamaño 30 cm x 30 cm x 15 cm utilizado para calibrar dosímetros y simular el torso humano) a 3 metros de distancia de una fuente de radiación de Cs137, con la finalidad de poder hacer una aproximación computacional y comparar el valor del laboratorio con el valor simulado.

La primera parte busca introducir conceptos de radiación ionizante, formas no estocásticas de medición por medio de valores esperados para procesos de interacción con la materia, se describe el concepto de atenuación exponencial y se describe brevemente la geometría de haz estrecho. Además, se analiza el decaimiento radioactivo, en general, y se profundiza un poco en el decaimiento beta, por las características de la fuente que se está simulando. Por último, se describen los procesos de interacción de rayos γ y β con la materia y se describe acerca de la física computacional.

En la siguiente parte se detalla sobre algunas de las funcionalidades del programa GAMOS, se describen librerías útiles y funciones que pueden ser implementadas para crear geometrías, caracterizar fuentes y diferentes procesos de radiación. Se describen librerías útiles para poder optimizar el tiempo de cómputo y diferentes contadores que el programa ofrece para poder obtener información relevante de la simulación. Después se describe detalladamente la simulación, con las funciones implementadas, librerías utilizadas, técnicas de optimización y extracción de resultados.

En la última parte se presentan los resultados obtenidos, se dan gráficos de como se distribuye la dosis depositada en el fantoma, desde los tres ejes coordenados y se dan gráficos en dos dimensiones que ayudan a visualizar la distribución de dosis en el fantoma. Después se analiza los resultados de implementar técnicas de optimización en la simulación y como estas ayudan a obtener mejores tiempos al aumentar la cantidad de partículas simuladas. Para finalizar se explica la obtención de resultados, se compara con los datos del LSCD y se discuten dichos resultados.

1 Introducción

Desde la creación y la detonación de la primera bomba atómica, se marcó el comienzo de la “Era Atómica”, desde entonces nuestro entendimiento e implementación de radiación, material radiactivo y nuclear ha cambiado. Durante la Guerra Fría, hubo experimentación de ambos lados de las propiedades y usos del material nuclear en varios reactores de prueba, para obtener la ventaja estratégica ofensiva con el poder de material nuclear, para crear bombas nucleares y potenciales usos en otros campos como medicina, radiografía, etc.

La base de la industria en temas de radiación radica en la medición y control de la energía recibida en cualquier proceso. De ahí proviene la importancia de la dosimetría de la radiación, ya que ofrece una manera de replicar el proceso en el laboratorio, transferirlo a la práctica, llevarlo a la industria y establecer un control tecnológico. La dosimetría por radiación es la rama más joven de la ciencia de radiación ionizantes, con un gran interés de parte de la Comisión Internacional de Unidades y Mediciones Radiacionales (ICRU).

El departamento de Protección y Seguridad Radiológica, de la Dirección General de Energía de Guatemala, trabaja actualmente en el fortalecimiento del control Regulatorio por el pleno cumplimiento del Decreto 11-86, “Ley para el Control, Uso y Aplicación de Radioisótopos y Radiaciones Ionizantes” y sus Reglamentos. Los servicios de los Laboratorios de Aplicaciones Nucleares brindan, contribuyen a la promoción de las aplicaciones seguras y pacíficas de la Energía Nuclear del país.

GAMOS es un marco de trabajo (framework) basado en GEANT4, con un conjunto de herramientas para la simulación del paso de partículas a través de la materia, con áreas de aplicación en física de altas energías, nuclear y aceleradores lineales, así como estudios en ciencias médicas y espaciales, desarrollado por el CIEMAT. Usa tecnología “plug-in”, el cual está cuidadosamente diseñada en forma de módulos, que permite explotar con facilidad todas las funciones de GEANT4.

2 Objetivos

A. General

Desarrollar una simulación estadística que caracterice la fuente de radiación de **Cs-137** del Laboratorio Secundario de Calibración Dosimétrica (LSCD) utilizando el Método de Monte Carlo y el conjunto de herramientas que brinda GEANT4 en la plataforma GAMOS.

B. Específicos

1. Realizar una documentación completa del código, las características y variables para su futura reproducción experimental y análisis.
2. Obtener una distribución de dosis en varias dimensiones que pueda describir las zonas con mayor actividad en el fantoma y estimar la dosis total retenida en el fantoma.
3. Optimizar el tiempo de cómputo y reproducción del experimento para poder reproducir la simulación, cambiar variables, analizar datos y validarlos de forma rápida.
4. Comparar la dosis medida experimentalmente con la dosis obtenida por la simulación para obtener una evaluación de la utilidad de la simulación para el LSCD de Guatemala (SSDL-GUA).

3 Conceptos fundamentales

3.1 Radiación ionizante

Las radiaciones ionizantes están caracterizadas por su habilidad de excitar e ionizar átomos de la materia con la que interactúan. La energía necesaria para causar que un electrón escape de un átomo es del orden de $4 - 25eV$, por ende la radiación debe ser mayor a este valor de energía para poder recibir el nombre de “ionizante”. (Attix, 1986)

Se consideran los siguientes tipos de radiación ionizante:

I *Rayos γ* : Radiación electromagnética emitida por un núcleo o por el proceso de aniquilación entre materia y antimateria. La energía cuántica de cualquier fotón electromagnético está dada en keV como se puede observar en la ecuación (2.1). (Attix, 1986)

$$E_{\gamma} = h\nu = \frac{hc}{\lambda} = \frac{1.2398[keVnm]}{\lambda[nm]} \quad (3.1)$$

II *Rayos X*: radiación electromagnética emitida por partículas cargadas, usualmente electrones, al cambiar de nivel de energía, o al ser sometidas a un campo de fuerza de Coulomb para ser frenados. Estas tienen las mismas características que los *rayos γ* , la única diferencia es la forma en que son emitidas.(Attix, 1986)

III *Electrones rápidos*: conocidos como *rayos β* , que puede ser positivo o negativo, cuando estos son emitidos por un núcleo y como *rayos δ* cuando estos son el resultado de una colisión. Cuando estos son de carga positiva reciben el nombre de positrones. Con generadores de Van de Graaff es posible obtener un haz de electrones con energías arriba de $12MeV$. (Attix, 1986)

IV *Partículas pesadas cargadas*: usualmente obtenidas por un campo de fuerza de Coulomb en un generador de Van de Graaff, o por un acelerador de partículas pesadas. También conocidas como *Partículas α* cuando son emitidas por un núcleo radioactivo. (Attix, 1986)

V Neutrones: partículas de carga neutra, obtenidas por reacciones nucleares, ya que estas no pueden ser aceleradas mediante campos electro magnéticos.(Attix, 1986)

VI Neutrinos: son partículas elementales que no tienen carga eléctrica y prácticamente no tienen masa, además, tienen una sección transversal extremadamente pequeña para interactuar con la materia. Por esto, la energía transmitida o acarreada por los neutrinos es despreciada en el contexto de física radiológica y dosimetría. (Attix, 1986)

3.2 Interacción de la radiación ionizante con la materia

La radiación ionizante puede ser descrita de forma no estocástica en términos del valor esperado del momento, o de la energía de las partículas irradiadas. Las cantidades no estocásticas más utilizadas para describir la interacción de un campo de radiación con la materia, en términos de los valores esperados son: el Kerma (K), dosis absorbida (D), la exposición (X). (Attix, 1986)

I Kerma (K): Es el coeficiente entre dE_{tr} y dm . Donde dE_{tr} el promedio de las energías cinéticas iniciales de todas las partículas cargadas liberadas en un volumen con masa dm por partículas sin carga que inciden en dm . Las unidades son J/kg , esta unidad recibe el nombre especial de gray (Gy). (Oxford University Press, 2011)

$$K = \frac{dE_{tr}}{dm} \quad (3.2)$$

II Dosis absorbida (D): es el coeficiente entre $d\epsilon$ y dm . Donde $d\epsilon$ es la energía promedio impartida por radiación ionizante a la materia con masa dm . La unidad es Gy . (Oxford University Press, 2011)

$$D = \frac{d\epsilon}{dm} \quad (3.3)$$

III Exposición (X): es el coeficiente entre dq y dm . Donde dq es el valor absoluto de la media entre la carga de los iones producidos por todos los electrones y positrones liberados o creados por fotones incidentes, en un volumen con masa dm de aire seco. Esta cantidad se mide en C/kg . (Oxford University Press, 2011)

$$X = \frac{dq}{dm} \quad (3.4)$$

IV Cema (C): es el coeficiente entre dE_{el} y dm . Donde dE_{el} es la energía promedio perdida en interacciones electronicas en un volumen con masa dm de una material a causa de partículas cargadas incidentes, excluyendo partículas sin carga. Esta cantidad se mide en Gy . (Oxford University Press, 2011)

$$C = \frac{dE_{el}}{dm} \quad (3.5)$$

V Energía depositada (ϵ_i): Es la energía depositada en una interacción i . Está dada por la siguiente ecuación. (Oxford University Press, 2011)

$$\epsilon_i = \epsilon_{in} - \epsilon_{out} + Q \quad (3.6)$$

Donde ϵ_{in} es la energía de la partícula incidente, sin contar energía de reposo, ϵ_{out} es la suma

de las energías de todas partículas ionizantes cargadas y sin carga saliendo de la interacción, sin incluir energía de reposo, Q es la carga en la energía de reposo del núcleo y todas las partículas elementales involucradas en la interacción. La unidad en la que se mide esta cantidad es J . (Oxford University Press, 2011)

VI Energía lineal (y): es el coeficiente entre ϵ_s y l . Donde ϵ_s es la energía impartida a la materia en un volumen dado por un simple deposito de energía en un evento, l es la longitud media del volumen. Esta cantidad se mide en J/m . (Oxford University Press, 2011)

$$y = \frac{\epsilon_s}{l} \quad (3.7)$$

VII Energía específica (z): es el coeficiente entre ϵ y m . Donde ϵ es la energía impartida por radiación ionizante a la materia en un volumen con masa m . Esta cantidad se mide Gy .(Oxford University Press, 2011)

$$z = \frac{\epsilon}{m} \quad (3.8)$$

3.3 Atenuación exponencial

Este concepto es relevante cuando se trata de radiación ionizante sin carga, como fotones y neutrones, ya que estas partículas pierden su energía luego de pocas interacciones. Por su lado, las partículas cargadas pierden su energía gradualmente luego de varias colisiones pequeñas. Una partícula sin carga tiene una probabilidad significativa de pasar a través de una capa de materia sin perder energía, mientras que una partícula cargada, siempre pierde una parte o toda su energía. Una partícula sin carga no tiene un rango límite para el que no pueda a travesar la materia, en cambio, las partículas cargadas tienen un rango límite para el cual se quedan sin energía cinética. (Attix, 1986)

3.3.1. Atenuación exponencial simple

Considere un haz mono energético compuesto por un número grande de partículas sin carga N_0 , que incide perpendicularmente sobre un plano de materia de largo L . Por simplicidad, se asume el caso ideal en el que cada partícula es completamente absorbida con una simple interacción, sin producir radiación secundaria, o pasa directamente a través del plano sin cambios en su energía o dirección, como se puede observar en la Figura (2.1). (Attix, 1986)

Sea μ la probabilidad de que una partícula individual interactúe en una unidad de espesor de material atravesado. Entonces la probabilidad de que la partícula interactúe en un grosor infinitesimal dl es μdl . Si N partículas están incidiendo sobre dl , el cambio dN en el número N

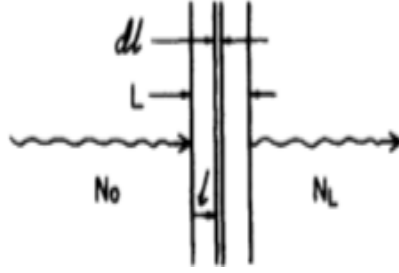


Figura 3.1: Atenuación exponencial simple.

Fuente: Attix (1986)

debido a la absorción está dado por la ecuación (2.9). (Attix, 1986)

$$dN = -\mu N dl \quad (3.9)$$

Donde μ es típicamente dado en unidades de cm^{-1} o m^{-1} , y dl en cm o m . La razón de cambio de N debido a la absorción de partículas en dl es simplemente:

$$\frac{dN}{N} = -\mu dl \quad (3.10)$$

Integrando sobre el ancho l de 0 a L , de igual forma sobre las poblaciones de partículas de N_0 a N_L se obtiene:

$$\frac{N_L}{N_0} = e^{-\mu L} \quad (3.11)$$

A esta expresión se le conoce como la ley de atenuación exponencial, la cual aplica para el caso ideal que considera absorción simple y no considera dispersión de partículas o radiación secundaria. Se puede tomar en cuenta el caso en el que hay dispersión y se producen partículas secundarias, pero no están contadas en N_L . La cantidad μ es conocida como coeficiente de atenuación lineal o coeficiente de atenuación. (Attix, 1986)

3.3.2. Atenuación exponencial para modos plurales de absorción

Suponga que hay presentes más de un proceso de absorción en el caso anterior. Entonces se puede escribir el coeficiente de atenuación μ como la suma de sus partes:

$$\mu = \mu_1 + \mu_2 + \mu_3 + \dots \quad (3.12)$$

o

$$1 = \frac{\mu_1}{\mu} + \frac{\mu_2}{\mu} + \frac{\mu_3}{\mu} + \dots \quad (3.13)$$

Donde μ_1 es el coeficiente de atenuación lineal parcial del proceso 1, y así sucesivamente para el resto de procesos. De esta manera para un proceso x, se obtiene sustituyendo en la ecuación (2.14).

$$\frac{N_x}{N_0} = e^{-(\mu_1 + \mu_2 + \mu_3 + \dots)L} \quad (3.14)$$

En esta ecuación se puede observar que el número N_L de partículas entrando a través de la superficie del material de largo L, depende del efecto de todos los coeficientes de atenuación exponencial. El número total de interacciones por todos los tipos de procesos está dado por:

$$\Delta N = (N_0 - N_L) = N_0(1 - e^{-\mu L}) \quad (3.15)$$

Y el número de interacciones de un proceso x.

$$\Delta N_x = (N_0 - N_L) \frac{\mu_x}{\mu} = N_0(1 - e^{-\mu L}) \frac{\mu_x}{\mu} \quad (3.16)$$

Donde μ_x/μ es la fracción de las interacciones por proceso x. La relación puede quedar escrita como se expresa en la siguiente ecuación.

$$\frac{N_x}{N_0} = (1 - e^{-\mu L}) \frac{\mu_x}{\mu} \quad (3.17)$$

(Attix, 1986)

3.4 Geometría de haz estrecho

El haz de fotones o neutrones interactúa con la materia mediante procesos que generan radiación secundaria tanto con carga, como sin carga. El número total de partículas que salen de la superficie de la Figura (2.2), es mucho mayor que las partículas que sufren dispersión dentro de la superficie, por lo que es importante determinar que será incluido en N_L . Es importante distinguir las partículas secundarias cargadas de las no cargadas, las partículas cargadas tienden a ser absorbidas por el atenuador y son excluidas. Las partículas secundarias no cargadas, pueden o no ser tomadas en cuenta en N_L , si son tomadas la ec. (2.11) se vuelve inválida, ya que viola la suposición de que solo eventos simples de absorción pueden ocurrir. Si partículas secundarias llegan al detector, pero solo se consideraron las primarias en N_L , se dice que este fenómeno tiene geometría de haz ancho, pero atenuación de haz estrecho. Como consecuencia la ec. (2.11) sigue siendo inválida bajo estas condiciones.

El valor del coeficiente de atenuación μ contiene todos los coeficientes parciales de interacciones provocadas por partículas primarias, siempre que una partícula deje de ser primaria por cualquier tipo de interacción μ siempre será numéricamente mayor que el valor de cualquier coeficiente

de atenuación efectivo μ' que es observado bajo condiciones de atenuación de haz ancho. Si gradualmente se reduce la fracción de radiación secundaria medida, la atenuación de haz ancho se aproxima a la atenuación de haz estrecho, y μ' se aproxima a μ . (Attix, 1986)

Hay dos métodos para lograr la atenuación de haz estrecho:

- I *Discriminación* hacia cualquier tipo de partículas dispersas y secundarias que alcancen el detector, en función de la energía de las partículas, habilidad de penetración, dirección, coincidencia, anti coincidencia, etc.
- II *Geometría de haz estrecho*, la cual impide que cualquier partícula dispersa o secundaria alcance el detector.

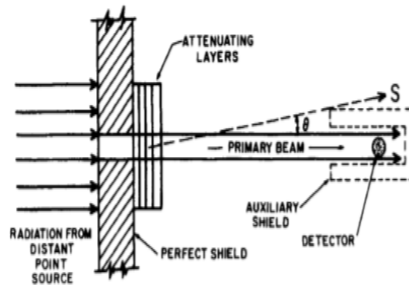


Figura 3.2: Geometría de haz estrecho. El diámetro del haz de fotones o neutrones es lo suficientemente largo para cubrir el detector de manera uniforme.

El detector está colocado a una distancia lo suficientemente larga del atenuador para que las partículas dispersas o secundarias (S) que llegan al detector sean despreciables en comparación a las primarias.

Fuente: Attix (1986)

En la Figura (2.2) se puede observar las características principales de la geometría de haz estrecho. El detector es posicionado lo suficientemente lejos de las capas de atenuación de modo que ninguna partícula S que es desviada por una interacción, alcance el detector. El haz es colimado para ser lo suficientemente largo para cubrir el detector uniformemente, de modo que minimice el número de partículas dispersadas o secundarias generadas por el atenuador. La fuente de radiación está localizada a una gran distancia del atenuador de modo que las partículas incidentes sean perpendiculares al mismo. Además, la intensidad del haz primario en el detector será casi independiente de la distancia al atenuador, siempre que la intensidad de las partículas dispersas y secundarias decrezcan de forma de cuadrado inverso a la distancia. Experimentalmente no es difícil lograr una geometría de haz estrecho, de modo que sea posible aproximar la atenuación de haz estrecho. (Attix, 1986)

3.5 Decaimiento radioactivo

El término radioactivo se refiere al fenómeno asociado a transformaciones espontáneas que implican cambios en el núcleo de los átomos o en los estados de energía de los átomos. La energía liberada en estas transformaciones es emitida como partículas nucleares (partículas alfa, electrones y positrones) y/o fotones. Estas transformaciones representan procesos estocásticos. Todo el átomo está involucrado en este proceso, ya que las transformaciones nucleares también afectan la estructura del átomo y causan emisión o captura de electrones, fotones o ambos. (Oxford University Press, 2011)

Considere un número N de átomos radiactivos idénticos, se define σ como la constante de decaimiento radioactivo total, y es el coeficiente entre $-dN/N$ y dt , donde dN/N es el cambio fraccional promedio del núcleo en ese estado de energía debido a transformaciones nucleares espontáneas en el intervalo de tiempo dt . (Oxford University Press, 2011) La dimensional de esta constante es (s^{-1}) . Asumimos que el producto entre σ por un tiempo consistente es la probabilidad de que un átomo decaiga en este intervalo de tiempo. (Attix, 1986)

El cambio de N de un grupo, que no se repone por una fuente de más núcleos, en el tiempo es igual a la actividad, como se puede observar en la ecuación (2.18).

$$\frac{-dN/N}{dt} = \sigma \quad (3.18)$$

Separando las variables e integrando de $t = 0$, cuando $N = N_o$, a un tiempo t , se obtiene:

$$\frac{N}{N_o} = e^{-\sigma t} \quad (3.19)$$

3.5.1. Constantes de decaimiento parcial

Si el núcleo tiene más de un posible modo de desintegración, la constante total de decaimiento se puede escribir como la suma de las constantes de decaimiento parcial σ_i :

$$\sigma = \sigma_A + \sigma_B + \dots \quad (3.20)$$

La actividad total está dada por $N\sigma$:

$$N\sigma = N\sigma_A + N\sigma_B + \dots \quad (3.21)$$

La actividad parcial de un grupo de N núcleos respecto a i -ésimo modo de desintegración puede ser escrito como la ecuación (2.22).

$$\sigma N = \sigma_i N_o e^{-\sigma t} \quad (3.22)$$

donde N está expresado en términos de N_o de acuerdo con la ecuación (2.22). La actividad, originalmente se midió en curie (CI), que están definidas en desintegraciones por segundo que ocurren en una masa de $1g$ de $^{226}_{88}\text{Ra}$. Recientemente el cuerpo internacional se estándares estableció una nueva unidad especial para la actividad, el *becquerel* (Bq), el cual es igual a $(1s^{-1})$. (Attix, 1986)

3.5.2. Actividad

Un nucleido es una especie de átomo con un número específico de protones y neutrones en su núcleo. Un nucleido inestable que se transforma a un nucleido estable o progenie inestable, es llamado radionucleido. La transformación resulta en otro nucleido o en una transición a un estado de energía más bajo del mismo nucleido. (Oxford University Press, 2011)

La actividad (A), de un grupo de radionucleidos en un estado particular de energía, en un determinado tiempo, es el coeficiente entre $-dN$ y dt . Donde dN es el cambio promedio del número de núcleos en dicho estado de energía debido a transformaciones nucleares espontáneas en el tiempo dt . La dimensional de esta cantidad es (s^{-1}) , pero recibe el nombre especial de *becquerel* (Bq). (Oxford University Press, 2011)

$$A = \frac{-dN}{dt} \quad (3.23)$$

3.5.3. Mitad de vida radioactiva y vida media

La mitad de vida radioactiva, $t_{1/2}$, está definida como el tiempo para que la mitad de núcleos iniciales N_o decaiga. Se puede encontrar mediante el coeficiente entre $N(t)/N_o = 1/2$. Como se puede observar en la ecuación (2.24). (Dewji and Hertel, 2019)

$$t_{1/2} = \frac{0.693}{\sigma} \quad (3.24)$$

La vida media τ , es el tiempo promedio que un núcleo sobrevivirá antes de decaer. Si el número de núcleos que decaen en un intervalo t y $t + dt$ es $|dN/dt|dt$, entonces la vida media se puede obtener de la siguiente ecuación. (Dewji and Hertel, 2019)

$$\tau = \frac{\int_0^{\infty} t |dN/dt| dt}{\int_0^{\infty} |dN/dt| dt} \quad (3.25)$$

donde vemos que se reduce a la inversa de la constante de decaimiento.

$$\tau = \frac{1}{\sigma} \quad (3.26)$$

La vida media de las partículas asume que las partículas están en reposo. Para partículas que están viajando a una fracción significativa de la velocidad de la luz (c), los efectos de la relatividad

sobre la dilatación de tiempo y la contracción de la longitud deben ser considerados. Estos efectos se toman en cuenta en el caso de aceleradores de partículas o estudios cosmológicos. (Dewji and Hertel, 2019)

3.6 Decaimiento beta

Decaimiento beta es un término utilizado para describir tres procesos: emisión de electrones, emisión de positrones o captura de electrones. El decaimiento beta implica la transición de un neutrón en un par protón – electrón, y resulta en el incremento en una unidad del número Z. (Dewji and Hertel, 2019)



Si esta ecuación estuviese completa, el electrón sería emitido de manera monoenergética. Pero, en realidad el electrón emitido tiene una distribución de energía desde cero al máximo de energía predicha por la masa del electrón. Este fenómeno dio lugar a la teoría que una segunda partícula, el neutrino, era emitida simultáneamente del núcleo del átomo y compartía la energía cinética. Luego esta teoría se revisó con el antineutrino, $\bar{\nu}$, que al igual que el neutrino, no tiene carga, con $\text{spin} = 1/2$ y masa muy pequeña. El antineutrino comparte una porción de su energía cinética con el electrón, la fracción de energía del electrón y antineutrino variará con cada decaimiento, la energía del electrón observado tendrá la forma del un espectro. Tomando en cuenta esto se reescribe la ecuación (1.28). (Dewji and Hertel, 2019)



En el caso del decaimiento de positrones, para un núcleo inestable como resultado de tener algunos neutrones, los protones ser convertirán en un neutrón y positrón mediante el proceso de decaimiento del positrón. El positrón es idéntico al electrón, excepto por su carga positiva. El proceso de decaimiento también involucra un neutrino. (Dewji and Hertel, 2019)



El núcleo resultante tendrá un protón más, para mantener la paridad de carga, se emitirá un electrón.

Para el caso de la captura de electrones, luego de el decaimiento de positrones, la masa del átomo debería de ser mayor que la masa inicial del decaimiento para que esto sea posible energéticamente, pero no es el caso. El núcleo es neutrón deficiente, la captura de un electrón orbital permite un

proceso de decaimiento alternativo. (Dewji and Hertel, 2019)



Este proceso es llamado captura de electrón y resultado en la transformación de un protón en un neutrón. Por eso, el número Z decrece en una unidad. El resultado es el mismo que en decaimiento de positrón, lo que indica que es un proceso competitivo para altas razones de Z/A y es un modo de decaimiento más común. (Dewji and Hertel, 2019)

Un núcleo que tiene exceso de neutrones, tiende a emitir un electron (partícula β^{-}), esto dejando el núcleo con un neutrón menos y un protón más, entonces el número atómico Z incrementa en 1. A la inversa, un núcleo con exceso de protones, usualmente emite un positrón (β^{+}), en efeto disminuyendo Z en 1 mientras aumenta el número de neutrones en 1. En ambos casos el número total del núcleo (protones + neutrones) permanece constante. La emisión de rayos - β deja un gran número de núcleos de varios tipos en estado excitado, y uno o más rayos - γ son emitidos para poder alcanzar el estado base. (Attix, 1986)

3.7 Interacción de los rayos γ y β con la materia

Existen cinco tipos de interacciones de los rayos γ y β con la materia que son considerados en física radiológica:

- a) Efecto Compton
- b) Efecto fotoeléctrico
- c) Producción de pares
- d) Dispersión de Rayleigh
- e) Interacciones fotonucleares

Las tres primeras son las más importantes, estas resultan en la transferencia de energía a electrones, la cual imparten a la materia mediante interacciones con la fuerza de Coulomb a lo largo de sus trayectorias. La dispersión de Rayleigh es elástica; los fotones son redireccionados mediante un ángulo pequeño sin pérdida de energía. Las interacciones fotonucleares significantes para fotones con energías por encima de MeV, donde podrían crear problemas de radio-protección mediante la producción de neutrones y por consiguiente radio activación. La importancia relativa del efecto Compton, efecto fotoeléctrico y producción de pares, depende de la energía cuántica del fotón ($E_{\gamma} = h\nu$) y el número atómico Z del medio absorbente. Se puede observar en la Figura (2.3), las regiones de Z y E_{γ} en las cuales predomina la interacción. (Attix, 1986)

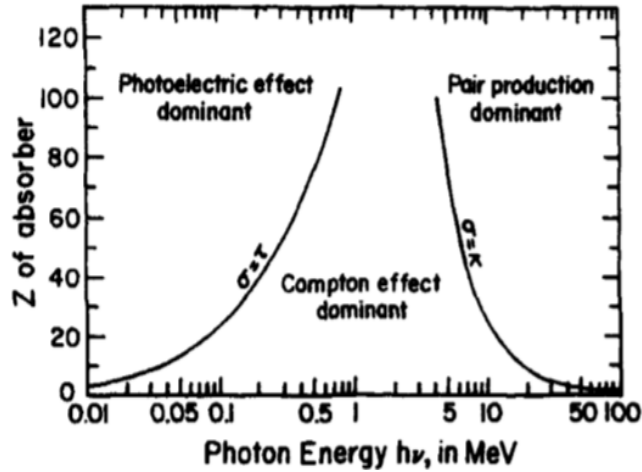


Figura 3.3: Importancia relativa de los tres tipos de interacciones de rayos γ . Las curvas muestran para que valores de Z y E_γ los efectos son predominantes.

Fuente: Attix (1986)

El efecto fotoeléctrico es dominante cuanto el fotón tiene bajas energías, el efecto Compton tiene efecto en medianas energías, y la producción de pares toma lugar en altas energías. Para medios con bajo Z , como el carbón, aire, agua, tejido humano, la región del efecto Compton toma dominio, esta región se encuentra entre $\approx 20\text{KeV}$ hasta $\approx 30\text{MeV}$. (Attix, 1986)

3.7.1.Efecto Compton

La descripción del efecto Compton puede ser subdividida convenientemente en dos aspectos: cinemático y la sección transversal. El primero de ellos relaciona energías y ángulos de las partículas involucradas y el segundo predice la probabilidad de que ocurra el efecto Compton en una interacción. En ambas se asume que el electrón golpeado por el fotón incidente está desenlazado y estacionario. (Attix, 1986)

3.7.1.1.Cinemática del efecto Compton

En la Figura (2.4), se puede observar un fotón con energía $h\nu$ colisionando con un electrón. El fotón incide con un momento $h\nu/c$, donde c es la velocidad de la luz en el vacío. El electrón estacionario no tiene energía cinética inicial o momento. Luego de la colisión el electrón parte con un ángulo θ , con energía cinética T y momento p . El fotón se dispersa con un ángulo ϕ con una nueva energía cuántica $h\nu'$ y momento $h\nu'/c$. La conservación de la energía requiere que se cumpla la siguiente ecuación. (Attix, 1986)

$$T = h\nu - h\nu' \quad (3.31)$$

La conservación del momento a lo largo de la dirección inicial del fotón (0°) puede ser expresado

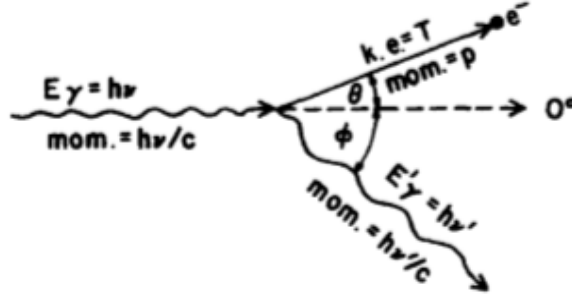


Figura 3.4: Cinemática del efecto Compton. Un fotón con energía cuántica $h\nu$ incide por la izquierda y choca contra un electrón estacionario sin enlace, dispersándolo con ángulo θ relativo a la dirección incidente del fotón, con energía cinética T . El fotón dispersado con energía resultante $h\nu'$ se desvía un ángulo ϕ en la dirección opuesta al electrón en el plano dispersado. La energía y momento se conservan. La suposición de que el electrón está desenlazado se refiere a que la relación cinemática es independiente del número atómico del medio.

Fuente: Attix (1986)

como:

$$h\nu = h\nu' * \cos(\phi) + pc * \cos(\theta) \quad (3.32)$$

La conservación del momento perpendicular a la dirección de incidencia del fotón da la ecuación (2.33).

$$h\nu' * \sin(\phi) = pc * \sin(\theta) \quad (3.33)$$

3.7.1.2. Interacción con la sección transversal

1. Dispersión de Thomson: J. J. Thomson provee una descripción teórica del proceso por el cual un fotón de rayo γ puede ser dispersado por un electrón. En esta teoría se asume que el electrón es libre de oscilar bajo la influencia de un campo eléctrico de una onda electromagnética clásica incidente, luego se remite puntualmente un fotón con la misma energía. El electrón no retiene energía cinética como resultado de este evento de dispersión elástico. Este resultado concuerda con la predicción cinemática que más tarde el tratamiento relativista de Compton que se cumple hasta cerca de energías $h\nu = 0.01 Mev$. Thomson dedujo la sección transversal diferencial por electrón para un fotón dispersado en un ángulo ϕ , por unidad de ángulo sólido, de la siguiente forma:

$$\frac{d_e \sigma_o}{d\Omega_\phi} = \frac{r_o^2}{2} (1 + \cos^2 \phi) \quad (3.34)$$

Donde $r_o = e^2/m_o c^2$ es llamado radio del electrón clásico.

La sección transversal total σ_o puede ser obtenida al integral la ecuación anterior alrededor de todas las direcciones de dispersión. Esta sección transversal es numéricamente igual a la probabilidad de que un evento de dispersión de Thomson ocurra cuando un fotón pase mediante

una capa que contiene un electrón por unidad de centímetro cuadrado. (Attix, 1986)

2. Sección Transversal de Klein – Nishina: En 1928 Klein y Nishina aplicaron la teoría de la relatividad de Dirac, al efecto Compton para obtener secciones transversales. El tratamiento de Klein – Nishina es remarcablemente exitoso para predecir el valor experimental correcto, incluso mediante la suposición de que los electrones no tienen enlaces y están inicialmente en reposo. La sección transversal diferencial de un fotón dispersado un ángulo ϕ , por unidad de ángulo sólido y por electrón, correspondiente a la ecuación (2.34) de la teoría de Thomson, puede ser descrita como:

$$\frac{d_e\sigma_o}{d\Omega_{phi}} = \frac{r_o^2}{2} \left(\frac{h\nu'}{h\nu}\right)^2 \left(\frac{h\nu}{h\nu'} + \frac{h\nu'}{h\nu} \sin^2\phi\right) \quad (3.35)$$

Donde r_o es el radio del electrón clásico, $h\nu'$ es la energía cuántica del fotón dispersado y $h\nu$ es la energía cuántica del fotón incidente. Para bajas energías esta ecuación es igual a la ecuación (2.34), verificando que la sección transversal $K - N$ se reduce a la de Thomson para el caso especial de fotones de bajas energías. (Attix, 1986)

3.7.2.Efecto fotoeléctrico

El efecto fotoeléctrico es la interacción más importante para fotones de bajas energías con la materia. Mientras que la interacción de la sección transversal en el efecto Compton se aproxima a un valor constante y su energía transferida disminuye cuando $h\nu$ decrece a valores por debajo de $0.5MeV$, la sección transversal correspondiente al efecto fotoeléctrico aumenta fuertemente, especialmente para los medios en los que Z tiene valores altos. En consecuencia, el efecto fotoeléctrico predomina totalmente sobre el efecto Compton con fotones de bajas energías, particularmente cuando se refiere a la energía transferida a electrones secundarios.(Attix, 1986)

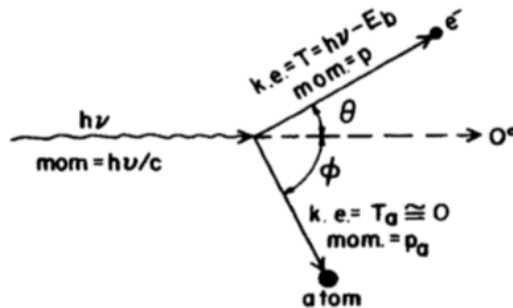


Figura 3.5: Cinemática del efecto fotoeléctrico. Un fotón con energía cuántica $h\nu$ incide por la izquierda de la Figura y choca contra un electrón enlazado a un átomo con energía E_b . El fotón se desvanece, dando su energía cinética de $T = h\nu - E_b$ al electrón, el cual departe con un ángulo θ relativo a la dirección del fotón incidente. Luego de la colisión el átomo departe con un ángulo ϕ .

Fuente: Attix (1986)

3.7.2.1. Cinemática

En el caso del efecto Compton se observó que un fotón no puede dar toda su energía al chocar con un electrón libre. Sin embargo, puede hacerlo al encontrarse con un electrón que esté unido estrechamente con el núcleo del átomo, especialmente en átomos con alto número atómico. A este fenómeno se le denomina efecto Fotoeléctrico. Como se puede observar en la Figura (2.5), un fotón incidente con energía cuántica $h\nu$ interactúa con un electrón atómico con energía potencial E_b . El efecto fotoeléctrico no puede tomar lugar respecto al electrón dado a menos que $h\nu > E_b$. El fotón es totalmente absorbido en la interacción y deja de existir. La energía cinética dada al electrón, independiente del ángulo de dispersión θ , esta dada por la siguiente ecuación.

$$T = h\nu - E_b \quad (3.36)$$

El electrón parte luego de la interacción con un ángulo θ relativo a la dirección del fotón incidente, con un momento p . Siempre que el fotón sea totalmente absorbido, este no provee un fotón disperso en congruencia con la conservación de momento, como en el caso del efecto Compton. (Attix, 1986)

3.7.2.2. Interacción con la sección transversal para el efecto fotoeléctrico

La interacción con la sección transversal por átomo para el efecto fotoeléctrico, integrando sobre todos los ángulos de emisión del fotoelectrón, puede ser escrita como:

$$a^\tau \approx k \frac{Z^n}{(h\nu)^m} (cm^2/atom) \quad (3.37)$$

Donde k es una constante, $n \approx 4$ para $h\nu < 0.1 MeV$ y aumenta gradualmente a 4.6 a $3 MeV$, $m \approx 3$ a $0.1 MeV$ y disminuye gradualmente a 1 a $5 MeV$. (Attix, 1986)

3.7.3. Producción de pares

Producción de pares es un proceso de absorción en el cual los fotones desaparecen y dan lugar a un electrón y positrón. Este puede ocurrir en un campo de fuerza de Coulomb, usualmente cerca de núcleos atómicos. Sin embargo, puede tomar lugar con menos probabilidad en el campo de un electrón atómico, a este proceso usualmente se le conoce como “producción triple”, porque el electrón que proporciona el campo de Coulomb también adquiere energía cinética. El electrón o positrón es expulsado del sitio de la interacción. La energía mínima requerida por el fotón es $2m_0c^2 = 1.022 MeV$, para que ocurra la producción de pares en un campo nuclear. Para producción de triples es requerida una energía $4m_0c^2$. (Attix, 1986)

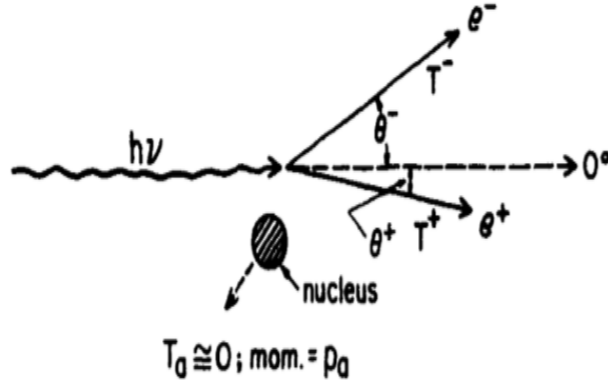


Figura 3.6: Producción de pares en un campo de fuerza de Coulomb de un núcleo atómico. Un fotón incidente con energía cuántica $h\nu$ se desvanece, dando lugar a un par electrón - positrón. El átomo participa en conservación de momento, pero recibe energía cinética $T_a \approx 0$.

Fuente: Attix (1986)

3.7.3.1. Producción de pares con campo de fuerza de Coulomb nuclear

Se ilustra en la Figura (2.6) un esquema de un evento de producción de pares en un campo nuclear. El fotón incidente con energía $h\nu$ da toda su energía cuántica en la creación del par electrón positrón con energía cinética T^- y T^+ . (Attix, 1986)

La ecuación de conservación de energía, ignorando la pequeña energía cinética dada el núcleo es la siguiente:

$$h\nu = 2m_0c^2 + T^- + T^+ = 1.022MeV + T^- + T^+ \quad (3.38)$$

3.7.3.2. Producción de pares en campo de Electrón

En la cinemática de producción de pares en un campo de electrón o producción de triples, el fotón divide su energía entre el positrón y electrón generado, y el electrón local. La conservación de la energía permite escribir la siguiente ecuación:

$$h\nu = 1.022MeV + T_1^- + T_2^- + T^+ \quad (3.39)$$

(Attix, 1986)

3.8 Física computacional

La física computacional consiste en la realización de experimentos numéricos en laboratorios de computación que permiten reproducir o predecir el comportamiento de un sistema, es decir, como evoluciona el sistema en el tiempo. Para ello, la física computacional necesita herramientas como

lo es el análisis numérico y la programación. (José and Ché, 2018)

3.8.1. Técnica de Monte Carlo

La técnica de Monte Carlo se define como una herramienta matemática desarrollada para resolver problemas relacionados con integrales múltiples, mediante una aproximación estadística utilizando números aleatorios. Consiste en diseñar un algoritmo computacional, a través de un código o programa, que genere un conjunto de números aleatorios distribuidos uniformemente bajo un rango predefinido. Siendo descartados aquellos números que no cumplen con la restricción dada por la integral que se desea aproximar. De esta forma, se obtiene la probabilidad de acierto, es decir, la razón entre la cantidad de números que cumplen la condición y la cantidad de números generados aleatoriamente. Esta probabilidad permite aproximar el valor de la magnitud de interés. Posteriormente, este proceso se realiza una cantidad de N veces, cada una con una semilla diferente, es decir, cada conjunto de números aleatorios es independiente entre sí. Una vez realizados, se calcula el valor aproximado de la magnitud de interés como el valor medio de todos los resultados. (José and Ché, 2018)

Sin embargo, la resolución por Monte Carlo requiere un número elevado de repeticiones del mismo experimento para obtener una incertidumbre aceptablemente baja. Esto se debe a que al teorema del límite central que establece que la incertidumbre asociada al valor esperado es proporcional a $1/\sqrt{N}$. Por lo tanto, si se busca disminuir la incertidumbre un orden de magnitud, es necesario multiplicar por 100 el número de repeticiones. Teniendo que los cálculos de Monte Carlo implican un método estadístico de estudiar los problemas, brindando una estimación lo suficientemente precisa de la solución exacta. Aunque se trata de un método específicamente desarrollado para abordar situaciones de índole probabilística, su aplicación también es válida en problemas de naturaleza determinista. (José and Ché, 2018)

4 GAMOS

4.1 Introducción a GAMOS

GAMOS o por su acrónimo en inglés “Geant4-based Architecture for Medicine-Oriented Simulations”, es software que simula el método de Monte Carlo y se basa en el conjunto de herramientas de Geant4. El objetivo de GAMOS es proveer una interfaz que sirva para que personas inexpertas puedan simular sus proyectos, son necesidad de tener conocimientos avanzados de C++ y con un conocimiento mínimo de Geant4, al mismo tiempo proporcionando un conjunto de funcionalidades avanzadas que sean fáciles de integrar con las funcionalidades de GAMOS. Está ideada para ser autónoma, de modo que no sea necesario profundizar en conocimientos de Geant4 para poder utilizarla, a menos que se necesite realizar un trabajo muy especializado. (CIEMAT, 2013)

GAMOS provee varias herramientas que ayudan a comprender a detalle la simulación que se realiza, controlando la verbosidad, haciendo histogramas de varias variables, contando diferentes cantidades, además de herramientas que ayudan a optimizar la simulación. Está compuesto por un software central que cubre la funcionalidad principal de una simulación Geant4 y un conjunto de aplicaciones para dominios específicos. (CIEMAT, 2013)

4.1.1. Estructura de GAMOS

Al compilar GAMOS se crean los siguientes directorios internos *tmp*, *lib*, *bin* y *module*. Otros directorios que utiliza GAMOS son:

- *source*: en este directorio es donde yace el código C++ de GAMOS.
- *examples*: en este directorio se encuentran ejemplos útiles para iniciarse en la plataforma GAMOS.
- *tutorials*: en este directorio se encuentran nueve tutoriales detallados paso por paso. Incluyendo ejercicios donde se va incrementando el nivel de dificultad los cuales van mejorando los resultados dependiendo de las soluciones que se provean.
- *analysis*: en este directorio se encuentran algunas herramientas útiles para analizar los resultados de GAMOS.
- *data*: en este directorio los algoritmos de GAMOS buscan archivos de datos.

4.1.2. Concepto plug-in

GAMOS provee una gran flexibilidad en escoger diferentes componentes para una simulación como: geometría, física, acciones de usuario, histogramas, etc, y combina estos componentes de una manera simple utilizando el concepto plug-in. El concepto plug-in significa que la estructura del programa se ejecuta sin componentes predefinidos y el usuario puede escoger que componentes utilizar en cada corrida listándolos de manera sencilla en un archivo de texto. Este mecanismo permite que el usuario pueda definir nuevos componentes que no sean nativos de GAMOS y usarlos juntamente con los componentes propios de GAMOS.

4.1.3. Geant4

Geant4 es un conjunto de herramientas computacionales de libre distribución para simular el transporte de partículas a través de la materia. Incorpora un conjunto muy extenso de partículas y materiales, así como de modelos de interacciones tanto hadrónicas como leptónicas, que permiten reproducir fenómenos de colisiones a energías comprendidas en un rango que se extiende desde 250eV hasta los TeV . Es un código desarrollado en el lenguaje C++ y diseñado de acuerdo con la Programación Orientada a Objetos (POO). Además, ha sido desarrollado de manera que el usuario pueda diseñar una simulación que se ajuste a sus intereses, así como también, obtener todo tipo de información a partir de la simulación. (José and Ché, 2018)

4.1.3.1. Estructura de Geant4

Dentro del código de Geant4 se puede encontrar objetos que permiten una gran flexibilidad y versatilidad al momento de definir la geometría y materiales que el problema involucra. Por otra parte, se encuentran clases con que se puede elegir los modelos físicos que describan un experimento similar. Además, se cuenta con la lista que define todas las partículas fundamentales incluidas en el PDG (*Particle Data Group*). Clases que permiten diseñar los detectores o volúmenes sensibles, y otras que facilitan la visualización de la geometría y la salida de los datos generados en la simulación. (José and Ché, 2018)

4.1.3.2. Núcleo de Geant4

En la Figura (3.1) se muestra un esquema de la relación que existe entre las diferentes categorías de Geant4. Las clases que están situadas en la parte de abajo del esquema son usadas por aquellas situadas en la parte superior.

- *Global*: define el sistema de unidades, constantes físicas y generados de números aleatorios.
- *Material y Particles (Materiales y Partículas)*: describe todas las propiedades físicas de las partículas y materiales incluidos en Geant4 para poder simular la interacción de cada

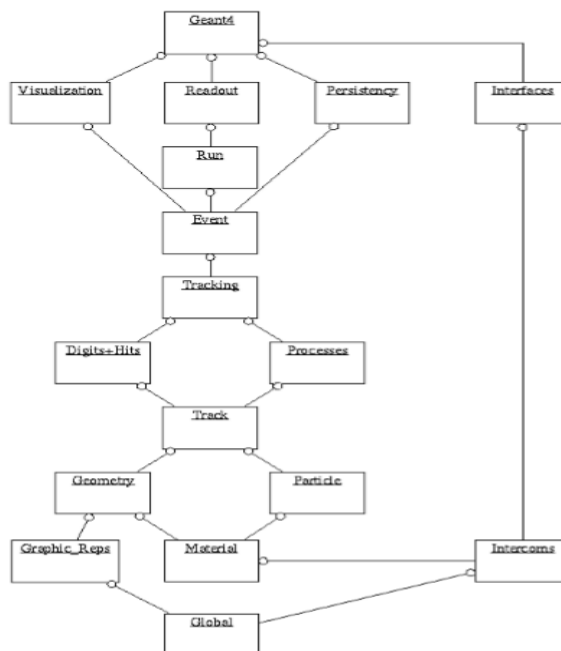


Figura 4.1: Estructura de las clases de Geant4.

Fuente: [http:// geant4-userdoc.web.cern.ch/geant4-userdoc/UsersGuides/IntroductionToGeant4/html/_images/classCategory.png](http://geant4-userdoc.web.cern.ch/geant4-userdoc/UsersGuides/IntroductionToGeant4/html/_images/classCategory.png). Geant4 Collaboration (2019)

partícula con la materia.

- *Geometry (Geometría)*: proporciona las clases que se encargan de describir la geometría de la simulación.
- *Track (Rastro)*: contiene las clases que se encargan de describir a la partícula en su posición inicial respecto de un volumen, además, las clases que se encargan de modificarlo en cada paso de la simulación.
- *Processes (Proceso)*: incluye todos los modelos y datos evaluados para reproducir las interacciones tanto electromagnéticas como hadrónicas, definiendo como se modifica el estado de las partículas en cada paso.
- *Tracking (Rastreo)*: controla la información relativa a la evolución del estado de cada partícula durante el transporte.
- *Digits and Hits (Impacto y Respuesta)*: las clases encargadas de controlar la respuesta de los volúmenes sensibles a partir de la información referente de la clase tracking.
- *Event (Evento)*: es la categoría que controla un evento completo, constituido por todas las partículas primarias generadas a partir de la situación en específico y están definidas por el generador de partículas primarias.

- *Run (Ejecución)*: las clases que se encargan de controlar la información de un conjunto de eventos que comparte el mismo generador de partículas primarias y una misma geometría.
- Las categorías encargadas de construir la interfaz de comunicación de Geant4 con el exterior son: *visualization (visualización)*, *persistency (persistencia)*, *interfaces (entornos)* y *readout (lectura de salida)*.

4.1.4. Tecnicismos relacionados a la arquitectura de GAMOS

Durante el desarrollo de la simulación se usarán tecnicismos, relacionados con la arquitectura de Geant4 y a su vez de GAMOS.

- *Run (corrida)*: consiste en una sucesión de eventos que comparten la misma geometría, la misma lista de física y el mismo generador de partículas primarias. La clase que la representa es *G4Run*, mientras que *G4RunManager* es la clase que gestiona todos los aspectos relativos a una corrida.
- *Event (evento)*: es la unidad básica de simulación de Geant4. Define todos los procesos transcurridos a partir de una determinada situación inicial dada por el generador de partículas primarias, puede ser una o varias. La clase que representa un evento es *G4Event*, el cual es gestionado por la clase *G4EventManager*.
- *Track (pista o trayectoria)*: es la imagen de la partícula en un instante dado. Esta información (posición, energía, momento) está contenida en la clase *G4Track*, que se actualiza por la información recibida en cada paso de la simulación de *G4Track* es *G4TrackingManager*.
- *Step (paso)*: define el cambio producido en el estado de una partícula entre un punto y otro punto de su camino. Se representa por un objeto de la clase *G4Step*, que es quien se encarga de modificar la información contenida en *G4Track*. Cada paso es gestionado por la clase *G4SteppingManager*.

4.1.5. Interfaz gráfica de GAMOS

La lista de botones que se pueden encontrar en la interfaz gráfica de GAMOS en orden como se observa en la Figura (3.2), son:

- *Open file (abrir un archivo)*: abre un archivo de entrada que ya haya sido creado.
- *Run (ejecutar)*: ejecuta GAMOS con el archivo de entrada que está en pantalla, este archivo es guardado antes de ejecutar.
- *Open log (registro)*: abre el archivo de registro de las acciones que realiza GAMOS con el archivo de entrada que se ha ejecutado.

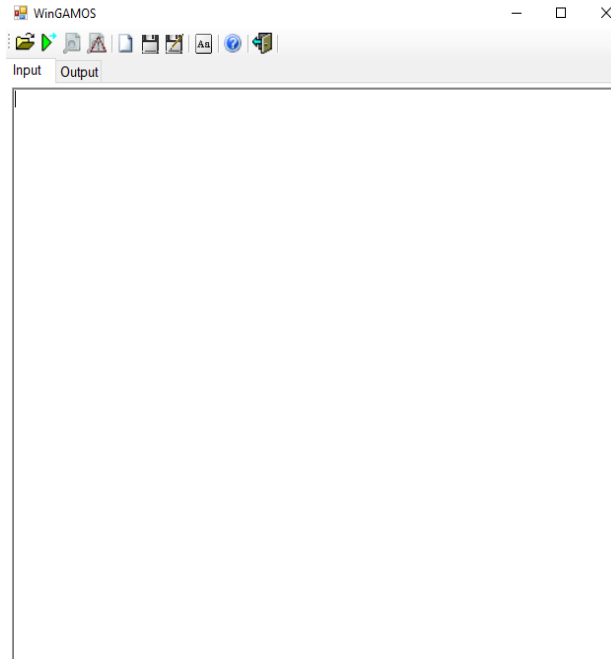


Figura 4.2: Interfaz gráfica GAMOS

- *Open error log (registro de errores)*: abre el archivo que registro de errores o excepciones que haya tenido el programa GAMOS al ejecutarse, si no contiene errores o excepciones, es un archivo en blanco.
- *New File (archivo nuevo)*: crea un nuevo archivo de entrada.
- *Save File (guardar archivo)*: guarda el archivo que se encuentra en la ventana, con el nombre actual que posee.
- *Save File As (guardar archivo como)*: guarda el archivo que se encuentra en la ventana, con la opción de cambiar el nombre que posee.
- *Selec Font (selección de la fuente)*: permite seleccionar el tipo de fuente para el archivo de entrada y los archivos de registro.
- *Help (ayuda)*: abre el archivo PDF “GAMOS User’s Guide” (Manual de usuario GAMOS).
- *Exit (salir)*: cierra la interfaz gráfica de GAMOS.

4.1.6. Interfaz gráfica de GAMOS en Linux (Ubuntu)

La lista de botones que se pueden encontrar en la interfaz gráfica de GAMOS en Linux en orden como se observa en la Figura (3.3), son:

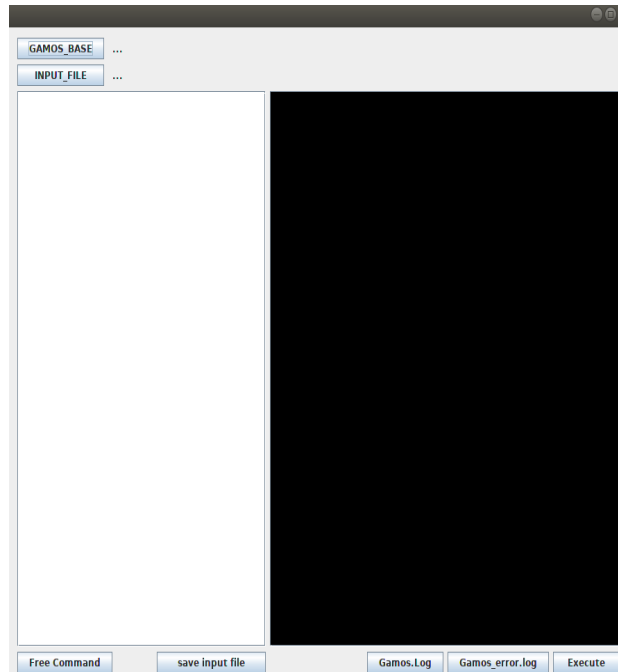


Figura 4.3: Interfaz gráfica GAMOS en Linux

- *GAMOS_BASE (base de GAMOS)*: con este botón se le indica a GAMOS cual es la carpeta base donde se encuentran todos los componentes de GAMOS. Para la máquina virtual ya instalada que ofrece GAMOS para VirtualBox este directorio es `/home/gamos/gamos/GAMOS.6.1.0`
- *INPUT_FILE (archivo de entrada)*: con este botón se le indica a GAMOS cual es el archivo `.in` que se desea ejecutar en la GUI (Interfaz Gráfica).
- *Free Command (Comando libre)*: este botón permite ejecutar comandos libres de archivo de entrada que se haya ingresado en el botón anterior.
- *save input file (guardar archivo de entrada)*: este botón permite guardar los cambios hechos desde la GUI en el archivo de entrada, para que estos cambios sean ejecutados es necesario utilizar este botón, de lo contrario seguirá ejecutando el archivo de entrada original.
- *Gamos.log (archivo de salida)*: este botón permite al usuario abrir el archivo de salida del programa ejecutado sin necesidad de ir a buscar este archivo al directorio.
- *Gamos_error.log (archivo error de salida)*: este botón permite al usuario abrir el archivo de error del programa ejecutado sin necesidad de ir a buscar este archivo al directorio.
- *Execute (ejecutar)*: este botón ejecuta el archivo de entrada que el usuario haya cargado.

4.2 Geometría

Existen tres formas diferentes de describir una geometría en GAMOS: a través de un archivo de texto, utilizando alguna de las geometrías de los ejemplos provistos en las librerías de GAMOS o Geant4, por una clase C++ heredada de G4VUserDetectorConstruction. La más práctica de usar y la única desarrollada en esta tesis es la geometría proveniente de un archivo de texto.

4.2.1. Geometría desde un archivo de texto

Para implementar las geometrías definidas por el usuario es necesario utilizar el comando `/gamos/geometry GmGeometryFromText` en el archivo de entrada donde está nuestra simulación, para indicar que se desea importar de un archivo de texto. Luego se especifica el nombre del archivo de texto que se desea implementar utilizando la siguiente línea. `/gamos/setParam GmGeometryFromText:FileName NombreDeLaGeometria.geom`

La descripción de la geometría es basada en etiquetas. Una etiqueta es la primera palabra que aparece al inicio de cada línea e indica que significa la línea. No hay restricción del orden de las etiquetas en el archivo, excepto de algunas restricciones lógicas. Por ejemplo, no se puede posicionar un volumen en el espacio sino se ha definido. Se indicarán en esta sección algunas de estas etiquetas, los parámetros que reciben y para que sirven.

- **Isótopo (:ISOT)**: Esta etiqueta sirve para definir isótopos. Acepta cuatro parámetros: Nombre, Z, N, A. Nombre es el nombre que se le quiere dar al isótopo, Z es el número de protones, N es el número de neutrones y A es la masa atómica.
- **Material hecho de una mezcla de varios elementos (:MIXT)**: Esta etiqueta sirve para definir materiales compuestos por varios elementos. Acepta cinco parámetros: Nombre, Densidad, número de componentes, nombre del material, proporción del material en la mezcla. Además, en el archivo “source/materials/src/G4NistMaterialBuilder.c” y “/data/PET_materials.txt” en el directorio de GAMOS se encuentra una lista de materiales predefinidos que pueden ser fácilmente utilizados. Es necesario ingresar los parámetros en el orden que se especifica y si no saltar parámetros a menos que se especifique que este ya tiene un valor determinado por el programa que queramos usar.
- **Propiedades del material (:MATE_MEE)**: Esta etiqueta sirve para definir propiedades de un material anteriormente definido. Acepta dos parámetros: Nombre del material ya definido, promedio de excitación de energía.
- **Sólidos (:SOLID)**: Esta etiqueta sirve para definir algún volumen sólido, sin cavidades. Acepta tres parámetros: el nombre que se le quiere poner al sólido, el tipo de sólido y los

parámetros correspondientes al tipo de sólido elegido. En la siguiente sección se definen los tipos de sólido.

- **Volumen** (:VOLU): Esta etiqueta sirve para definir volúmenes lógicos, se puede definir de un sólido a de un material asociado a él. Esta etiqueta recibe tres parámetros: el nombre del volumen, el nombre del sólido ya definido, el nombre del material.
- **Situar** (:PLACE): Esta etiqueta sirve para situar volúmenes lógicos, es necesario tener ya definido un volumen madre o primario en el que se desee colocar el volumen secundario. Esta etiqueta recibe siete parámetros: el nombre del volumen, número de la copia que se está colocando, el nombre del volumen madre o primario, nombre de la matriz de rotación, posición en X, posición en Y, posición en Z.
- **Situar con parametrización** (:PLACE_PARAM): Esta etiqueta sirve para situar volúmenes lógicos con varias copias a lo largo de un eje, un círculo o una malla bidimensional, es necesario tener ya definido un volumen madre o primario en el que se desee colocar el volumen secundario. Esta etiqueta recibe nueve parámetros: el nombre del volumen, el número de la copia del volumen, el nombre del volumen madre o primario, tipo de parametrización, nombre de la matriz de rotación, número de copias que se está colocando, separación entre las copias, compensación, argumentos extras que dependen del tipo de parametrización.
 - Parametrización lineal: (*LINEAR_X*, *LINEAR_Y*, *LINEAR_Z*), es necesario agregar los parámetros extra: *DIR_X*, *DIR_Y*, *DIR_Z*, la compensación para la parametrización lineal representa la distancia del centro hacia la primer copia en el punto (0,0,0) a lo largo de la línea.
 - Parametrización circular: el círculo esta por defecto alrededor del eje Z, se pueden proveer como parámetros extra el eje de simetría y la posición de la primera copia.
 - Parametrización cuadrada: esta parametrización se puede hacer en los planos (*XY*, *XZ*, *YZ*), se definen como (*SQUARE_XY*, *SQUARE_XZ*, *SQUARE_YZ*), se le pueden poner como parámetros extra, dos números para la cantidad de copias, dos espacios de separación diferente, opcionalmente dos compensaciones.
- **Situar fantasmas con parametrización** (:PLACE_PARAM): Esta etiqueta también se puede usar para situar una malla tridimensional de voxels (cubos pequeños) del mismo tamaño. Esta etiqueta recibe once parámetros: el nombre del volumen, el número de copia del volumen, el nombre del volumen madre o primario, tipo de parametrización (PHANTOM para este caso), nombre de la matriz de rotación, número de copias en el eje X, número de copias en el eje Y, número de copias en el eje Z, separación entre las copias en X, separación entre las copias en Y, separación entre las copias en Z.

- **Matriz de rotación (:ROTM):** La matriz de rotación es interpretada como una rotación aplicada a un objeto tomando el marco de referencia del volumen madre. Puede ser definida de tres maneras distintas:
 - a) Utilizando tres ángulos de rotación a lo largo de X, Y, Z. Por lo tanto, aceptaría cuatro parámetros: nombre de la matriz de rotación, ángulo de rotación en X, ángulo de rotación en Y, ángulo de rotación en Z.
 - b) Utilizando ángulo polar y azimutal de X, Y, Z. Por lo tanto, aceptaría siete parámetros: nombre de la matriz de rotación, ángulo polar en X, ángulo azimutal en X, ángulo polar en Y, ángulo azimutal en Y, ángulo polar en Z, ángulo azimutal en Z.
 - c) Utilizando los nueve elementos de una matriz de rotación en tres dimensiones: XX, XY, XZ, YX, YY, YZ, ZX, ZY, ZZ. Por lo tanto, aceptaría diez parámetros: nombre, nueve elementos de matriz en el orden ya dado.

4.2.1.1. Tipos de sólido

GAMOS provee algunas figuras geométricas básicas para poder realizar geometrías sencillas, se enlistan algunas de estas figuras útiles.

- **Cajas (:BOX):** Esta etiqueta recibe tres parámetros: la mitad de la longitud en X, la mitad de la longitud en Y, la mitad de la longitud en Z.
- **Tubo (:TUBE):** Esta etiqueta recibe tres parámetros: radio interno, radio externo y la mitad de largo en Z.
- **Cono (:CONE):** Esta etiqueta recibe cinco parámetros: radio interno en la parte de abajo del cono, radio externo en la parte de abajo del cono, radio interno en la parte de arriba del cono, radio externo en la parte de a arriba del cono, la mitad de largo en Z.
- **Trapezoide (:TRD):** Esta etiqueta recibe cinco parámetros: la mitad de la distancia en X en la parte de abajo del trapezoide, la mitad de la distancia en X en la parte de arriba del trapezoide, la mitad de la distancia en Y en la parte de abajo del trapezoide, la mitad de la distancia en Y en la parte de arriba del trapezoide, la mitad de largo en Z.
- **Esfera (:SPHERE):** Esta etiqueta recibe seis parámetros: radio interno, radio externo, ángulo inicial phi del segmento, delta o tamaño del segmento de ángulo phi, ángulo inicial theta, delta o tamaño del segmento de ángulo theta.
- **Toro (:TORUS):** Esta etiqueta recibe cinco parámetros: radio interno, radio externo, radio de barrido del toro, ángulo inicial phi, delta o tamaño del segmento de ángulo phi.

4.2.1.2. Etiquetas de visualización

GAMOS permite realizar modificaciones de visibilidad, color y transparencia, para poder visualizar de una mejor manera las geometrías.

- **Visibilidad** (:VIS): Esta etiqueta sirve para definir si queremos que la geometría que definimos sea visible, al momento de hacer la reconstrucción de la simulación o no. Recibe dos parámetros: Nombre del volumen, Verdadero (TRUE) o falso (FALSE).
- **Color y transparencia** (:COLOUR /:COLOR): Esta etiqueta hace posible definir el color de un volumen y el nivel de transparencia que queremos que tenga, los valores de proporción entre cada color y transparencia pueden estar entre 0 – 1, por defecto está definido el uno para cada color. Recibe cinco parámetros: el nombre del volumen, proporción de color rojo, proporción de color verde, proporción de color azul, transparencia.

4.2.2. Unidades

GAMOS trae algunos valores de unidades por defecto, los cuales pueden ser cambiados en la simulación dependiendo de cuales se necesiten. Los valores de unidades por defecto son los siguientes:

- longitud: mm
- ángulo: grados
- densidad: g/cm^3
- masa atómica: g/mol
- temperatura: kelvin
- presión: atmósferas

El usuario puede cambiar las unidades de GAMOS, indicando la unidad que desea utilizar en cada valor ingresado. Esto puede ser implementado agregando al final del valor el nombre de la unidad, por ejemplo 3. * m.

4.2.3. Árbol de volúmenes

GAMOS provee el un comando `/gamos/geometry/printVolumeTree NIVEL_VERBOSIDAD` que imprime en el archivo `gamos.log` la cantidad de información sobre los volúmenes definidos que el usuario desee conocer. El parámetro `NIVEL_VERBOSIDAD` acepta un dígito de dos números, el segundo dígito (de las unidades) controla la información sobre sólidos y materiales:

- = 1 Imprime el nombre del volumen lógico

- = 2 Imprime el nombre del sólido, nombre del material, volumen cúbico y la masa del volumen
- = 3 Imprime los parámetros del sólido y los atributos de visualización.

El primer dígito (el de las décimas) controla la información de los volúmenes físicos.

- = 1 Imprime el nombre del volumen físico, número de copias y el nombre del volumen madre o principal.
- = 2 Imprime la posición y rotación del volumen físico.
- = 3 Imprime los datos de replica o parametrización del volumen físico.

4.3 Generador

El generador de GAMOS provee varias distribuciones de tiempo, energía, posición y dirección que el usuario puede combinar a su voluntad. El usuario puede seleccionar si desea generar partículas primarias o la cantidad de partículas que desea generar junto con uno o varios isótopos diferentes, además, puede configurar que tipos de distribuciones desea implementar en la simulación para las partículas primarias o isótopos.

El comando para indicar que se va a utilizar el generador de GAMOS es: `/gamos/generator GmGenerator`, este comando inicia el generador GAMOS, si no se encuentra antes de utilizar el generador, GAMOS retornará una excepción. Luego de iniciar el generador GAMOS se agrega una o varias fuentes, no existe una fuente por defecto, por ende, GAMOS retornará una excepción sino se agrega una. Es posible agregar varias fuentes de partículas por evento, usando repetidas veces los comandos para agregar fuentes.

4.3.1. Fuentes de partículas

4.3.1.1. Fuente simple de partículas

Para agregar una fuente simple de partículas se utiliza el siguiente comando `/gamos/generator/addSingleParticleSource NOMBRE_DE_LA_FUENTE NOMBRE_DE_LA_PARTICULA ENERGIA`, donde `NOMBRE_DE_LA_FUENTE` es el nombre que el usuario desea darle a la fuente, `NOMBRE_DE_LA_PARTICULA` es una de las partículas que provee Geant4 y `ENERGIA` es la energía inicial de la partícula. Sino se cambia ninguna de las propiedades de la partícula, se genera en un tiempo $t=0$, posición $(0,0,0)$ y dirección random, más adelante se detalla como cambiar alguna de estas propiedades.

4.3.1.2. Fuente de isótopos

GAMOS implementa un generador de isótopos, que simula la actividad de diferentes isótopos que decaen en uno o varios fotones, electrones o positrones. La lista de isótopos disponibles

en GAMOS se encuentra en el siguiente directorio *MY_GAMOS_DIR/data/isotopes.dat*. Cada isótopo está definido por la etiqueta *:ISOTOPE*, luego se encuentra separada por espacio el nombre del isótopo, en las siguientes líneas se encuentran cada una de las posibles partículas en las que puede decaer el isótopo, el primer valor corresponde a la ENERGIA de decaimiento, el segundo valor es la probabilidad de decaimiento y el tipo de partícula. Es posible añadir nuevos isótopos a este archivo se conoce la información necesaria y disponiendo la misma como se describió anteriormente. Para poder utilizar alguno de los isótopos de este archivo se utiliza el comando */gamos/generator/addIsotopeSource NOMBRE_FUENTE NOMBRE_ISÓTOPO ACTIVIDAD_DE_LA_FUENTE*, donde *NOMBRE_FUENTE* es el nombre que el usuario le da a la fuente, *NOMBRE_ISÓTOPO* es el nombre del isótopo que se desea utilizar de la base de datos *isotopes.dat*, y *ACTIVIDAD_DE_LA_FUENTE* es la actividad que se le desea dar a la fuente en dimensional de becquerel (Bq). Para declarar varias fuentes, se utiliza el mismo comando con las diferentes fuentes que se desean utilizar.

4.3.2.Distribuciones

4.3.2.1.Distribuciones de tiempo

GAMOS provee tres diferentes tipo de distribuciones para el tiempo.

- **Tiempo constante:** el comando para generar una distribución de tiempo constante es */gamos/generator/timeDist NOMBRE_FUENTE GmGenerDistTimeConstant TIEMPO* donde *NOMBRE_FUENTE* es el nombre de la fuente que ya se definió, *TIEMPO* es la cantidad de tiempo que se desea realizar la distribución.
- **Tiempo cambiando en intervalos constante:** el comando para generar esta distribución es */gamos/generator/timeDist NOMBRE_FUENTE GmGenerDistTimeConstantChange INTERVALO_DE_TIEMPO COMPENSACION_TIEMPO*, donde *NOMBRE_FUENTE* es el nombre de la fuente que ya se definió, *INTERVALO_DE_TIEMPO* es el intervalo que el usuario desea que el tiempo vaya cambiando y *COMPENSACION_TIEMPO* es el tiempo que desea que se pase hasta el primer evento.
- **Tiempo de decaimiento:** el comando para generar este tipo de distribución es */gamos/generator/timeDist NOMBRE_FUENTE GmGenerDistTimeDecay ACTIVIDAD TIEMPO_DE_VIDA* donde *NOMBRE_FUENTE* es el nombre de la fuente que ya se definió, *ACTIVIDAD* es la actividad de la fuente y *TIEMPO_DE_VIDA* es el tiempo de vida escalado con la actividad seleccionada.

4.3.2.2. Distribuciones de energía

GAMOS provee siete distribuciones de energía distintas. En esta sección solo se adjunta la tipo de distribución y el comando.

- **Energía constante:** el comando para generar este tipo de distribución es `/gamos/generator/energyDist NOMBRE_FUENTE GmGenerDistEnergyConstant ENERGIA`, donde NOMBRE_FUENTE es el nombre de la fuente que ya se definió, ENERGIA es la ENERGIA que el usuario desea que tengan las partículas primarias.
- **Energía de decaimiento constante:** el comando para generar esta distribución es `/gamos/generator/energyDist NOMBRE_FUENTE GmGenerDistEnergyConstantIsotopeDecay`, donde NOMBRE_FUENTE es el nombre de la fuente que ya se definió.
- **Energía aleatoria con probabilidad constante:** el comando para generar este tipo de distribución es `/gamos/generator/energyDist NOMBRE_FUENTE GmGenerDistEnergyRandomFlat ENERGIA_MIN ENERGIA_MAX` donde NOMBRE_FUENTE es el nombre de la fuente que ya se definió, ENERGIA_MIN es la energía mínima y ENERGIA_MAX es la energía máxima.
- **Energía de decaimiento Beta:** el comando para generar este tipo de distribución es `/gamos/generator/energyDist NOMBRE_FUENTE GmGenerDistEnergyBetaDecay` donde NOMBRE_FUENTE es el nombre de la fuente que ya se definió.
- **Gaussiana:** el comando para generar este tipo de distribución es `/gamos/generator/energyDist SOURCE_NAME GmGenerDistEnergyGaussian MEDIA SIGMA`, donde NOMBRE_FUENTE es el nombre de la fuente que ya se definió, MEDIA Y SIGMA son parámetros de la función Gaussiana.
- **Utilizando probabilidades desde un archivo de texto:** el comando para generar este tipo de distribución es `/gamos/generator/energyDist NOMBRE_FUENTE GmGenerDistEnergyFromFile NOMBRE_ARCHIVO TIPO_CALCULO UNIDAD`, donde NOMBRE_FUENTE es el nombre de la fuente que ya se definió, NOMBRE_ARCHIVO es el nombre del archivo que ya se encuentra en el directorio, TIPO_CALCULO son el tipo de calculo que se desea hacer y UNIDAD son las unidades de energía.
- **Isolethargica:** el comando para generar este tipo de distribución es `/gamos/generator/energyDist SOURCE_NAME GmGenerDistEnergyIsolethargical ENERGIA_MIN ENERGIA_MAX`, donde NOMBRE_FUENTE es el nombre de la fuente que ya se definió, ENERGIA_MIN es la energía mínima y ENERGIA_MAX es la energía máxima.

4.3.2.3. Distribución de posición

GAMOS provee varios tipos de distribuciones para posición, se detallaran dos de los más comunes y útiles.

- **Posición en un punto:** el comando para generar este tipo de distribución es */gamos/generator/positionDist NOMBRE_FUENTE GmGenerDistPositionPoint POS_X POS_Y POS_Z*, donde NOMBRE_FUENTE es el nombre de la fuente que ya se definió, POS_X, POS_Y, POS_Z son las posiciones en X, Y, Z.
- **Posición en un volumen Geant4:** el comando para generar esta distribución es */gamos/generator/positionDist NOMBRE_FUENTE GmGenerDistPositionInG4Volumes NOMBRE_VOL1 NOMBRE_VOL2 ...*, donde NOMBRE_FUENTE es el nombre de la fuente que ya se definió, NOMBRE_VOL1, NOMBRE_VOL2, ... es la lista de volúmenes ya definidos en los que se desea hacer la distribución.

4.3.2.4. Distribuciones de dirección

GAMOS provee cuatro tipos de distribuciones de dirección, se detallan en la siguiente lista.

- **Distribución aleatoria:** el comando para generar este tipo de distribución es */gamos/generator/directionDist NOMBRE_FUENTE GmGenerDistDirectionRandom*, donde NOMBRE_FUENTE es el nombre de la fuente que ya se definió.
- **Distribución constate:** el comando para generar esta distribución es */gamos/generator/directionDist NOMBRE_FUENTE GmGenerDistDirectionConst DIR_X DIR_Y DIR_Z*, donde NOMBRE_FUENTE es el nombre de la fuente que ya se definió, DIR_X, DIR_Y, DIR_Z son las direcciones en X, Y, Z.
- **Distribución en forma de cono:** el comando para generar esta distribución es */gamos/generator/directionDist NOMBRE_FUENTE GmGenerDistDirectionCone DIR_X DIR_Y DIR_Z ANGULO_APERTURA*, donde NOMBRE_FUENTE es el nombre de la fuente que ya se definió, DIR_X, DIR_Y, DIR_Z son las direcciones en X, Y, Z en la que se desea que abra el cono, ANGULO_APERTURA es el ángulo de apertura del cono.
- **Gaussiana a lo largo de un eje:** el comando para generar esta distribución es */gamos/generator/directionDist NOMBRE_FUENTE GmGenerDistDirectionGaussian DIR_X DIR_Y DIR_Z SIGMA_Y SIGMA_Z*, donde NOMBRE_FUENTE es el nombre de la fuente que ya se definió, DIR_X, DIR_Y, DIR_Z son las direcciones en X, Y, Z en la que se desea que abra la gaussiana, SIGMA_Y, SIGMA_Z son parámetros de la Gaussiana.

4.4 Acciones de usuario

Las acciones de usuario de Geant4 son la forma en las que el usuario puede interactuar con el programa al principio, al final y durante la simulación. GAMOS provee todas las funcionalidades de las clases de Geant4 para las acciones de usuario. Además, permite al usuario definir varios tipos de acciones de usuarios del mismo tipo en el mismo trabajo, al mismo tiempo. Estas acciones de usuario pueden ser asociadas a filtros o clasificadores. Todas las acciones de usuario se inician utilizando el siguiente comando: `/gamos/userAction NOMBRE_ACCION_USUARIO`. Las acciones de usuario pueden actuar sobre una corrida, un evento, una trayectoria y los pasos de una trayectoria. Algunas de las acciones de usuario son: *GmUserRunAction*, *GmUserEventAction*, *GmUserTrackingAction*, *GmUserSteppingAction* y *GmUserStackingAction*.

4.4.1. Filtros

Un filtro es una clase de Geant4, que recibe información de las trayectorias (track) o de cada paso (step) del evento. Un filtro GAMOS tiene dos métodos principales.

- *AcceptStep* (Acepta el paso): recibe un objeto del tipo G4Step y decide si retorna verdadero o falso.
- *AcceptTrack* (Acepta la trayectoria): recibe un objeto del tipo G4Track y decide si devolver verdadero o falso.

Un filtro puede implementar uno de estos métodos o los dos, si el método *AcceptStep* no está implementado en el filtro, se invoca el método *AcceptTrack* que busca la trayectoria en la que el paso está involucrado, si *AcceptTrack* no está implementado en el filtro, el mismo retorna siempre verdadero.

Los filtros pueden actuar sobre acciones de usuario o contadores. Si uno o varios filtros son conFigurados de modo que actúen sobre una acción de usuario, los métodos se aplica el método *AcceptTrack* como verdadero, si todos los métodos aplicados son verdaderos. En el caso en que se apliquen filtros a contadores, estos solo actuaran si el contador ya ha sido llamado y si el paso ocurre en el volumen seleccionado.

Combinar filtros y contadores son herramientas poderosas, que pueden ayudar a obtener información detallada de la simulación al usar algunos comandos simples. Varios filtros necesitan parámetros extra que ayudan a controlar las variables de su entorno y no eliminar información que puede ser importante para los resultados para utilizar estos filtros se utilizan el siguiente comando `/gamos/filter NOMBRE_FILTRO CLASE_FILTRO PARAMETRO1 PARAMETRO2 ...`, donde *NOMBRE_FILTRO* es el nombre del filtro que le ha asignado el usuario, para asignárselo a un contador, *CLASE_FILTRO* es la clase del filtro que se está usando. Y *PARAMETRO1*

PARAMETRO2 ... es la lista de parámetros que el filtro necesita.

Para asignar un filtro a una acción de usuario se utiliza el siguiente comando `/gamos/userAction ACCION_USUARIO NOMBRE_FILTRO` y se puede usar filtros para que actúen sobre contadores utilizando el siguiente comando `/gamos/scoring/addFilter2Scorer NOMBRE_FILTRO NOMBRE_CONTADOR`.

4.4.1.1. Filtros simples

GAMOS provee una lista de filtros disponibles que pueden ser utilizados de forma simple con los comandos dados en la sección anterior. Algunos de estos son:

- *GmGammaFilter*
- *GmElectronFilter*
- *GmPositronFilter*
- *GmElectronOrPositronFilter*
- *GmEMParticleFilter*
- *GmProtonFilter*
- *GmNeutronFilter*
- *GmParticleFilter*
- *GmChargedFilter*
- *GmNeutralFilter*
- *GmPrimaryFilter*
- *GmSecondaryFilter*
- *GmDepositedEnergyFilter*
- *GmKineticEnergyFilter*
- *GmProcessFilter*
- *GmParticleProcessFilter*

4.4.1.2.Filtros de volumen

GAMOS provee una lista de condiciones para poder crear filtros de volumen dependiendo de la necesidad del usuario. Estas condiciones son las siguientes:

- *In*: la partícula está en el volumen
- *Enter*: la partícula está entrando a un volumen. El método *AcceptTrack* siempre devuelve falso.
- *Exit*: la partícula está saliendo de un volumen. El método *AcceptTrack* siempre devuelve falso.
- *Traverse*: la partícula atraviesa un volumen, no entra en el ni sale de él.
- *Start*: la particular inicia en el volumen. El método *AcceptTrack* puede devolver verdadero solo si el primer paso de la trayectoria está en el volumen.
- *End*: la particular termina en el volumen. El método *AcceptTrack* puede devolver verdadero solo si el último paso de la trayectoria está en el volumen.

Un ejemplo del comando necesario para utilizar estos filtros es el siguiente `/gamos/filter NOMBRE_FILTRO GmEndLogicalVolumeFilter NOMBRE_VOLUMEN`, este filtro como se puede ver busca filtrar todas las partículas que terminan en el volumen llamado NOMBRE_VOLUMEN. Algunos tipos de volumen disponibles en GAMOS se encuentran en la siguiente lista, con una breve descripción.

- *LogicalVolume* o volumen lógico: es un volumen del tipo *G4LogicalVolume*.
- *PhysicalVolume* o volumen físico: es un volumen del tipo *G4PhysicalVolume*.
- *Touchable* o tocable: es un volumen del tipo *G4VTouchable*.
- *Traverse*: la partícula atraviesa un volumen, no entra en el ni sale de él.
- *Region*: es un volumen del tipo *G4Region*.

4.4.1.3.Filtros de filtros

Para combinar uno o varios filtros GAMOS provee filtros que ayudan a combinar varios filtros para trabajos más complejos. Se enlistan algunos de estos.

- *GmORFilter*: devuelve verdadero si alguno de los filtros devuelve verdadero.
- *GmXORFilter*: devuelve verdadero su solo uno de los filtros devuelve verdadero.
- *GmANDFilter*: devuelve verdadero si todos los filtros devuelven verdadero.
- *GmInverseFilter*: devuelve lo opuesto del filtro que recibe como parámetro.

4.4.2. Contadores

Geant4 provee diferentes clases para poder contar diferentes cantidades en el volumen seleccionado. GAMOS provee todas las funcionalidades de Geant4 en cuanto a comandos y ofrece algunas funcionalidades extra. Para que GAMOS cree un detector multifuncional y asociar este a una lista de volúmenes lógicos se utiliza el siguiente comando `/gamos/scoring/createMFDetector NOMBRE_DETECTOR NOMBRE_VOLUMENES_LOGICOS`, donde `NOMBRE_DETECTOR` es el nombre que el usuario le asigna al detector, `NOMBRE_VOLUMENES_LOGICOS` es la lista de volúmenes lógicos ya definidos en la simulación.

Luego se le agrega al detector uno de los contadores de GAMOS, utilizando el comando `textit /gamos/scoring/addScorer2MFD NOMBRE_CONTADOR CONTADOR_CLASE NOMBRE_DETECTOR PARAMETROS_CONTADOR`, donde `NOMBRE_CONTADOR` es el nombre que el usuario le asigna al contador, `CONTADOR_CLASE` es la clase del contador, `NOMBRE_DETECTOR` es el nombre del detector creado por el comando anterior, `PARAMETROS_CONTADOR` son los parámetros opcionales que dependen del contador.

Para cada uno de los contadores agregados es posible agregar un filtro, para seleccionar que trayectorias serán contadas, se utiliza el siguiente comando `textit /gamos/scoring/addFilter2Scorer NOMBRE_FILTRO/CLASE NOMBRE_CONTADOR`, donde `NOMBRE_FILTRO` es el nombre que el usuario le agrego a uno de los filtros, `CLASE` es la clase del filtro, `NOMBRE_CONTADOR` es el nombre que el usuario le agrego al contador.

Finalmente se puede seleccionar el formato que se desea utilizar para imprimir los resultados del contador asociando una de las impresoras disponibles para cada contador con el siguiente comando `/gamos/scoring/addPrinter2Scorer NOMBRE_IMPRESORA NOMBRE_CONTADOR`, donde `NOMBRE_IMPRESORA` es el nombre que de una impresora del tipo *GmVPSPrinter* que se construye utilizando el comando `/gamos/printer`. Los contadores pueden ser clasificados en tres tipos:

- *Contadores de trayectorias*
 - *GmG4PSTrackLength*: suma de todas los largos de trayectoria en cada paso de la misma.
 - *GmG4PSPassageTrackLength*: Solo suma los largos de trayectoria, en el volumen seleccionado.

- *Contadores de energía depositada*
 - *GmG4PSEnergyDeposit*: suma la energía que depositan las partículas en cada paso en la celda. Devuelve un valor esperado de la energía depositada por partícula.
 - *GmG4PSDoseDeposit*: suma la energía depositada en la celda en cada paso, dividido por la masa de la celda. Devuelve un valor esperado de la dosis depositada por partícula.

- *GmG4PSKerma*: suma el kerma de todas las partículas en cada paso en las celdas.
- *Contadores de corriente y flujo*
 - *GmPSSurfaceFlux*: Este es un contador basado en la superficie, toma la cantidad de trayectorias que entran a la superficie, dividido el área y el coseno del ángulo entre la trayectoria y la normal de la superficie.
 - *GmG4PSPassageCellCurrent*: Corriente basada en el volumen, la corriente está definida como el número de trayectorias que pasan a través de un volumen.
 - *GmG4PSCellFlux*: Flujo basado en el volumen, se calcula como la longitud de trayectoria dentro del volumen, dividido el volumen de la celda.
 - *GmG4PSPassageCellFlux*: Flujo basado en el volumen, se calcula como la longitud de trayectoria dentro del volumen, dividido el volumen de la celda. La diferencia con el anterior, es que no se toman en cuenta partículas que terminaron su trayectoria dentro de la celda.

4.4.2.1. Impresoras de contadores

La impresora de un contador sirve para seleccionar el formato de salida del contador. Esta funcionalidad es única de GAMOS. Para poder asociar una impresora a un contador se utiliza el comando `/gamos/scoring/addPrinter2Scorer NOMBRE_IMPRESORA NOMBRE_CONTADOR`, donde NOMBRE_IMPRESORA es el nombre que el usuario le asigna a la impresora, este servirá para poder agregar una clase a la impresora y los parámetros, NOMBRE_CONTADOR es el nombre del contador al que se le asigna la impresora, este nombre se lo asignó el usuario al agregar el contador. Para poder configurar la clase de la impresora y los parámetros específicos se utiliza el siguiente comando `/gamos/scoring/printer NOMBRE_IMPRESORA CLASE_IMPRESORA PARAMETROS`, donde NOMBRE_IMPRESORA es el nombre de la impresora que se le asignó en el comando anterior al contador, CLASE_IMPRESORA es la clase de impresora que el usuario desea en que GAMOS retorne la información recopilada por el contador, PARAMETROS son los parámetros opcionales específicos de cada impresora. Las diferentes impresoras que proporciona GAMOS se enlistan a continuación:

- *GmPSPrinterG4cout*: Imprime la salida estándar del resumen del contador. El resumen estándar imprime la siguiente información: nombre del detector (“MultiFunctionalDet”), nombre de contador (“PrimitiveScorer”), número de entradas (“Number of entries”), la información recolectada por el contador con índice y el valor de la cantidad medida, la suma de todos los valores medidos con su respectivo error.
- *GmPSPrinterTextFile*: Imprime en un archivo de texto la misma información de la impresora *GmPSPrinterG4cout*.

- *GmPSPrinterBinFile*: Imprime en un archivo binario la misma información que la impresora *GmPSPrinterG4cout*.
- *GmPSPrinterHistos*: Imprime los datos recolectados por el contador en histogramas.

4.5 Física electromagnética de GAMOS

Para poder utilizar la lista de objetos electromagnéticos físicos básicos de GAMOS, la cual define todas las partículas electromagnéticas como: fotones, electrones, positrones y fotones ópticos, se utiliza el siguiente comando: `/gamos/physicsList GmEMPhysics`. Este comando indica a GAMOS, donde extraer la información para los procesos de partículas gamma, electrones, positrones y fotones ópticos. Para partículas gamma y electrones. Para partículas gamma y electrones utiliza los modelos de bajas energías, para positrones utiliza los modelos estándar. A continuación se detallan los procesos disponibles para cada una de las partículas anteriormente mencionadas.

- Partículas *gamma*
 - *Fotoeléctrico*: es del tipo `G4PhotoElectricEffect`, utiliza el modelo `G4LivermorePhotoElectricModel` para energías por debajo de 1GeV y el modelo `G4PEEffectFluoModel` para energías mayores.
 - *Dispersión de Compton*: es del tipo `G4ComptonScattering`, utiliza el modelo `G4LivermoreComptonModel` para energías por debajo de 1GeV y el modelo `G4KleinNishinaCompton` para energías mayores.
 - *Conversión gamma*: es del tipo `G4GammaConversion`, utiliza el modelo `G4LivermoreGammaConversionModel` para energías por debajo de 1GeV y el modelo `G4BetheHeitlerModel` para energías mayores.
 - *Dispersión de Rayleigh*: es del tipo `G4RayleighScattering`, utiliza el modelo `G4LivermoreRayleighModel` para energías por debajo de 1GeV y nada para energías mayores.
- *Electrones*:
 - *Ionización*: Es del tipo `G4eIonisation`, utiliza el modelo `G4LivermoreIonisationModel` con el submodelo `G4UniversalFluctuation` para energías por debajo de 1GeV y el modelo `G4MollerBhabhaModel` para energías mayores.
 - *Bremsstrahlung*: Es del tipo `G4eBremsstrahlung`, utiliza el modelo `GmLivermoreBremsstrahlungModel` para energías por debajo de 1GeV y el modelo `G4eBremsstrahlungRelModel` para energías mayores.
 - *Dispersión múltiple*: Es del tipo `G4VMultipleScattering`, utiliza el modelo `G4UrbanMscModel96`.

- *Positrones:*
 - *Ionización:* Es del tipo G4eIonisation, utiliza el modelo G4MollerBhabhaModel.
 - *Bremsstrahlung:* Es del tipo G4eBremsstrahlung, utiliza el modelo G4SeltzerBerger Model para energías por debajo de 1GeV y el modelo G4eBremsstrahlungRelModel para energías mayores.
 - *Dispersión múltiple:* Es del tipo G4VMultipleScattering, utiliza el modelo G4UrbanMsc Model96.

- *Fotón óptico:*
 - *Centelleo:* El del tipo G4Scintillation, este proceso es activado por todas las partículas, excepto aquellas que tienen vida corta.
 - *Absorción óptica:* Es del tipo G4OpAbsorption.
 - *Rayleigh óptico:* Es del tipo G4OpRayleigh.
 - *Proceso óptico de frontera:* Es del tipo G4OpBoundaryProcess, sin un modelo unificado.

4.5.1. Producción de cortes

Varios procesos físicos, como bremsstrahlung, ionización de cualquier partícula cargada y producción de pares de muones tienen secciones transversales grandes a bajas energías. Por lo tanto, es necesario implementar una producción de cortes de manera que todas las partículas debajo de esta energía no sean generadas, pero la información de su energía sí es contada como energía depositada. Geant4 implementa la producción de cortes en rangos, por ejemplo, 1mm para fotones significa que no se producirán fotones si el rango esperado en el material es menor a 1mm .

Si se utiliza la lista de física electromagnética de GAMOS, el valor por defecto para la producción de cortes es 0.1mm , para todos los procesos en todos los materiales. Es importante configurar la producción de cortes antes de utilizar el comando `/run/initialize`, si se desea que el cambio se haga efectivo. Para obtener la lista de materiales y cortes por material, luego de correr la simulación es posible encontrarla en `gamos.log`, con el título de “*Table of registered couples*”, un ejemplo de cómo se encuentran los datos se presenta a continuación.

```

Index:0                used in the      geometry:Yes    recalculation   needed:No
Material                :G4_AIR
Range cuts:            gamma 100 um    e- 100 um      e+ 100 um      proton 100 um
Energy thresholds:     gamma 990 eV    e- 990 eV      e+ 990 eV      proton -1 MeV
Region(s)              which           use            this           couple:
DefaultRegionForTheWorld

```

4.5.1.1. Producción de cortes por región

Una región en Geant4 es un conjunto de G4LogicalVolume's que tienen propiedades en común. Se puede definir una región en el archivo de texto donde se encuentra la geometría utilizando la etiqueta `:REGION NOMBRE_REGION NOMBRE_VOLUMENES_LOGICO`, donde `NOMBRE_REGION` es el nombre que define la región y `NOMBRE_VOLUMENES_LOGICO` es la lista de G4LogicalVolume's que pertenecen a la región. Por ejemplo: `:REGION miRegion Cristal Pared`

Otra alternativa es definir las regiones en el archivo principal del programa o main, utilizando el comando `/gamos/geometry/createRegion NOMBRE_REGION NOMBRE_VOLUMEN_LOGICO`. Es importante tener en cuenta que en Geant4 cuando un volumen lógico pertenece a una región automáticamente todos los volúmenes hijos pertenecen a la misma región, a menos que haya otra región explícitamente definida para alguno de los volúmenes hijos. Además, una región en Geant4 tiene que ser definida de forma jerárquica, si se coloca un volumen A en el mundo y dentro de el se coloca un volumen B, no se puede crear una región en B a menos que ya se haya creado una región específica para A.

Luego de definir las regiones, se puede configurar la producción de cortes en el archivo de donde se encuentra la geometría utilizando la etiqueta `:CUTS NOMBRE_REGION gamma_CORTE e-_CORTE e+_CORTE`, donde `NOMBRE_REGION` es el nombre de la región que ya se definió, `gamma_CORTE`, `e-_CORTE` y `e+_CORTE` son los cortes para las partículas gamma, electrones y positrones respectivamente. El corte para los positrones es opcional, si no se coloca un valor explícito, GAMOS tomará el valor configurado para los electrones. Además, se puede realizar esta tarea en el programa principal utilizando el comando `/gamos/physics/setCuts NOMBRE_REGION gamma_CORTE e-_CORTE e+_CORTE`.

4.5.1.2. Producción de cortes por energía

Para generar los cortes por energía, es necesario inicializar la acción de usuario `/gamos/userAction GmCutsEnergy2RangeUA`, en el archivo principal del programa. Luego utilizando el comando `/gamos/physics/ECuts2RangeCuts NOMBRE_MATERIAL VALOR_CORTE NOMBRE_PARTICULA`.

4.6 Verbosidad

Al correr una simulación en GAMOS, es posible controlar la cantidad de información que se obtiene en la pantalla y en los archivos de registro, con el administrador de verbosidad de GAMOS. Existen seis niveles de verbosidad, cada nivel incluye la verbosidad del nivel anterior.

- *Silencio* (`=-1`): ninguna información de la simulación es impresa, solo si hay una excepción o error en la simulación y esta se detiene, se obtendrá información detallada de la causa.

- *Error (=0)*: solo se imprimen mensajes de error.
- *Advertencia (= 1)*: solo mensajes de error y advertencia son impresos.
- *Información (=2)*: se imprime la información detallada de lo que está pasando durante la compilación de la simulación. Principalmente mensajes en cada corrida y en cada evento.
- *Depuración (=3)*: se imprime la información detallada de lo que está pasando. Principalmente mensajes en cada trayectoria y en cada paso.
- *Prueba (=4)*: este nivel solo se utiliza para probar el código la primera vez que se compila.

La verbosidad de GAMOS esta clasificada por diferentes tipos, de modo que se pueden configurar diferentes opciones de verbosidad para diferentes partes del código. Algunas de las partes en las que tienen verbosidad independiente son las siguientes:

- *GmBaseVerbosity*: controla la verbosidad las clases base de GAMOS, como: filtros, clasificadores, administrador de archivos de entrada y salida.
- *GmGeomVerbosity*: controla la verbosidad de las clases de geometría.
- *GmGenerVerbosity*: controla la verbosidad del generador GAMOS.
- *GmPhysicsVerbosity*: controla la verbosidad de las clases de física electromagnética de GAMOS.
- *GmUAVerbosity*: controla la verbosidad de las clases que administran las acciones de usuario.
- *GmDataVerbosity*: controla la verbosidad de los datos de GAMOS.
- *GmAnalysisVerbosity*: controla la verbosidad de las clases de análisis.

4.7 Optimización del CPU

4.7.1.Límites del usuario

Este es un mecanismo que Geant4 ofrece para limitar la trayectoria de una partícula. Existen cinco diferentes formas para poder realizar esto:

- Limitando el tamaño de paso.
- Limitando el tamaño de la trayectoria
- Limitando el tiempo de vuelo
- Deteniendo la partícula cuando la energía cinética está por debajo del límite y deposita su energía localmente.

- Deteniendo la partícula cuando el rango de energía esperado esta por debajo del límite y deposita su energía localmente.

Existen comandos para realizar todas estas limitaciones juntas o individuales. Estos se enlistan a continuación.

- */gamos/physics/userLimits/setUserLimits* NOMBRE_LIMITE_USUARIO NOMBRE_VOLUMEN_LOGICO NOMBRE_PARTICULA MAX_PASOS MAX_LONGITUD_TRAYECTORIA MAX_TIEMPO_VUELO MIN_ENERGIA_CINETICA RANGO_MIN_ENERGIA
- */gamos/physics/userLimits/setMaxStep* NOMBRE_LIMITE_USUARIO NOMBRE_VOLUMEN_LOGICO NOMBRE_PARTICULA MAX_PASOS
- */gamos/physics/userLimits/setMaxTrkLen* NOMBRE_LIMITE_USUARIO NOMBRE_VOLUMEN_LOGICO NOMBRE_PARTICULA MAX_LONGITUD_TRAYECTORIA
- */gamos/physics/userLimits/setMaxTOF* NOMBRE_LIMITE_USUARIO NOMBRE_VOLUMEN_LOGICO NOMBRE_PARTICULA MAX_TIEMPO_VUELO
- */gamos/physics/userLimits/setMinEKin* NOMBRE_LIMITE_USUARIO NOMBRE_VOLUMEN_LOGICO NOMBRE_PARTICULA MIN_ENERGIA_CINETICA
- */gamos/physics/userLimits/setMinRange* NOMBRE_LIMITE_USUARIO NOMBRE_VOLUMEN_LOGICO NOMBRE_PARTICULA RANGO_MIN_ENERGIA

Donde NOMBRE_LIMITE_USUARIO es el nombre que el usuario le da a la función, NOMBRE_VOLUMEN_LOGICO es el nombre del volumen lógico ya definido, NOMBRE_PARTICULA es el nombre de la partícula, MAX_PASOS es el número máximo de pasos, MAX_LONGITUD_TRAYECTORIA es la longitud máxima de la trayectoria, MAX_TIEMPO_VUELO es el tiempo máximo de vuelo, MIN_ENERGIA_CINETICA es la energía cinética mínima de la partícula, RANGO_MIN_ENERGIA es el rango de energía mínimo de la partícula.

4.7.2. Eliminando partículas

Existen dos acciones de usuario que puede ser combinadas con filtros para poder eliminar las partículas que el usuario sabe que no están afectando la simulación, ni los resultados de una manera sensible. Estos son *GmKillAtStackingActionUA* y *GmKillAtSteppingUA*. Un ejemplo de como combinar estos comandos con filtros es: */gamos/filter* NOMBRE_FILTRO *GmSecondaryFilter* y el siguiente comando es */gamos/userAction* *GmKillAtStackingActionUA* NOMBRE_FILTRO, este ejemplo elimina todas las partículas secundarias que se generen en la simulación.

5 Simulación

5.1 Caracterización de la geometría

La simulación recrea computacionalmente un entorno real, se busca igualar las condiciones físicas de este entorno para alcanzar un grado de validez aceptable en los resultados de la simulación. Para poder realizar esta tarea, es necesario caracterizar de la mejor manera posible las condiciones del entorno donde se llevó a cabo el experimento e igualarlas a las condiciones simuladas computacionalmente. Para esto se tomaron en cuenta las características del Laboratorio Secundario de Calibración Dosimétrica (LSCD) y el fantoma utilizado.

Todo lo desarrollado a continuación se realiza en un archivo de texto aparte llamado “ContruendoGeometria.geom”, el cual llama al archivo principal de la simulación a través del comando `/gamos/setParam GmGeometryFromText:FileName ContruendoGeometria.geom`. El cual indica que la geometría se encuentra en un archivo de texto en el directorio, llamado `ContruendoGeometria.geom`

5.1.1.Materiales

GAMOS provee un archivo de texto donde se encuentran una lista de materiales ya definidos utilizando los estándares del Instituto Nacional de estándares y Tecnología (NIST), que pueden ser de utilidad para la simulación. Utilizando esta lista se agregaron dos materiales necesarios para caracterizar la geometría. El primero de ellos es el polimetacrilato de metilo o comúnmente conocido como acrílico.

```
:MIXT      "NIST_Polymethyl Methacrylate"  1.19      3
H           0.080541
C           0.599846
O           0.319613
:MATE_MEE  "NIST_Polymethyl Methacrylate"  74 * eV
```

:MIXT es la etiqueta que se utiliza para indicarle a GAMOS que se va definir una material compuesto por una mezcla de varios elementos, luego sigue el nombre "NIST_Polymethyl Methacrylate", la densidad (1.19) y el número de elementos o componentes (3). En las siguientes líneas se tabulan a detalle el elemento y la proporción de este en la mezcla, en este caso tenemos: hidrógeno (H) con proporción 0.080541, carbono (C) con proporción 0.599846, oxígeno (O) con proporción 0.319613.

:MATE_MEE es la etiqueta que se utiliza para especificar propiedades del material ya definido, "NIST_Polymethyl Methacrylate" es el nombre de la mezcla que ya se definió y la energía de excitación promedio ($74 * eV$). El símbolo (*) simboliza el cambio de dimensionales a eV . El segundo material que se agregó es el "Barite" o conocido también como concreto.

```

:MIXT      "NIST_Concrete, Barite (TYPE BA)"  3.35      9
H          0.003585
O          0.311622
Mg         0.001195
Al         0.004183
Si         0.010457
S          0.107858
Ca         0.050194
Fe         0.047505
Ba         0.4634
:MATE_MEE  "NIST_Concrete, Barite (TYPE BA)"  248.2*eV

```

:MIXT es la etiqueta que se utiliza para indicarle a GAMOS que se va definir una material compuesto por una mezcla de varios elementos, luego sigue el nombre "NIST_Concrete, Barite (TYPE BA)", la densidad (3.35) y el número de elementos o componentes (9). En las siguientes líneas se tabulan a detalle el elemento y la proporción de este en la mezcla, en este caso tenemos: hidrógeno (H) con proporción 0.003585, oxígeno (O) con proporción 0.311622, (Mg) con proporción 0.001195, (Al) con proporción 0.004183, (Si) con proporción 0.010457, (S) con proporción 0.107858, (Ca) con proporción 0.050194, (Fe) con proporción 0.047505, (Ba) con proporción 0.4634.

:MATE_MEE es la etiqueta que se utiliza para especificar propiedades del material ya definido, "NIST_Concrete, Barite (TYPE BA)" es el nombre de la mezcla que ya se definió y la energía de excitación promedio ($248.2 * eV$).

5.1.2. Matriz de rotación

En el capítulo anterior, se detallaron varias de las funcionalidades de GAMOS para poder crear geometrías. Uno de los parámetros necesarios para varias de las funciones detalladas, es la matriz de rotación. Aunque no se utilice específicamente la matriz de rotación en los elementos de geometría creados es necesario definir una. La matriz de rotación se definió utilizando la etiqueta :ROTM RM0 0. 0. 0. Donde RM0 es el nombre que se le asignó a la matriz de rotación, para utilizarla en otros comandos más adelante. Los parámetros 0. 0. 0. son correspondientes a los ángulos medidos desde el eje X, Y, Z respectivamente. Se definió la matriz de rotación de esta forma, debido a que no fue necesario aplicar ninguna rotación a los sólidos y volúmenes creados en la geometría.

5.1.3. Volumen madre o primario

El compilador de GAMOS construye geometrías utilizando una jerarquía en los sólidos o volúmenes que se van definiendo. Al volumen base se le llama Volumen Madre (“Parent Volume”) o Volumen primario (“Primary Volume”). Para poder indicar el volumen primario se le llama a este volumen *world*. La geometría del LSCD donde se lleva a cabo la simulación es una cámara llena de aire, reforzada de paredes de concreto. Por eso el volumen primario que se utilizó para poder caracterizar este laboratorio es un volumen de aire con las dimensiones de la cámara, este volumen se definió utilizando el comando :VOLU world BOX 2750 3750 1750 G4_AIR.

La etiqueta :VOLU indica que se definirá un volumen, con nombre world, el tipo BOX indica que es en forma de caja, con dimensiones 2750 3750 1750. Estas dimensiones están en milímetros, e indican la mitad de la longitud de la caja en el eje X 2750 mm, la mitad de la longitud de la caja en el eje Y 3750 mm, la mitad de la longitud de la caja en el eje Z 1750 mm. Lo que se traduce a una cámara con 5.5m de ancho *7.5m de fondo *3.5m de alto. G4_AIR es la molécula o elemento que compone el volumen.

5.1.4. Paredes de concreto

Las paredes que rodean el LSCD deben ser de un grosor considerable por el tipo de experimentos que se realizan adentro con la fuente de radiación. El grosor de las cuatro paredes del laboratorio es de 38cm, el grosor del techo es de 55cm. El grosor del piso no se conocía con precisión, así que se tomó que este tenía el mismo grosor del techo de 55cm. Los comandos utilizados para definir estas paredes son:

```
:SOLID ConcretoExt BOX 3130. 4130. 2300.  
:SOLID ConcretoInt BOX 2750. 3750. 1750.  
:VOLU ParedConcreto SUBTRACTION ConcretoExt ConcretoInt RM0 0. 0. 0.  
"NIST_Concrete, Barite (TYPE BA)"  
:COLOR ParedConcreto 1 1 1 1  
:PLACE ParedConcreto 1 world RM0 0. 0. 0.
```

Se define los sólidos en forma de caja (BOX) con nombres ConcretoExt y ConcretoInt, el primero con las dimensiones del volumen de aire “world” más las dimensiones del grosor 3130. 4130. 2300. El segundo con las dimensiones exactas del volumen de aire “world” 2750. 3750. 1750. Luego se define un volumen llamado ParedConcreto, que es igual a la diferencia (“SUBTRACTION”) entre el sólido ConcretoExt y el sólido ConcretoInt con matriz de rotación RM0, se ubica ambos sólidos en un origen (0. 0. 0.), para que la SUBTRACTION sea simétrica. Este nuevo volumen está compuesto del material definido en la sección anterior “NIST_Concrete, Barite (TYPE BA)”

Se utiliza la etiqueta :COLOR ParedConcreto 1 1 1 0 para indicar que el color del volumen

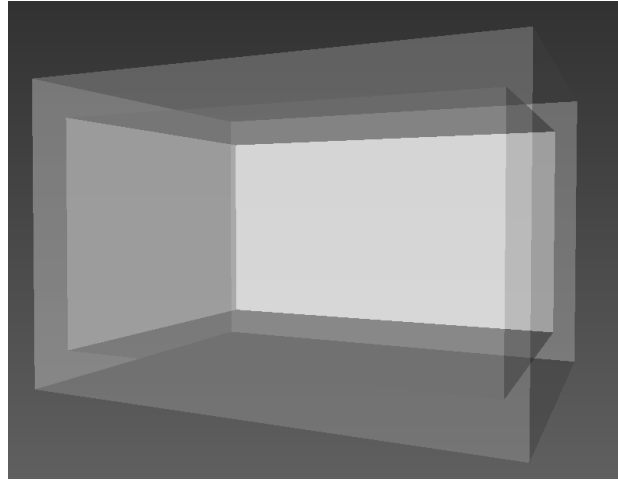


Figura 5.1: Geometría de las paredes de concreto LSCD

ParedConcreto con proporción máxima del color rojo, verde y azul, para conseguir un color blanco o grisáceo, indicado por los primeros tres parámetros. Visibilidad máxima, en este caso 0, indicado por el último parámetro. Por último, se utiliza la etiqueta :PLACE ParedConcreto 1 world RM0 0. 0. 0. para indicar que se posiciona 1 copia del volumen ParedConcreto en el volumen world, con la matriz de rotación RM0, en la posición 0. 0. 0. en X, Y, Z.

5.1.5.Fantoma

El fantoma es una pieza clave para la simulación, este busca simular el torso de un humano, para determinar la dosis de radiación que este recibiría si es expuesto a una fuente de radiación. (José and Ché, 2018) El fantoma está compuesto por una caja de acrílico de 1cm de grosor por los lados, 0.5cm de grosor por la parte de enfrente y de atrás, y 1cm de grosor en la tapa y la base, cuyo interior está lleno de agua. Primero se define la caja de acrílico utilizando la misma técnica que en la sección anterior para crear las paredes de concreto. Los comandos utilizados fueron los siguientes.

```
:SOLID CajaAcrylicaEXT BOX 150. 75. 150.
:SOLID CajaAcrylicaINT BOX 140. 70. 140.
:VOLU CajaFantoma SUBTRACTION CajaAcrylicaEXT CajaAcrylicaINT RM0 0. 0. 0.
"NIST.Polymethyl Methacrylate"
:PLACE CajaFantoma 1 world RM0 0. 280. 700.
```

En las primeras dos líneas se utiliza la etiqueta :SOLID, para definir dos sólidos en forma de caja (BOX), el primero con nombre CajaAcrylicaEXT y dimensiones 150. 75. 150. correspondientes a la mitad de longitud en X, Y, Z del exterior del fantoma. El segundo con nombre CajaAcrylicaINT y dimensiones 150. 75. 150. correspondientes a la mitad de longitud en X, Y, Z del interior del

fantoma.

Luego se utiliza la etiqueta :VOLU para crear un volumen con nombre CajaFantoma, utilizando la función SUBTRACTION con el sólido CajaAcrylicaEXT y CajaAcrylicaINT, aplicando la matriz de rotación RM0. Para que esta sustracción sea simétrica se indica que ambos volúmenes estén posicionados en el origen con los parámetros 0. 0. 0.. El último parámetro indica la composición de la caja de acrílico, donde se utiliza uno de los materiales definidos en la sección anterior "NIST_Polymethyl Methacrylate".

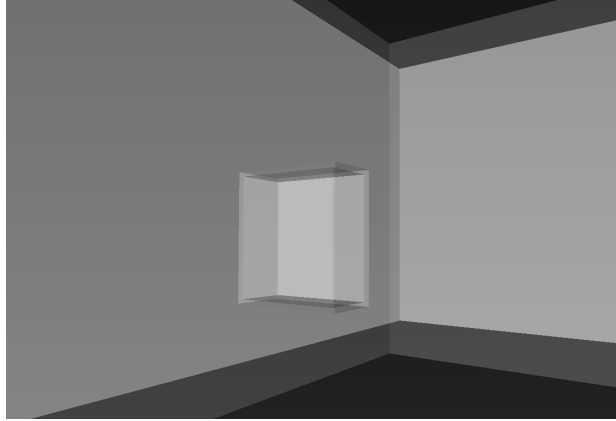


Figura 5.2: Geometría de la caja del fantoma de acrílico

Por último, se posiciona 1 copia de el volumen CajaFantoma, utilizando la etiqueta :PLACE, en el volumen madre world, con matriz de rotación RM0 en la posición 0. 280. 700.. A continuación se detalla los cálculos necesarios para explicar la procedencia de los datos de posición utilizando notación vectorial. El volumen madre está posicionado en la coordenadas $(0, 0, 0)$. Esto quiere decir que las paredes en "X" se encuentran a $(2750, 0, 0)$ y $(-2750, 0, 0)$, en "Y" se encuentran a $(0, 3750, 0)$ y $(0, -3750, 0)$, en "Z" se encuentran a $(0, 0, 1750)$ y $(0, 0, -1750)$. Para ubicar el fantoma es necesario saber que la fuente de radiación se encuentra a una distancia de 470mm de la pared "XZ" o pared frontal, a una distancia de 1050mm del suelo y en la coordenada 0 del eje "X", ósea alineado al eje "X" del plano coordenado ubicado en el punto $(0, 0, 0)$ del laboratorio que estamos definiendo.

Sabiendo esta información y que el fantoma que se está simulando se encuentra a 3m de la fuente de radiación, podemos obtener las coordenadas $(0, 280, 700)$. Sabemos que el fantoma está centrado con la fuente respecto al eje X, de modo que no hay variación en esta coordenada. Sabemos que la pared Y se encuentra a 3750mm del centro, de modo que el fantoma se encuentra a:

$$3750\text{mm} - 470\text{mm} - 3000\text{mm} = 280\text{mm} \quad (5.1)$$

Donde 470mm es la distancia de la pared XZ a la fuente, y 3000mm es la distancia de la fuente al

fantoma. De igual manera se obtiene la coordenada en Z.

$$1750mm - 1050mm = 700mm \quad (5.2)$$

Donde $1050mm$ es la distancia del suelo a la fuente, y el fantoma se encuentra a la misma altura de la fuente, medida desde el centro del mismo. A partir de estos cálculos se ubica el fantoma en la coordenada $(0, 280, 700)$.

5.1.6. Volumen de agua del fantoma y voxels

Luego de definir la caja de acrílico del fantoma se define el volumen interno de agua, el cual se dividirá en voxels o cubos de agua más pequeños para poder medir la dosis en cada uno de estos. El número de voxels es arbitrario, se dividió el volumen del agua en una malla de mil voxels. El comando utilizado para definir este volumen de agua es el siguiente.

```
:VOLU contenedor_fantoma BOX 140. 70. 140. G4_WATER
:PLACE contenedor_fantoma 1 world RM0 0. 280. 700.
:COLOR contenedor_fantoma 0 0 1 1
:VOLU fantoma BOX 14.0 70.0 14.0 G4_WATER
:PLACE_PARAM fantoma 1 contenedor_fantoma PHANTOM 10 10 10 28. 14. 28.
```

Primero se utiliza la etiqueta `:VOLU` para definir el volumen con nombre `contenedor_fantoma` en forma de caja (BOX) con dimensiones $140. 70. 140.$, que son las dimensiones interiores de la caja de acrílico definida en la sección anterior. El último parámetro indica la composición del volumen `G4_WATER` que en este caso es de agua.

En la segunda línea se utiliza la etiqueta `:PLACE` para ubicar una copia el volumen `contenedor_fantoma` en el volumen `world` con matriz de rotación `RM0` en las coordenadas $0. 280. 700.$. Se utiliza la etiqueta `:COLOR` para darle al volumen `contenedor_fantoma` color azul, con el parámetro $0 0 1$ y visibilidad 1 .

Se utiliza la etiqueta `:VOLU` para definir el primer voxel, llamado `fantoma`, con forma de caja (BOX) con dimensiones $14.0 7.0 14.0$ que en este caso con el 10 veces más pequeño del tamaño del volumen `contenedor_fantoma` y está compuesto de agua `G4_WATER`.

Por último, se utiliza la etiqueta `:PLACE_PARAM` que sirve para ubicar un volumen con parámetros extra. Se ubica 1 copia del último volumen creado llamado `fantoma`, en el volumen `contenedor_fantoma`, el parámetro `PHANTOM` sirve para especificar que el volumen que se está ubicando son los voxels de un fantoma. El siguiente parámetro sirve para identificar cuantos voxels se ubicarán en cada dimensión $10 10 10$, en este caso son 10 en $X * 10$ en $Y * 10$ en Z . Y los últimos parámetros son los espacios entre las copias, para que el volumen se ajuste $28. 14. 28.$ en este caso coincide con ser el doble de las dimensiones de cada voxel.

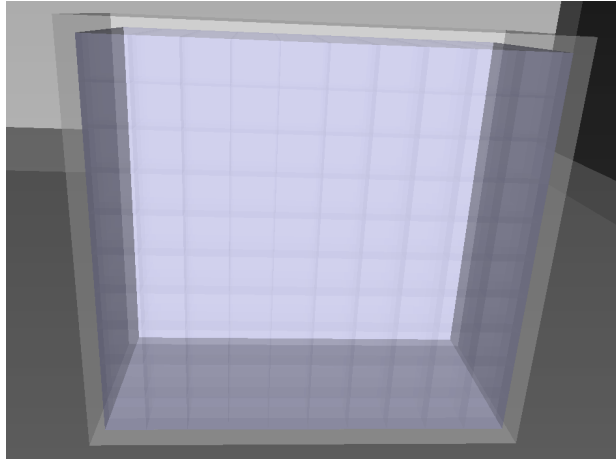


Figura 5.3: Geometría del fantoma con caja de acrílico, dividida por voxels de agua

5.2 Archivo principal

Las siguientes secciones detallan las líneas de código utilizadas para hacer la simulación y en el orden que se utilizaron, es importante tomar en cuenta el orden lógico para que la simulación no tenga excepciones o errores al compilar.

5.2.1. Verbosidad

Para poder desarrollar la simulación en el menor tiempo posible se recomienda configurar la verbosidad en *silent* (= -1), esto indica al programa que no se desea tener el resumen detallado de cada paso de la simulación, sino que solamente se desea saber si hay una excepción en la compilación o error. Si se está realizando pruebas GAMOS, se recomienda utilizar la verbosidad en *debug* (= 1). Los otros niveles de verbosidad y sus características se detallan en el capítulo anterior. El comando para poder indicar que verbosidad se desea utilizar es `/tracking/verbose -1`. En este caso está configurado para que la verbosidad esté en *silent*.

5.2.2. Geometría desde un archivo de texto

Para llamar la geometría desde un archivo de texto se necesitan tres líneas de comandos. La primera de ellas es `/gamos/geometry GmGeometryFromText` la cual sirve para indicarle a GAMOS que la geometría es proveniente de un archivo de texto.

El segundo comando necesario es `/gamos/setParam GmGeometryFromText:FileName ContruyendoGeometria.geom`, donde se establece como parámetro de la línea anterior que el archivo de texto donde se extrae la geometría tiene el siguiente nombre “ContruyendoGeometria.geom”.

El tercer comando necesario es `/gamos/setParam GmReadPhantomGeometry:Phantom: FileName ContruyendoGeometria.geom`, este parámetro indica que el fantoma que se va a utilizar en

la simulación se encuentra en el archivo “ContruyendoGeometria.geom”. En el caso en el que el fantoma se haya construido en un archivo de texto separado el nombre de la geometría sería diferente. En este caso se construyó el fantoma en el mismo archivo que el resto de la geometría, ya que no era una geometría compleja. Si no se encuentra en el archivo “ContruyendoGeometria.geom” un comando donde se establezca un objeto como fantoma utilizando el comando :PLACE_PARAM con parámetro PHANTOM el programa devolverá un error.

5.2.3. Comandos generales

Es necesario indicarle a GAMOS que lista de procesos físicos electromagnéticos se desea utilizar, en este caso se utiliza la lista de procesos simples, para configurar esto se utiliza el comando `/gamos/physicsList GmEMPhysics` donde *GmEMPhysics* es la lista de proceso electromagnéticos que se estará utilizando. En cualquier simulación de GAMOS, es necesario añadir siempre este comando, de otro modo la simulación siempre devolverá un error.

5.2.3.1. Producción de cortes para procesos físicos.

La producción de cortes es una herramienta de optimización orientada a los procesos físicos de GAMOS. Los valores necesarios para realizar los cortes se pueden obtener luego de correr la simulación en el archivo *gamos.log*. En el anexo (8.1) se adjunta los datos obtenidos de este archivo.

El primer paso para crear los cortes es crear regiones utilizando el comando `#/gamos/geometry/createRegion`. En este caso se crean cuatro regiones, la pared de concreto, caja de acrílico del fantoma, el volumen de agua y el volumen de aire. Donde el primer parámetro indica el nombre de la región (*Pared*, *FantomaAcrilico*, *FantomaAgua*, *Aire*), el segundo parámetro indica el o los volúmenes lógicos con los que se crea la región (*ParedConcreto*, *CajaFantoma*, *contenedor_fantoma*, *world*).

```
#/gamos/geometry/createRegion Pared ParedConcreto
#/gamos/geometry/createRegion FantomaAcrilico CajaFantoma
#/gamos/geometry/createRegion FantomaAgua contenedor_fantoma
#/gamos/geometry/createRegion Aire world
```

Luego de crear las regiones se crean los cortes para los diferentes materiales utilizando el comando `#/gamos/physics/setCuts`. El primer parámetro corresponde al nombre de la región creada, el segundo parámetro al corte para las partículas gamma, el segundo para los electrones, el tercero para los positrones y el cuarto para los protones.

```
#/gamos/physics/setCuts Pared 100. * um 100. * um 100. * um 100. * um
#/gamos/physics/setCuts FantomaAcrilico 100. * um 100. * um 100. * um 100. * um
#/gamos/physics/setCuts FantomaAgua 100. * um 100. * um 100. * um 100. * um
#/gamos/physics/setCuts Aire 100. * um 100. * um 100. * um 100. * um
```

Para producir los cortes para energía se agrega el comando `/gamos/userAction GmCutsEnergy2RangeUA`. Luego se utiliza el comando `#/gamos/physics/ECuts2RangeCuts` para producir los cortes de energía para cada material. El primer parámetro indica la región a la cual se le hace el corte, el segundo la energía y el tercero la partícula a la cual se le hace el corte. En este caso solo se realiza el corte para partículas gamma, ya que las partículas secundarias y electrones son eliminados, en la sección de filtros se detalla más de esta parte.

```
#/gamos/physics/ECuts2RangeCuts G4_AIR 990. * eV gamma
#/gamos/physics/ECuts2RangeCuts "NIST.Concrete, Barite (TYPE BA)" 5.31198 * keV
gamma
#/gamos/physics/ECuts2RangeCuts "NIST.Polymethyl Methacrylate" 1.1173 * keV gamma
#/gamos/physics/ECuts2RangeCuts G4_WATER 1.10981 * keV gamma
```

5.2.4. Caracterización de la fuente

La fuente de radiación que se está simulando es una fuente de radiación ionizante de Cesio 137, esta fuente se produce por fisión nuclear, la energía de los fotones que libera es de 660 keV. Esta fuente tiene una forma de cilindro con dimensiones 6.25mm de radio y 8.95mm de alto. Para configurar el generador de GAMOS con estas características se necesitan tres comandos.

El primer comando es `/gamos/generator/addIsotopeSource fuenteIO Cs137 660. * keV`, este comando indica que se agrega un isótopo como fuente de radiación, con el nombre `fuenteIO`. El isótopo es Cs137, este es extraído de la lista de isótopos radioactivos que provee GAMOS, la cual se encuentra en el directorio `GAMOS.5.0.0/data/isotopes.dat`. El último parámetro es la energía de los fotones que libera la fuente, en este caso sabemos que es 660. * keV, se agrega *keV para indicar las dimensionales de energía, ya que las que trae por defecto son MeV.

El segundo comando es `/gamos/generator/positionDist fuenteIO GmGenerDistPositionPoint 0 3280 700`, este comando indica la posición en la que el usuario desea ubicar la fuente, el primer parámetro es el nombre de la fuente que se desea posicionar, en este caso `fuenteIO`. El segundo parámetro `GmGenerDistPositionPoint` indica que se desea ubicar la fuente de radiación en un punto específico el cuál está dado por el último parámetro 0 3280 700

El cálculo de la coordenada Y se obtiene de la siguiente forma.

$$3750mm - 470mm = 3280mm \quad (5.3)$$

Donde 3750mm es la distancia del centro a la pared XZ, y 470mm es la distancia de la pared a la fuente.

El tercer comando es muy importante, ya que este comando nos ahorra tiempo de cómputo y simulación. Este comando indica en que distribución el usuario desea que tenga su fuente de

radiación y en que dirección esté esta distribución. Si se simulara la fuente normal, esta irradiaría en todas las direcciones aleatoriamente, y sería necesario crear una geometría de haz estrecho para poder direccionar la radiación en la dirección deseada.

El comando `/gamos/generator/directionDist fuenteIO GmGenerDistDirectionCone 0. -1. 0. 0.155862` nos permite configurar la generación de partículas a una distribución de dirección que en este caso tiene forma de cono, simulado la funcionalidad de una geometría de haz estrecho. El primer parámetro `fuenteIO` indica a que fuente de radiación se le quiere asignar la distribución de dirección, el segundo parámetro indica la forma de la distribución, para que tenga forma de cono se utiliza `GmGenerDistDirectionCone`, los tres parámetros siguientes indican la dirección en la que el usuario desee el abra el cono, `0. -1. 0.`, en este caso la fuente se encuentra posicionada en la coordenada $(0, 3280, 700)$ y el fantoma se encuentra en la coordenada $(0, 2280, 700)$, por lo tanto la dirección de la distribución cónica debe ser hacia en la dirección Y negativa. El último parámetro es el doble del ángulo sólido con que se desea que abra el cono.

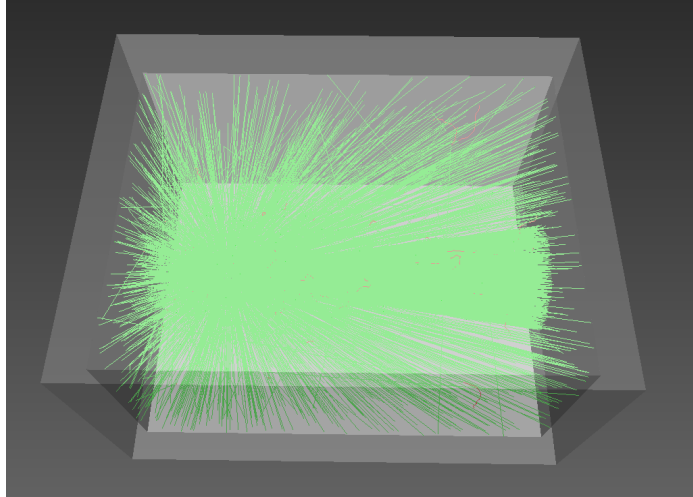


Figura 5.4: Simulación de 3000 partículas generadas por el generador GAMOS

Para el cálculo del ángulo sólido es necesario saber que el diametro de apertura del colimador de haz estrecho a un metro es 31.5cm . La distancia de la fuente de radiación de Cs137 a él fantoma es 3m . Y el área de un círculo está dada por la fórmula $\pi * r^2$. El segmento de ángulo sólido está dado por la ecuación (4.4).

$$\Omega = \frac{S}{R^2} = \frac{\pi r^2}{R^2} = \frac{\pi \left(\frac{31.5/2\text{cm}}{2}\right)^2}{(100\text{cm})^2} = 0.0779311\text{sr} \quad (5.4)$$

Donde S es el área del círculo y R es el radio del centro de la esfera al punto donde se desea tener el área S de apertura. Este resultado se multiplica por 2, ya que el parámetro de ángulo sólido toma la mitad del valor que se le ingrese, por lo tanto el parámetro ingresado es $2 * 0.0779311\text{sr} = 0.155862$.

5.2.5. Árbol de volúmenes

GAMOS provee una herramienta que permite obtener información acerca de los volúmenes creados, en este caso necesitaremos la masa del volumen fantoma que GAMOS utiliza para poder obtener el valor de la dosis. Para obtener esta información se utiliza el comando `/gamos/geometry/printVolumeTree 2`. En el anexo (8.4) está la información que GAMOS devuelve al colocar este comando en el archivo `gamos.log`.

5.2.6. Filtros y límites de usuario para optimizar CPU

Cuando las partículas primarias interactúan con la materia a través de ciertos procesos, como el efecto Compton o el efecto fotoeléctrico, se generan partículas secundarias y/o electrones. Los cuales no aportan información de peso a la variable que se desea medir, pero consumen memoria del CPU, que limita la eficiencia de la simulación. Para esto se agregan dos filtros llamados `AcceptStepP` y `AcceptStepS` cuya función es atrapar las partículas secundarias y electrones creadas por la simulación y junto con la acción de usuario `GmKillAtStackingActionUA` eliminan estas partículas de la simulación.

El comando para aplicar el filtro para partículas secundarias es `/gamos/filter AcceptStepS GmSecondaryFilter`, donde el primer parámetro es el nombre del filtro y el segundo parámetro especifica que partículas se desea filtrar. Luego la acción de usuario `/gamos/userAction GmKillAtStackingActionUA AcceptStepS` aplicada al filtro `AcceptStepS` elimina todas estas partículas.

Utilizando la misma lógica se aplica el filtro `/gamos/filter AcceptStepP GmElectronFilter` a electrones y se eliminan estas partículas con el comando `/gamos/userAction GmKillAtStackingActionUA AcceptStepP`. Muchas de las partículas gamma generadas por el generador GAMOS, pasan de largo el fantoma sin interactuar o salen del fantoma luego de haber interactuado, estas partículas crean trayectorias largas que no aportan información a la variable de medición y consumen memoria del CPU. Para poder limitar estas partículas se utilizó un límite de usuario.

El límite de usuario es `/gamos/physics/userLimits/setMaxTrkLen LimiteR world gamma 3.e+03` este límite de usuario, con nombre `LimiteR`, restringe la longitud de la trayectoria de una partícula gamma a 3×10^3 , en el volumen `world`. Esto elimina todas las trayectorias de las partículas luego de salir del fantoma en diferentes direcciones y elimina las partículas que son desviadas al interactuar con partículas de aire antes de llegar al fantoma. Esto ayuda a reducir la cantidad de memoria que necesita el CPU para procesar cada partícula y hace más eficiente el código.

Luego de aplicar los filtros y el límite de usuario, se elimina bastante información que no es relevante para la simulación y se libera la memoria del CPU. En la Figura (4.5), se puede observar el resultado de haber aplicado estas restricciones, las trayectorias rojas, correspondientes a partículas secundarias que se generan luego de procesos electromagnéticos son eliminadas por

los filtros. Luego las trayectorias verdes que dejan el fantoma y que son desviadas también son eliminadas.

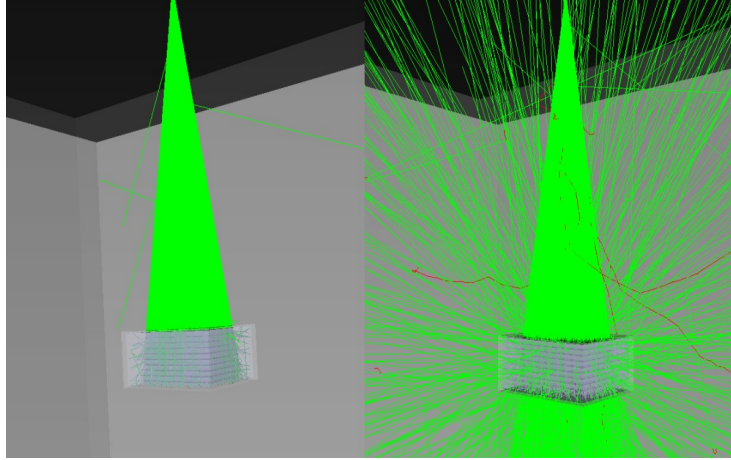


Figura 5.5: Comparación entre simulación con filtros y límites de usuario / simulación sin filtros y límites de usuario

5.2.7. Dosis en el fantoma

Para poder medir la energía y dosis depositada en el fantoma por partícula es necesario crear un detector, agregarle un contador al detector y agregar algún tipo de impresora para poder extraer los datos. En este caso se utilizarán tres tipos de impresoras distintas *GmPSPrinter3ddose* para poder imprimir la dosis que recibe cada voxel por partícula, *RTPSPDoseHistos* para poder imprimir histogramas en formato “.root” y *GmPSPrinterTextFile* para poder imprimir la energía depositada por partícula en cada voxel.

El primer comando utilizado es `/gamos/scoring/createMFDetector DoseDet fantoma`, este comando sirve para crear un detector llamado *DoseDet* y se le asigna a un volumen lógico ya definido, en este caso el volumen de agua que contienen nuestro fantoma se llama *fantoma*.

El segundo comando es `/gamos/scoring/addScorer2MFD DoseScorer GmG4PSDoseDeposit DoseDet`, este comando, con nombre *DoseScorer*, sirve para añadir un contador al detector *DoseDet* creado en el comando anterior. La cantidad que deseamos que mida es la dosis, y para configurar el contador para que mida esta cantidad se agrega el segundo parámetro *GmG4PSDoseDeposit*.

El tercer y cuarto comando es para agregar las impresoras que el usuario desea para obtener los datos de la simulación. El comando de la primera impresora es `/gamos/scoring/addPrinter2Scorer GmPSPrinter3ddose DoseScorer`, tiene como función devolver un archivo de texto con la dosis en cada uno de los voxels llamado “3ddose.out”, esta se configura con el primer parámetro *GmPSPrinter3ddose*, con el segundo parámetro se le asigna esta impresora al contador creado en el comando anterior con nombre *DoseScorer*.

El cuarto comando es `/gamos/scoring/addPrinter2Scorer GmPSPrinterSqdose DoseScorer`, tiene como función devolver histogramas de la distribución de dosis en dos dimensiones, para poder ver estos histogramas es necesario instalar Root en el computador. Para configurar esta impresora se utiliza el segundo parámetro `GmPSPrinterSqdose`, el cual se asigna al contador ya definido con nombre `DoseScorer`.

Para el caso de la energía depositada se utiliza el comando `/gamos/scoring/addScorer2MFD EnergyScorer GmG4PSEnergyDeposit DoseDet`, donde se le asigna al contador el nombre de `EnergyScorer`, al detector `DoseDet` creado anteriormente. Para indicar que se desea medir la energía depositada por partícula y no la dosis se utiliza el segundo parámetro que es `GmG4PSEnergyDeposit`. Y se utiliza el comando `/gamos/scoring/addPrinter2Scorer GmPSPrinterTextFile EnergyScorer` para indicar que se desea utilizar la impresora `GmPSPrinterTextFile` para imprimir la información en un archivo de texto con el mismo nombre de la impresora.

5.2.8. Visualización 3D

Gamos provee una gran herramienta para visualizar las geometrías y las partículas generadas en tres dimensiones. Para poder visualizar, es necesario descargar un programa que permita leer archivos con terminación “.wrl”. El programa utilizado por el autor para visualizar y crear las imágenes mostradas anteriormente es CORTONA3D, este es un software gratuito que permite abrir los archivos creados por GAMOS.

Los comandos utilizados para generar las geometrías y las trayectorias se detallan a continuación:

```

/control/execute ../MY_WORK/visVRML2FILE.in
#/control/execute ../MY_WORK/visOGLIX.in
/control/execute ../MY_WORK/visVRML2FILE.in
/vis/scene/endOfEventAction accumulate 3000

```

Para que estos comandos funcionen es necesario que en la carpeta donde están los archivos de la simulación, como la geometría y el archivo principal, estén los archivos “visASCII.in”, “visOGLIX.in” y “visVRML2FILE.in”. Estos archivos se pueden encontrar en el directorio de gamos “/GAMOS.5.0.0/examples”. En este caso la carpeta donde está la simulación antes detallada es “MY_WORK”, depende del nombre que el usuario coloque a esta carpeta la forma de los comandos.

El último comando, `/vis/scene/endOfEventAction accumulate 3000`, sirve para indicar la cantidad máxima de eventos que el usuario desea que sean mostrados en las gráficas 3D, entre mayor número de eventos mayor es la memoria requerida. El número por defecto de eventos que el programa permite es 1000.

5.2.9. Cantidad de partículas a correr

El último comando necesario para correr la simulación es `/run/beamOn 1000`, el único parámetro que recibe este comando es el número de partículas que el usuario desea que sean ejecutadas. Sin este comando, la simulación está incompleta y devolverá un error.

6 Resultados

6.1 Distribuciones de dosis en el fantoma

Con la impresora RTPSPDoseHistos utilizada en el contador que se destinó a medir la dosis, se generó un archivo llamado *“dose.RTPSPDoseHistos.root”*, los datos son analizados por medio de root, una framework para el procesamiento de datos desarrollado por el CERN para la investigación sobre física de altas energías. En este archivo se encuentran varios histogramas que detallan como se distribuye la dosis en el fantoma, en este trabajo se presentan los histogramas en una y dos dimensiones que ilustran la distribución de dosis por un millón de partículas irradiadas en el fantoma.

Se muestra en una dimensión la distribución de dosis a lo largo del eje X, Y, Z y se muestra en dos dimensiones la distribución de dosis en la cara frontal, lateral y superior. En los ejes es posible observar la posición a la que se encuentra el fantoma en cada uno de los ejes, los cuadros que dividen las imágenes de los histogramas en dos dimensiones son los voxels que se utilizaron para dividir el volumen de agua, por lo que cada una de las imágenes esta dividida en 10×10 voxels.

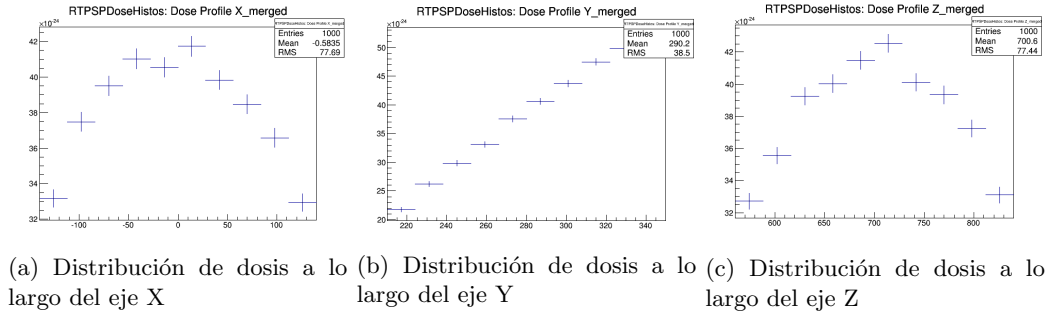


Figura 6.1: Distribución de dosis en el fantoma en 1D

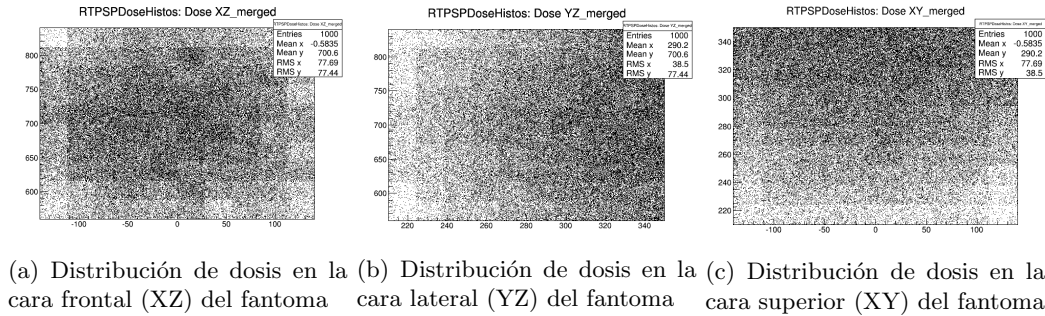


Figura 6.2: Distribución de dosis en el fantoma en 2D

Se puede observar en la Figura (5.1.a) que la distribución a lo largo del eje X se concentra en el centro del fantoma, lo que se corrobora con la imagen en dos dimensiones de la Figura (5.2.a). Para el eje Y es posible observar en la Figura (5.1.b) que la distribución de dosis es lineal, en la parte del fantoma que está frente a la fuente se tiene una dosis más alta, la cual disminuye a lo largo del eje Y del fantoma. Se comprueba con la Figura (5.2.b) que del lado derecho que es la cara frente al fantoma hay más concentración de dosis depositada que en el lado izquierdo. Por último, se puede observar la Figura (5.1.c) donde se observa la distribución de dosis en el eje Z, se puede observar que en la parte central del fantoma se encuentra la mayor concentración de dosis la cual disminuye hacia los extremos, esto se comprueba con la Figura (5.2.c).

6.2 Optimización del tiempo de cómputo

Se realizó la simulación sin utilizar las 3 capas de optimización: producción de cortes, filtros y límites de usuario. Luego se fue añadiendo capa por capa para poder obtener la curva de la Figura (5.3). En la cual se grafica cantidad de partículas irradiada contra tiempo. Se puede observar que a medida que se agregan las capas el tiempo de procesamiento disminuye, el mayor cambio se nota cuando se agregan los filtros que eliminan los electrones y partículas secundarias generadas por interacciones en el fantoma. Tomando en cuenta el tiempo de cómputo para un millón de partículas sin filtros y agregando todos los filtros se puede estimar una diferencia de 66% en el tiempo requerido para realizar la simulación.

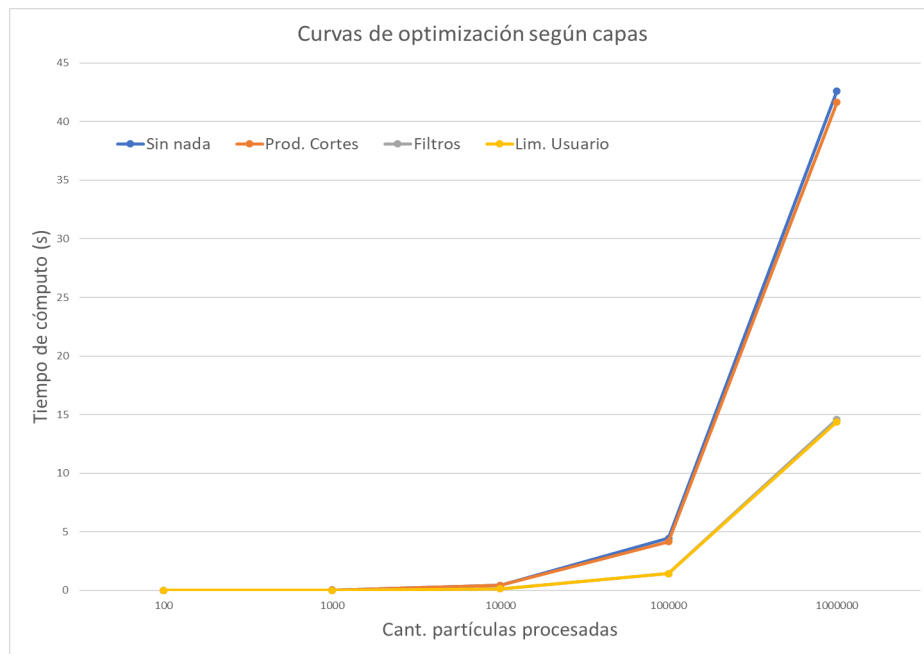


Figura 6.3: Curva de optimización de tiempo de cómputo

6.3 Dosis y energía depositada

Se realizó una muestra de 45 corridas utilizando diferentes semillas de números aleatorios en cada corrida para obtener estadística de la dosis y energía depositada en el fantoma por partícula. En el archivo *gamos.log* se encuentra la información extraída por los contadores *GmG4PSEnergyDeposit* y *GmG4PSDoseDeposit* luego de la simulación. Para el contador *GmG4EnergyDeposit* se obtuvo una lista con el índice de cada uno de los voxels del fantoma, seguido del valor esperado de la energía depositada por cada partícula en el voxel en *MeV*, con su respectivo error relativo. Al final de esta lista está la suma de todas las energías depositadas en *MeV*. Un ejemplo de esta lista se puede encontrar en la sección (8.2) de los anexos.

En el mismo archivo se encuentra la información extraída por el contador *GmG4PSDoseDeposit*, la dosis total obtenida con su respectivo error relativo. Un ejemplo de esta información se puede encontrar en la sección (8.3) de los anexos. Luego de realizar las 45 corridas, se extrajo la información dada por los contadores *GmG4PSEnergyDeposit* y *GmG4PSDoseDeposit*, se tabuló en un archivo *.xlsx* y se realizó estadísticas descriptivas para poder obtener la media de los valores medidos por GAMOS. En la siguiente tabla se encuentran los datos obtenidos por la estadística descriptiva de excel.

Cuadro 6.1: Resultados de energía y dosis depositada

Cantidad	Valor obtenido	Dimensional
Energía depositada	(25.99 ± 0.12)	[<i>eV</i>]
Dosis depositada	(0.3794 ± 0.0018)	[<i>fGy</i>]

6.4 Partículas irradiadas

La actividad típica de la fuente de Cs-137 según el Organismo Internacional de Energía Atómica para una fuente de calibración de Cs137 es $2200GBq$, para la fuente del LSCD se utiliza la actividad tomando en cuenta el decaimiento radioactivo para el 4 de julio de 2013 es de $773.3GBq$. Es necesario calcular la cantidad de partículas tomando en cuenta tres factores: el decaimiento de la fuente tomando en cuenta el tiempo que ha pasado desde que tomo la actividad, el factor de ángulo sólido del colimador de haz estrecho donde se encuentra la fuente y el tiempo que se irradiará. Con esta información se calcula la cantidad de partículas irradiada, como se muestra en la ecuación (5.1). (OIEA, 2009)

$$\#part. = 773.3GBq * 0.8923 * [6.201 * 10^{-3}] * 563.4s = 2.4106 * 10^{12}part. \quad (6.1)$$

Donde $773.3GBq$ es la actividad inicial de la fuente de Cs-137, $0.8923 = N/N_0$ es el factor de decaimiento de la fuente desde la fecha en que se tomo la actividad inicial, $6.20156 * 10^{-3}$ es el

factor de ángulo sólido tomando el ángulo sólido calculado en la ecuación (4.4) dividido $4\pi sr$ y $563.4s$ es el tiempo que se irradiará el fantoma para obtener una dosis de $1mSv$ a tres metros de la fuente.

6.5 Comparación con los datos de LSCD

Para comparar los datos obtenidos en la simulación con los datos del Laboratorio Secundario de Calibración Dosimétrica (LSCD), es necesario obtener el valor esperado de dosis recibida en el fantoma por $563.4s$ a tres metros de la fuente en dimensional de sievert (Sv).

Para obtener la dosis en Sv se multiplica el valor esperado de dosis por partícula, obtenido de la Tabla (5.1), por la cantidad de partículas simuladas, obtenida en la ecuación (5.1) y pasa de $[Gy]$ a $[Sv]$ utilizando el factor $1Gy = 1.204Sv$ según la Comisión Internacional de unidades de Radiación y medidas, por sus siglas en inglés (ICRU), en el artículo *Patient Dosimetry For X Rays Used In Medical Imaging*. La diferencia entre estos dos valores radica en que el sievert está corregido por el daño biológico que produce la dosis absorbida por la radiación, por su lado el Gy es la unidad utilizada para dosis absorbida por un material. Por esta razón se utiliza el sievert en el LSCD. (ICRU, 2005)

Con los datos del *Informe de dosimetría personal* el brindado por Laboratorios técnicos aplicaciones nucleares, del Ministerio de Energía y Minas de Guatemala, se obtuvo la dosis absorbida en las diez muestras brindadas. Utilizando el criterio de Chauvenet se descarta el dato 1.90, este se toma como un dato atípico y no se considera en la media de los datos enviados por el LSCD. En la siguiente tabla se compara la dosis obtenida en el LSCD y la dosis obtenida por el simulador de GAMOS.

Cuadro 6.2: Comparación entre la dosis absorbida por el fantoma según el LSCD y la dosis absorbida por el fantoma según el simulador de GAMOS

	Valor obtenido	Dimensional
LSCD	(0.842 ± 0.066)	$[mSv]$
Simulador	(1.101 ± 0.005)	$[mSv]$
Error	30.8%	

La incerteza del valor obtenido en la dosis depositada en el fantoma medido por el simulador puede acreditarse a varias fuentes de incertidumbre, la incertidumbre en la posición en X del fantoma, la incertidumbre en la posición en Y del fantoma, la incertidumbre en la posición en Z del fantoma, el ángulo de apertura del colimador de haz estrecho de la fuente de radiación, la altura a la que se encuentra la fuente, el espesor del contenedor de polimetilmetacrilato (acrílico) del fantoma. Esta incertidumbre puede ser estimada por la ecuación (5.2). Las variaciones en las

mediciones anteriormente mencionadas se les conoce como grados de libertad de la simulación.

$$\sigma_x = \sqrt{\sum_i \left(\frac{\partial D_i}{\partial x_i}\right)^2 \sigma_i^2} \quad (6.2)$$

Según la Tabla (5.2) el porcentaje de error de la dosis obtenida por la simulación es de 8.55%, se puede observar que la dosis obtenida es del mismo orden de magnitud de la dosis del LSCD. Este error puede ser aceptable para aplicaciones rápidas de dosimetría en un laboratorio donde se necesite recrear un escenario y no se cuente con mucho tiempo para realizar mediciones.

7 Conclusiones

Se documentó el código realizado para la simulación a detalle para que este pueda ser comprendido y replicado por futuros usuarios de GAMOS que deseen continuar realizando pruebas, además se realizó un repositorio con el código en línea para que puede ser descargado desde cualquier computador y se pueda reproducir con GAMOS.

Se obtuvo distribuciones de dosis en una dimensión para los ejes X, Y, Z, para que sea posible visualizar la distribución de dosis a lo largo de estos ejes. De igual forma se obtuvo la distribución de dosis en dos dimensiones para la cara frontal, lateral y superior del fantoma, de modo que sea posible la visualización en dos dimensiones de la distribución de dosis depositada en el fantoma.

Se optimizó el tiempo de computo de la simulación disminuyendo en un 66% el tiempo requerido para analizar un millón de partículas, utilizando producción de cortes, filtros y límites de usuario, para que sea posible reproducir de forma rápida el experimento sin importar las variables que se configuren en la simulación.

Se determinó que la dosis depositada en el fantoma a $3m$ de distancia, por un tiempo de $563.4s$ es de $1.101 \pm 0.005 mSv$, el cual presenta un 30.8% de error de acuerdo a la medición del LSCD según tabla (5.2). Por lo que se considera viable la implementación de sistemas computarizados que puedan simular dosis en fantomas a diferentes condiciones.

8 Bibliografía

- Attix, F. H. (1986). *Introduction to Radiological Physics and Radiation Dosimetry*. Wiley-VCH, Weinheim.
- CIEMAT (2013). Gamos User Guide. Retrieved from http://fismed.ciemat.es/GAMOS/gamos_userguide.php.
- Dewji, S. and Hertel, E. N. (2019). *Advanced Radiation Protection Dosimetry*. Chapman and Hall/CRC, Milton.
- Geant4 Collaboration (2019). Geant4 Scope of Application — IntroductionToGeant4 10.5 documentation. <http://geant4-userdoc.web.cern.ch/geant4-userdoc/UsersGuides/IntroductionToGeant4/html/IntroductionToG4.html>.
- José, E. and Ché, F. (2018). *SIMULACIÓN DE LA RADIACIÓN DISPERSA DE RAYOS X DE FLUOROSCOPIA POR POLIMETILMETACRILATO (PMMA) UTILIZANDO GEANT4*. Trabajo de graduación, Universidad San Carlos de Guatemala.
- OIEA (2009). *Clasificación de las fuentes radiactivas: guía de seguridad*, volume No. RS-G-1. Organismo Internacional de Energía Atómica, Viena.
- Oxford University Press (2011). *ICRU report 85: fundamental quantities and units for ionizing radiation*, volume 150. Oxford.

9 Anexos

9.1 Lista de materiales para producción de cortes

===== Table of registered couples =====

Index : 0 used in the geometry : Yes

Material : G4_AIR

Range cuts : gamma 100 um e- 100 um e+ 100 um proton 100 um

Energy thresholds : gamma 990 eV e- 990 eV e+ 990 eV proton -1 MeV

Region(s) which use this couple :

DefaultRegionForTheWorld

Index : 1 used in the geometry : Yes

Material : NIST_Concrete, Barite (TYPE BA)

Range cuts : gamma 100 um e- 100 um e+ 100 um proton 100 um

Energy thresholds : gamma 5.31198 keV e- 138.268 keV e+ 135.508 keV proton -1 MeV

Region(s) which use this couple :

DefaultRegionForTheWorld

Index : 2 used in the geometry : Yes

Material : NIST_Polymethyl Methacrylate

Range cuts : gamma 100 um e- 100 um e+ 100 um proton 100 um

Energy thresholds : gamma 1.1173 keV e- 93.0104 keV e+ 91.7686 keV proton -1 MeV

Region(s) which use this couple :

DefaultRegionForTheWorld

Index : 3 used in the geometry : Yes

Material : G4_WATER

Range cuts : gamma 100 um e- 100 um e+ 100 um proton 100 um

Energy thresholds : gamma 1.10981 keV e- 84.6589 keV e+ 83.5287 keV proton -1 MeV

Region(s) which use this couple :

DefaultRegionForTheWorld

9.2 Lista de energía depositada por voxel en el fantoma dados por la herramienta *GmG4PSEnergyDeposit*

SetUnit GmPSPrinterCout 1 Name MeV

MultiFunctionalDet: DoseDet

PrimitiveScorer: EnergyScorer

Number of entries= 1000

index: 0 = 1.79998e-08 +-(REL) 0.173165 MeV sumV2= 9.71559e-18

index: 1 = 1.54583e-08 +-(REL) 0.186367 MeV sumV2= 8.29994e-18

index: 2 = 1.27626e-08 +-(REL) 0.19542 MeV sumV2= 6.22051e-18

index: 3 = 1.19187e-08 +-(REL) 0.199443 MeV sumV2= 5.65071e-18

....

index: 997 = 3.20029e-08 +-(REL) 0.128633 MeV sumV2= 1.69477e-17

index: 998 = 2.97704e-08 +-(REL) 0.129178 MeV sumV2= 1.47901e-17

index: 999 = 2.86136e-08 +-(REL) 0.131166 MeV sumV2= 1.40869e-17

EnergyScorer SUM_ALL: 2.61209e-05 +-(REL) 0.00446387 MeV

9.3 Datos de dosis dados por la herramienta *GmG4PSDoseDeposit*

SetUnit GmPSPrinter3ddose 1e-12 Name Gy

HEADER NEvent 1e+06 Nvoxels 10 10 10

TYPE 0

SetUnit RTPSPDoseHistos 1e-12 Name Gy

CreateHisto1D RTPSPDoseHistos: Dose = 36201

CreateHisto1D RTPSPDoseHistos: Dose-volume = 36202

CreateHisto1D RTPSPDoseHistos: Dose Error REL = 36203

CreateHisto1D RTPSPDoseHistos: log10(Dose Error REL) = 36204

CreateHisto1D RTPSPDoseHistos: Dose Error = 36205

CreateHisto1D RTPSPDoseHistos: log10(Dose Error) = 36206

CreateHisto2D RTPSPDoseHistos: log10(Dose) .vs. log10(Dose Error) = 36207

CreateHisto2D RTPSPDoseHistos: $\log_{10}(\text{Dose})$.vs. $\log_{10}(\text{Dose Error REL}) = 36208$
 RTPSPDoseHistos MINIMUM DOSE 1.24155e-19 Gy
 RTPSPDoseHistos MAXIMUM DOSE 7.69349e-19 Gy
 RTPSPDoseHistos AVERAGE ERROR 20% = 0.145581
 RTPSPDoseHistos AVERAGE ERROR 50% = 0.126631
 RTPSPDoseHistos AVERAGE ERROR 90% = 0.105302
 RTPSPDoseHistos TOTAL DOSE = 3.8129e-16 +-(REL) 0.00446387
 CreateHisto1D RTPSPDoseHistos: Dose Profile X_merged = 66301
 CreateHisto1D RTPSPDoseHistos: Dose Profile Y_merged = 66302
 CreateHisto1D RTPSPDoseHistos: Dose Profile Z_merged = 66303
 CreateHisto2D RTPSPDoseHistos: Dose XY_merged = 66304
 CreateHisto2D RTPSPDoseHistos: Dose XZ_merged = 66305
 CreateHisto2D RTPSPDoseHistos: Dose YZ_merged = 66306

9.4 Árbol de volúmenes

```

/gamos/geometry/printVolumeTree 2
DumpG4PVLVTree 2
DumpPV 2
0$$$ LOGICAL VOLUME = world Solid: world MATERIAL: G4_AIR CUBIC_VOLUME: 2.31314e-
11 MASS: 3.12824e+08
DumpPV 2
1 $$$ LOGICAL VOLUME = ParedConcreto Solid: ParedConcreto MATERIAL: NIST_Concrete,
Barite (TYPE BA) CUBIC_VOLUME: 1.49575e-11 MASS: 3.12748e+08
DumpPV 2
1 $$$ LOGICAL VOLUME = CajaFantoma Solid: CajaFantoma MATERIAL: NIST_Polymethyl
Methacrylate CUBIC_VOLUME:
4.01849e-16 MASS: 2984.69
DumpPV 2
1 $$$ LOGICAL VOLUME = contenedor.fantoma Solid: contenedor.fantoma MATERIAL: G4_
WATER CUBIC_VOLUME:
1.75855e-15 MASS: 10976
DumpPV 2
2 $$$ LOGICAL VOLUME = fantoma Solid: fantoma MATERIAL: G4_WATER CUBIC_VOLUME:
1.75855e-18 MASS: 10.976
  
```

9.5 Archivo de entrada utilizado *Prueba.in*

Nota: Las líneas con tres numerales (###), son comentarios para documentar el código. El detalle del código se encuentra en el capítulo 3. Además, el archivo de entrada, el de geometría y los necesarios para crear las visualizaciones 3D se encuentran en el siguiente repositorio de GitHub: https://github.com/men16084/GAMOS_SIMULACION_TESIS.git.

```
### Inicio del código ### Configuración de verbosidad en silent, ver sección (2.6) /tracking/verbose -1
```

```
### Geometría desde archivo de texto, ver sección (2.2.1) /gamos/geometry GmGeometryFromText
/gamos/setParam GmGeometryFromText:FileName ContruyendoGeometria.geom
/gamos/setParam GmReadPhantomGeometry:Phantom:FileName ContruyendoGeometria.geom
```

```
### Activación de física electromagnética y generador, ver sección (2.3) y (2.5) /gamos/physicsList GmEMPhysics
/gamos/generator GmGenerator
```

```
### Generación de cortes (region), ver sección (2.5.1.1)
#/gamos/geometry/createRegion Pared ParedConcreto
#/gamos/geometry/createRegion FantomaAcrilico CajaFantoma
#/gamos/geometry/createRegion FantomaAgua contenedor_fantoma
#/gamos/geometry/createRegion Aire world
#/gamos/physics/setCuts Pared 100.*um 100.*um 100.*um 100.*um
#/gamos/physics/setCuts FantomaAcrilico 100.*um 100.*um 100.*um 100.*um
#/gamos/physics/setCuts FantomaAgua 100.*um 100.*um 100.*um 100.*um
#/gamos/physics/setCuts Aire 100.*um 100.*um 100.*um 100.*um
```

```
### Generacion de cortes (energia), ver sección (2.5.1.2)
/gamos/userAction GmCutsEnergy2RangeUA
#/gamos/physics/ECuts2RangeCuts G4_AIR 990.*eV gamma
#/gamos/physics/ECuts2RangeCuts "NIST_Concrete, Barite (TYPE BA)" 5.31198*keV gamma
#/gamos/physics/ECuts2RangeCuts "NIST_Polymethyl Methacrylate" 1.1173*keV gamma
#/gamos/physics/ECuts2RangeCuts G4_WATER 1.10981*keV gamma
```

```
### Inicio de la información para la corrida /run/initialize
```

```

### Fuente de radiación, ver sección (2.3.1.2)
/gamos/generator/addIsotopeSource fuenteIO Cs137 660.*keV
/gamos/generator/positionDist fuenteIO GmGenerDistPositionPoint 0 3280 700
/gamos/generator/directionDist fuenteIO GmGenerDistDirectionCone 0. -1. 0. 0.155862

### Impresión del árbol de volúmenes. Esta línea de código es opcional, ver sección (2.2.3)
/gamos/geometry/printVolumeTree 2

### Filtros y límites de usuario para optimización de cómputo, ver sección (2.4.1) y (2.7.2)
/gamos/filter AcceptStepP GmElectronFilter
/gamos/filter AcceptStepS GmSecondaryFilter

/gamos/userAction GmKillAtStackingActionUA AcceptStepP
/gamos/userAction GmKillAtStackingActionUA AcceptStepS

### Límites de usuario, ver sección (2.7.1) /gamos/physics/userLimits/setMaxTrkLen Lim-
iteR world gamma 3.e+03

### Dosis y energía depositada en el fantoma, ver sección (2.4.2)
/gamos/scoring/createMFDetector DoseDet fantoma

/gamos/scoring/addScorer2MFD EnergyScorer GmG4PSEnergyDeposit DoseDet
#/gamos/scoring/addPrinter2Scorer GmPSPrinterTextFile EnergyScorer

/gamos/scoring/addScorer2MFD DoseScorer GmG4PSDoseDeposit DoseDet
/gamos/scoring/addPrinter2Scorer GmPSPrinter3ddose DoseScorer
/gamos/scoring/addPrinter2Scorer GmPSPrinterSqdose DoseScorer
/gamos/scoring/addPrinter2Scorer RTPSPDoseHistos DoseScorer

### Visualización 3D (Esta parte es opcional, puede excluirse del código sin alterar resulta-
dos)
/control/execute ../MY_WORK/visVRML2FILE.in
#/control/execute ../MY_WORK/visOGLIX.in
/control/execute ../MY_WORK/visVRML2FILE.in
/vis/scene/endOfEventAction accumulate 3000

```

Número de partículas simuladas /run/beamOn 1000000

9.6 Archivo de entrada con la geometría *ContruyendoGeometria.geom*

Nota: En este archivo los comentarios no van con numerales sino con doble barra diagonal (//), ver las sección (2.2) y el capítulo 3.

// Definición Materiales

// Acrílico

:MIXT "NIST_Polymethyl Methacrylate" 1.19 3

H 0.080541

C 0.599846

O 0.319613

:MATE_MEE "NIST_Polymethyl Methacrylate" 74*eV

// Concreto

:MIXT "NIST_Concrete, Barite (TYPE BA)" 3.35 9

H 0.003585

O 0.311622

Mg 0.001195

Al 0.004183

Si 0.010457

S 0.107858

Ca 0.050194

Fe 0.047505

Ba 0.4634

:MATE_MEE "NIST_Concrete, Barite (TYPE BA)" 248.2*eV

// Matriz de Rotación

:ROTM RM0 0. 0. 0.

// Volumen de aire

:VOLUME world BOX 2750 3750 1750 G4_AIR

```

// Pared de Concreto
:SOLID ConcretoExt BOX 3130. 4130. 2300. //Pared exterior
:SOLID ConcretoInt BOX 2750. 3750. 1750. //Pared interior
:VOLU ParedConcreto SUBTRACTION ConcretoExt ConcretoInt RM0 0. 0. 0. "NIST_Concrete,
Barite (TYPE BA)"
:COLOR ParedConcreto 1 1 1 1
:PLACE ParedConcreto 1 world RM0 0. 0. 0.

```

```

//Fantoma
:SOLID CajaAcrilicaEXT BOX 150. 75. 150. //Exterior caja Acrilico
:SOLID CajaAcrilicaINT BOX 140. 70. 140. //Interior caja Acrilico
:VOLU CajaFantoma SUBTRACTION CajaAcrilicaEXT CajaAcrilicaINT RM0 0. 0. 0. "NIST_
Polymethyl Methacrylate"
:PLACE CajaFantoma 1 world RM0 0. 280. 700.

```

```

//Volumen de agua dentro del fantoma
:VOLU contenedor_fantoma BOX 140. 70. 140. G4_WATER
:PLACE contenedor_fantoma 1 world RM0 0. 280. 700.
:COLOR contenedor_fantoma 0 0 1 1
:VOLU fantoma BOX 14.0 7.0 14.0 G4_WATER
:PLACE_PARAM fantoma 1 contenedor_fantoma PHANTOM 10 10 10 28. 14. 28.

```

9.7 Cortona3D

El programa que se utilizó para la visualización en 3D se llama Cortona3D, es un programa gratuito que se puede descargar en el siguiente link: <http://www.cortona3d.com/en/cortona3d-viewer-download>.

9.8 Informe dosimetría personal extendido por LABORATORIOS TÉCNICOS APLICACIONES NUCLEARES