



# BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

---

---

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

**SIMULACIÓN MONTE CARLO DE TRANSPORTE DE  
LA RADIACIÓN ELECTROMAGNÉTICA EN  
TEJIDOS BIOLÓGICOS CON GEOMETRÍAS DE  
SENSADO MULTICANAL**

TESIS PRESENTADA

PARA OBTENER EL TÍTULO DE

**INGENIERO EN CIENCIAS DE  
LA COMPUTACIÓN**

PRESENTA:

**BIBIANA CUERVO SOTO**

ASESORES:

M.C. ALFONSO GARCÉS BÁEZ

DR. FELIPE ORIHUELA ESPINA





---

## Agradecimientos

### **A mis padres**

Que siempre me han mostrado su apoyo incondicional. A mi padre por siempre apoyarme en todos mis proyectos. A mi mami por ser mi ejemplo, por siempre luchar en la vida dando la mejor de las sonrisas y ayudando al prójimo, esas enseñanzas de vida las llevo en el corazón y me hacen sentir mucho más orgullosa de ti. Los amo con todo mi corazón.

### **A mi hermano**

Gracias Henry por tu apoyo, amistad, complicidad, pero sobre todo por el amor que nos une, y que aún en la distancia nos hace sentir tan cerca. Que este logro te sirva de motivación para terminar tu carrera.

### **A mi amoroso esposo y adorable hijo**

Mis tesoros más preciados en este mundo, ambos me motivan a ser mejor persona cada día y a superarme en la vida. Amor, gracias por tu apoyo incondicional, por motivarme a titularme, por tu conocimiento y apoyo en la realización de este tema de investigación, pero sobre todo Gracias por estar siempre a mi lado. Mi Angelito hermoso, eres mi motivo, mi orgullo y mi razón de existir, te amo con todo mi ser, va por ti este logro que estoy teniendo. Los amo corazoncitos.

### **Al Dr. Felipe Orihuela Espina**

Por su inagotable paciencia, orientación, motivación, sus conocimientos que han sido fundamentales para la realización de este trabajo. Por aceptarme como tesista, pero Gracias sobre todo por la amistad y respaldo proporcionado.

### **Al MC. Alfonso Garcés Báez**

Por todo el apoyo brindado a lo largo de la carrera, por la confianza depositada en mi, por la paciencia, por haber aceptado ser mi asesor, gracias por sus conocimientos y su tiempo. Mi total admiración y respeto.

### **Al Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE)**

Por la beca proporcionada para la realización de éste tema de investigación. Gracias por ese apoyo que me ayudó a llevar a buen término mi tema de tesis.

### **A la Facultad de Ciencias de la Computación de la BUAP**

Por el apoyo económico para poder asistir a la XVI Reunión de neuroimagen en el CIMAT, Guanajuato, Gto., y presentar ahí un cartel referente a mi tema de investigación.

Gracias a todos y cada uno de mis familiares y amigos que de alguna manera estuvieron cerca de mi alentándome a continuar. Sin todos ustedes esto no habría podido ser posible.

Gracias **Dios** por permitirme vivir esta experiencia y por todas las bendiciones que tengo en la vida.

Con cariño, amor y respeto,

Bibiana Cuervo Soto.



## Contenido

Agradecimientos .....	3
Contenido .....	5
Tabla de Figuras .....	7
Lista de Acrónimos .....	9
Resumen.....	11
Organización de la tesis .....	13
1. Introducción .....	15
1.1 Antecedentes .....	15
1.2 Transporte de la radiación.....	16
1.3 Motivación y Justificación .....	17
1.4 Problemática.....	18
1.5 Preguntas de Investigación .....	18
1.6 Hipótesis .....	19
1.7 Objetivos .....	19
1.7.1 Objetivo general:.....	19
1.7.2 Objetivos específicos:.....	19
1.8 Alcances y limitaciones .....	19
1.9 Contribuciones .....	20
1.10 Presentaciones.....	20
1.11 Publicaciones.....	21
2. Fundamentos teóricos.....	23
2.1 Teoría de la luz y el color .....	25
2.2 Imagen Óptica Difusa (DOI) .....	28
2.3 Caracterización óptica de tejidos biológicos .....	29
2.4 Método de Monte Carlo .....	31
3. Software de Transporte de la radiación .....	34
3.1 <i>mcml</i> .....	34
4. Metodología .....	39
4.1 Diseño e implementación de MOCARTS.....	39
4.2 Diseño del sistema .....	40
4.3 Diagrama de Clases.....	46

4.4	Diagrama de Casos de Uso .....	48
4.5	Diagrama de flujo .....	49
4.6	Abstracción del Tejido .....	50
4.7	Definición en XML de una simulación .....	51
4.8	Rejillas.....	55
4.9	Geometrías de sensado multicanal.....	55
5.	Experimentos y resultados .....	58
5.1	fNIRS.....	58
5.2	Verificación de MOCARTS .....	59
5.3	Validación .....	65
5.3.1	Medición del error cuadrático medio entre <i>mcml</i> y MOCARTS.....	66
6.	Simulador MOCARTS .....	70
7.	Conclusiones y trabajos futuros .....	74
7.1	Conclusiones .....	74
7.2	Trabajos Futuros.....	75
	Anexos .....	78
	Anexo 1 Archivo de definición de modelo de simulación.....	78
	Anexo 2 Fichero de salida del modelo simulado.....	81
	Anexo 3 Código fuente de MOCARTS.....	83
	Glosario.....	101
	Bibliografía .....	105

## Tabla de Figuras

Figura 1 Múltiples fuentes de iluminación .....	18
Figura 2 Incidencia de una piedra sobre un lago .....	24
Figura 3 Propiedades de la onda Seno. ....	24
Figura 4 Espectro Electromagnético .....	26
Figura 5 Ley de Snell. ....	27
Figura 6 Curvas de absorción de algunos compuestos biológicos en la ventana óptica de IR .....	28
Figura 7 Comportamiento de un fotón en el tejido. ....	30
Figura 8 Distribución de probabilidad exponencial para el tamaño del paso, donde $p(x)$ es la probabilidad del tamaño del paso, $a$ y $b$ son la distancia máxima y mínima (respectivamente) que el tamaño del paso puede tomar y $x_1$ es un ejemplo de un valor muestreado .....	31
Figura 9 Distribución angular de probabilidad Henyey-Greenstein para obtener la dirección del fotón. En el eje de ordenadas se representa la probabilidad de tomar el valor del coseno de $\theta$ indicado en la abscisa.....	32
Figura 10 Diagrama de Flujo del simulador <i>mcmI</i> .....	36
Figura 11 Modularización del sistema MOCARTS. ....	41
Figura 12 Diagrama de clases del paquete TissueModel.....	46
Figura 13 Diagrama de clases del paquete Simulation. ....	47
Figura 14 Diagrama de clases del paquete SensingGeometry. ....	47
Figura 15 Diagrama de Casos de Uso de MOCARTS.....	48
Figura 16 Diagrama de Flujo de MOCARTS.....	49
Figura 17 Imagen representativa de una célula .....	50
Figura 18 Tejido nervioso .....	50
Figura 19 Ejemplo de rebanada biológica.....	51
Figura 20 Definición del DTD (Document Type Definition) de XML para la definición de un modelo de simulación. ....	53
Figura 21 Definición del DTD (Document Type Definition) de XML para el fichero de salida del modelo de simulación. ....	54
Figura 22 Rejillas de simulación a) <i>mcmI</i> , b) MOCARTS.....	55
Figura 23 Matriz de listas ligadas para almacenar el valor de la energía ( $\nu$ ) proveniente de determinada fuente de luz ( $f_i$ ). ....	57
Figura 24 Principios de la interacción de la luz con la material. ....	58
Figura 25 Las 4 capas del tejido simulado. Aunque de forma didáctica, la imagen idealiza el modelo. La geometría del tejido simulado es plana.....	60
Figura 26 Trayectoria de un fotón con <i>mcmI</i> (azul) y la nueva herramienta de software (rojo). La secuencia acaba con la absorción del paquete fotónico. ....	61
Figura 27 Matriz de absorción en ambos simuladores.....	62
Figura 28 Matriz de reflectancia difusa en ambos sistemas.....	63
Figura 29 Matriz de transmitancia en ambos sistemas.....	64
Figura 30 Comparación de Absorción total entre <i>mcmI</i> y MOCARTS.....	66
Figura 31 Comparación de Reflectancia total entre <i>mcmI</i> y MOCARTS.....	67
Figura 32 Comparación de Transmitancia total entre <i>mcmI</i> y MOCARTS.....	68

Figura 33 Error cuadrático medio entre las simulaciones de <i>mcmI</i> y MOCARTS.....	69
Figura 34 Representación 3D del modelo de tejido simulado. ....	70
Figura 35 Superficie XY del tejido simulado con cortes en los ejes ZX y ZY, los cuales se intersectan en el origen. ....	71
Figura 36 Corte del volumen de tejido simulado donde se muestran 5 fuentes. ....	72
Figura 37 Interfaz Gráfica de Usuario de MOCARTS. ....	75

## Lista de Acrónimos

DOI	Diffuse Optical Imaging (Imagen Óptica Difusa)
IR	Infrarrojo
MC	Monte Carlo
RTE	Radiation Transport Equation (Ecuación de Transferencia de Radiación)
$\mu_a$	Coefficiente de Absorción
$\mu_s$	Coefficiente de Esparcimiento
DOT	Diffuse Optical Tomography (Tomografía de Óptica Difusa)
XML DTD	eXtensible Markup Language ('Lenguaje de Marcas Extensible) Document Type Definition (Definición de tipo de documento)
fNIRS	Functional Near Infrared Spectroscopy (Espectroscopia funcional en el Infra-rojo Cercano )
POO	Programación Orientada a Objetos
HHb	Hemoglobina reducida
HbO <sub>2</sub>	Hemoglobina oxigenada
GPU	Graphics Processing Units (Unidad de Procesamiento Gráfico )
RMSE	Root Mean Squared Error (Raíz del Error Cuadrático Medio)



---

## Resumen

La *imagen óptica difusa* es una modalidad de imagen médica no invasiva capaz de explorar el funcionamiento de órganos como por ejemplo el cerebro. Su uso requiere del entendimiento del fenómeno de transporte de la radiación electromagnética en el tejido biológico. Uno de los simuladores más populares para este fin (mcml) está limitado a simulaciones de un solo canal (par emisor-detector).

El objetivo de esta tesis es aumentar con capacidades de sensado multicanal el software de simulación de transporte de la radiación electromagnética existente.

La hipótesis de esta investigación establece que las geometrías de sensado de un solo canal son un caso específico de geometrías más complejas de sensado multicanal y la flexibilización de la arquitectura computacional no debe alterar el resultado de la simulación original. Para esto, se ha diseñado e implementado una arquitectura de software que ha tenido como resultado un simulador de transporte de la radiación con múltiples fuentes de sensado (MOCARTS).

Los métodos desarrollados e implementados en este simulador han sido verificados y validados experimentalmente. Se propusieron e implementaron estructuras de datos para el manejo de la información de cada uno de los canales de sensado. Por otro lado, se flexibilizó la especificación de parámetros de entrada de la simulación, así como de los datos de salida. Una vez implementado el simulador, se realizó la verificación mediante la medición del error entre los datos del simulador mcml y MOCARTS el cual mostró un comportamiento coherente y aceptable. La validación mostró una convergencia de los resultados entre ambos simuladores, comprobando así la hipótesis planteada.

Como resultado de este trabajo, el simulador MOCARTS ha extendido las capacidades de mcml en tres puntos principales: la definición de tejidos biológicos, la capacidad de simulación de múltiples canales y la especificación de formatos de archivos de entrada/salida más flexibles.

Este simulador proporciona una nueva herramienta para la comunidad de imagen óptica difusa con capacidades que actualmente no están disponibles en el software usado por la comunidad.



## Organización de la tesis

El documento presente se organiza de la siguiente forma. En el capítulo 1 Introducción, se plantean las bases, motivación, los alcances y limitaciones que dieron origen a este tema de investigación. Iniciando con una breve introducción al tema de los métodos ópticos los cuales analizan la interacción de la luz con la materia, la cual requiere de la definición de un modelo para estimar la irradiación observable a partir de las especificaciones de las propiedades ópticas del tejido a estudiar. Para el desarrollo de este tema de investigación se hace uso de las simulaciones de Monte Carlo.

En el capítulo 2 Fundamentos teóricos, se hace mención de los conceptos fundamentales para comprender la teoría que se desarrolla en esta tesis; el campo electromagnético, el infrarrojo cercano, los fenómenos observables en la interacción de la luz con la materia, así como las características ópticas de los tejidos biológicos. Se describe el método de Monte Carlo y se enlistan algunos sistemas para el transporte de la radiación.

El capítulo 3 Software de Transporte de la radiación, enlista software que permite conocer cómo se comporta la luz al interactuar con la materia, así como la descripción del funcionamiento del software de simulación que en este tema de investigación se tiene como gold standard, *mcml*.

En el capítulo 4 Metodología, se desarrolla la metodología para el diseño e implementación de MOCARTS (**MO**nte **CA**rlo **R**adiation **T**ransport **S**imulation). El uso de la ingeniería de Software se utilizó para el diseño de clases del sistema creando una arquitectura orientada a objetos. Se parte de las definiciones de tejido y órgano para crear una abstracción de éstas y poder modelarlas computacionalmente. Así como también se modela el árbol de definición en XML del proceso de simulación de los datos de entrada y los datos de salida. Se muestra el desarrollo de la estructura para poder irradiar un tejido con múltiples fuentes y la relación de éstas con sus detectores, así como seguir cada fotón irradiado en los detectores y saber a qué fuente de irradiación pertenece.

El capítulo 5 Experimentos y resultados, muestra los experimentos y resultados de haber desarrollado esta herramienta de simulación haciendo pruebas con un modelo de la cabeza adulta en la cual se siguió la trayectoria de un fotón comparándolo con el gold standard (*mcml*), además de obtener dicha trayectoria se simuló la irradiación de 100,000 fotones con lo cual se pudo obtener la matriz de absorción, transmitancia y reflectancia, nuevamente haciendo la comparación con *mcml*.

En el capítulo 6 Simulador MOCARTS, se muestran algunos aspectos de las nuevas capacidades de MOCARTS no presentes en *mcml*, como la inclusión de múltiples fuentes de iluminación.

En el capítulo 7 Conclusiones y trabajos futuros, se resumen las conclusiones derivadas de esta investigación y se incluyen los trabajos futuros de este tema de tesis.

Finalmente, se muestra la información complementaria al tema de investigación (Anexos), el glosario, el material bibliográfico utilizado para la documentación de este trabajo, así como también parte del código desarrollado para el simulador.



# 1. Introducción

## 1.1 Antecedentes

El estudio de la materia nos permite entender la física que gobierna los cuerpos, conocimiento que luego podemos aprovechar para desarrollar soluciones para la medicina o la industria entre otros. Se denominan métodos ópticos de análisis a un conjunto de técnicas y herramientas que aprovechan la interacción de la radiación electromagnética con la materia para el estudio de esta última. Los métodos ópticos de análisis se pueden clasificar en 2 tipos fundamentales: espectroscópicos y no espectroscópicos. Los primeros utilizan la medición de la intensidad y la longitud de onda de la energía radiante, mientras que los métodos no espectroscópicos miden cambios en la dirección o en las propiedades físicas de la radiación electromagnética [1].

De particular interés biomédico, entre los métodos espectroscópicos está la imagen óptica difusa (DOI por sus siglas en inglés) que es una técnica mediante la cual se pueden obtener imágenes del tejido biológico, es decir, imágenes del metabolismo de los tejidos. Ésta técnica mide la atenuación óptica de los cromóforos presentes en el tejido, y se basa en los espectros de extinción de las distintas moléculas que componen dichos tejidos.

La utilización de la imagen óptica difusa requiere resolver un problema inverso donde a partir de la intensidad de la radiación saliente de un tejido, se pueda establecer de forma unívoca una distribución espacial de los cromóforos, y a partir de esta distribución se lleva a cabo una inferencia de la actividad metabólica que presenta el tejido. En forma práctica, esto implica la definición de un modelo del tejido que será confirmado o refutado por la observación experimental de la radiación. Los tejidos biológicos tienen 3 características que los definen y que deben ser modeladas para el uso apropiado de la imagen óptica difusa:

- **Mecánicos:** resultante de la masa física y que incluye entre otros aspectos la resistencia del tejido o la deformación, tales como la rigidez. Afectan a los métodos ópticos alterando la propagación de las ondas electromagnéticas.
- **Geométricos:** referidos a la distribución espacial de las moléculas y determina entre otras los puentes entre capas y superficies así como la distribución espacial de las partículas atenuantes. Estas afectan a la interacción de la luz con la materia alterando la reflexión y refracción.
- **Ópticos:** responsables de la sombra proyectada por las moléculas y responsable último de los mecanismos básicos de interacción con la radiación electromagnética, tales como la absorción y esparcimiento.

Este trabajo se centra en la necesidad de resolver el modelado de tejido biológico para la decodificación de sus propiedades a partir de la radiación sensada (proceso inverso). A menudo, esto pasa por una simulación de la interacción de la luz con la materia a partir de un modelo

específico asumido (proceso directo). En esta tesis nos abstraeremos de las características mecánicas, propondremos una solución básica a las geométricas y nos centraremos en las ópticas.

## 1.2 Transporte de la radiación

La reconstrucción de la imagen es un problema inverso donde se intenta recuperar información histo-fisiológica a partir de mediciones de intensidad lumínica. Este proceso es consabido que es un problema mal planteado en el sentido de Hadamard [2], por lo que se requiere imponer ciertas condiciones y restricciones para poder ofrecer una solución única y aproximada. A menudo el problema se ataca mediante la producción previa de un modelo directo que permite estimar la irradiación observable a partir de la especificación explícita o implícita de las propiedades ópticas del tejido bajo estudio. Estas propiedades ópticas incluyen los coeficientes de absorción y esparcimiento o dispersión, los índices de refracción, el grosor de las capas del tejido y una distribución de la isotropía o anisotropía del esparcimiento [3]. Este modelo directo es una solución de la transferencia de la radiación en la materia originalmente expresado por las ecuaciones de Maxwell. Existen varias soluciones que difieren en el grado de fidelidad y suposiciones que conllevan; algunos ejemplos incluyen, la ecuación de Boltzman [4], la teoría de la difusión [5], los modelos de Kubelka-Munk [6], o los que se tratarán en este trabajo; las simulaciones de Monte Carlo [7].

El método de Monte Carlo [8] permite aproximar de manera sencilla procesos estocásticos complejos. Para ello, se simulan de forma masiva eventos singulares con un resultado aleatorio, y la solución final se obtiene mediante la integración de la distribución resultante de las muestras correspondientes a cada uno de los eventos singulares originalmente simulados. El grado de exactitud de la solución está directamente relacionado con el número de eventos singulares simulados, en otras palabras con el número de muestras. En el caso particular de la simulación de transporte de la radiación, estos eventos singulares son la irradiación de fotones individuales con una energía finita que se va perdiendo (o depositando en el tejido) a medida que ocurren fenómenos de absorción y dispersión.

Actualmente ya existe software que simula la transferencia de la radiación en tejidos biológicos, algunos de ellos como *mcm1* basados en el método de Monte Carlo. Cada uno de estos tiene diferentes ventajas y limitaciones que se describirán más adelante. En esta tesis centraremos nuestra atención en particular a proponer una solución a una de las limitaciones de la herramienta *mcm1*, la incapacidad de tolerar geometrías de irradiación y detección multicanal complejas de forma concomitante.

Por supuesto existe software de simulación de la transferencia de la radiación en tejidos biológicos capaz de llevar a cabo simulaciones para geometrías complejas; como por ejemplo el software NIRS-FAST, pero su motor de simulación está basado en modelado de elemento finito con otras implicaciones que exceden el ámbito de este trabajo.

## 1.3 Motivación y Justificación

### **Motivación:**

La comprensión que aún tenemos del funcionamiento in-vivo de muchos tejidos biológicos es limitada. Por ejemplo, del cerebro desconocemos aspectos incluso básicos como por ejemplo la caracterización específica de la respuesta hemodinámica ante una variedad de estímulos [9] y que podrían ser críticos para entender problemas neuropsiquiátricos y neuropsicológicos. La motivación de este tema de investigación inicia con el hecho de poder comprender la interacción de la luz con los tejidos biológicos complejos para la adquisición de información que aún en estos tiempos es incompleta.

El estudio ex-vivo [10] de la materia biológica nos ha permitido conocer muchos detalles de la estructura histológica de los tejidos e inferir aspectos importantes de su función necesarios para dar soluciones a problemas biológicos y médicos. No obstante, el funcionamiento in-vivo de los tejidos presenta inevitablemente diferencias críticas con su análogo in-vitro [11] que demandan la necesidad de desarrollar imaginologías biomédicas in-vivo y preferiblemente no invasivas, como por ejemplo la imagen óptica difusa [12].

En la actualidad ya existen simuladores de transferencia de radiación los cuales aportan soluciones parciales. Estos simuladores atienden la resolución del proceso directo permitiendo estimar la distribución posterior de la radiación bajo unas determinadas suposiciones sobre la física del problema. En el caso de la imagen óptica difusa las geometrías de sensado –la distribución espacial de las fuentes de iluminación y los fotosensores- suelen ser complejas y su resolución en los simuladores basados en Monte Carlo es en el mejor de los casos limitada.

### **Justificación:**

La imagen óptica difusa es una modalidad de bioimagen funcional con una alta versatilidad aplicable a diferentes tejidos; piel [13], colon [14], cerebro [15], entre otras de forma no invasiva, con una resolución espacial aceptable en comparación con otras modalidades alternativas. Aunado a su bajo coste, tanto de inversión inicial, como de adquisición, la imagen óptica difusa es una herramienta con claras aplicaciones en investigación, como en uso clínico rutinario, e incluso como potencial método de exploración poblacional –screening-.

La precisión y utilidad de éste método depende en última instancia de la comprensión del problema de transporte de la radiación en los tejidos, así como también el desarrollo de nuevas tecnologías que apoyen en esta tarea. La mejora de las condiciones de operación de las DOI permitirán entre otras aplicaciones avanzar en el conocimiento del funcionamiento del cerebro o la cura de ciertos tipos de cáncer.

## 1.4 Problemática

Este trabajo se centra en atender una de las limitaciones principales del simulador *mcmI*; la incapacidad de éste para realizar simulaciones con múltiples fuentes de iluminación y múltiples detectores entre los cuales se definen canales de sensado. Aunado a esto, se ha propuesto abordar el problema de flexibilización de definición de tejidos biológicos, el cual se realiza en *mcmI* bajo la definición de un conjunto de capas descritas por sus propiedades ópticas. En este sentido, el objetivo en este trabajo es proporcionar al usuario una definición más clara e intuitiva de un tejido (órgano) biológico. Por último, aunque el simulador *mcmI* puede ser ejecutado en diferentes sistemas operativos (con el archivo ejecutable específico para cada uno) y sus resultados son devueltos en un formato de texto plano, se busca incrementar la portabilidad tanto del simulador como de los archivos de resultados.

## 1.5 Preguntas de Investigación

En una geometría de sensado sencilla, un único iluminante irradia el tejido, y un único fotodetector recoge parte de la irradiación saliente, ya sea bajo un montaje de transiluminación como en un montaje de retrodispersión. Esta geometría de sensado provee de un único canal que enlaza a este emisor y a este receptor para dar una medida localizada de la respuesta electromagnética. La reconstrucción de volúmenes amplios implica tener varios canales interrogando varias localizaciones a un mismo tiempo. Esta multiplicidad de canales puede construirse de varias formas; por ejemplo, varios iluminantes y/o varios fotosensores, y su distinto acoplamiento. A esta forma de organizar fuentes de radiación, sensores y sus emparejamientos se les llama geometría de sensado, como ejemplo tenemos la Figura 1 que muestra la organización de múltiples fuentes. Algunas soluciones de simulación de transporte de radiación disponibles, como *mcmI*, sólo permiten expresar geometrías sencillas limitando el tipo de simulaciones que pueden ser llevadas a cabo.

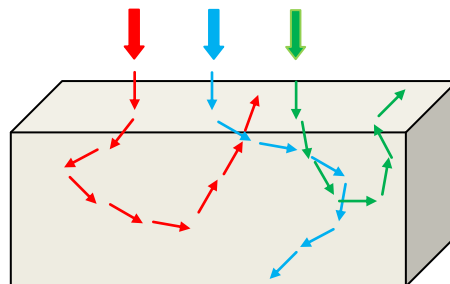


Figura 1 Múltiples fuentes de iluminación

Dada la necesidad de expresar geometrías de sensado complejas, ¿qué arquitectura de software podemos proponer que permita la definición flexible de geometrías de sensado multicanal?

¿Bajo qué condiciones ópticas y geométricas las relaciones entre las características ópticas y la absorción cumplen con el gold standard (*mcm*)? en este sentido, y más específicamente; la extensión a varios canales, ¿sigue respetando las leyes de la física asumiendo dispersión elástica sencilla y obviando la polarización de la radiación?

## 1.6 Hipótesis

Las geometrías de sensado de un solo canal son un caso específico de geometrías más complejas de sensado multicanal. La flexibilización de la arquitectura computacional para admitir geometrías complejas, no debe alterar el resultado de la simulación original cuyo origen viene determinado por la física y no por las limitaciones computacionales de la aproximación.

## 1.7 Objetivos

### 1.7.1 Objetivo general:

- Aumentar con capacidades de sensado multicanal el software de simulación de transporte de la radiación electromagnética existente (*mcm*).

### 1.7.2 Objetivos específicos:

- Generación de una arquitectura flexible de simulación de modelos ópticos mediante la técnica de Monte Carlo.
- Flexibilizar el acceso a la información resultado de la simulación permitiendo una exploración amplia del transporte de la radiación.
- Aceptar geometrías y configuraciones con múltiples puntos de irradiación y detección.
- Validar la arquitectura propuesta contra un estándar de oro (*mcm*) con un modelo existente de la cabeza adulta para neuroimagen.

## 1.8 Alcances y limitaciones

Este tema de investigación por el contexto en el que se desarrolla se vuelve multidisciplinario pues aborda temas referentes a óptica, biología (histología) y computación. Sin embargo esta tesis está centrada en la contribución computacional.

Como parte de las limitantes de esta tesis se tiene que no se consideran geometrías de tejido no homogéneas. Se mantiene la suposición física de la dispersión elástica –donde la luz que sale es del mismo color-. Además, se mantiene la ausencia del modelado de la polarización de la radiación –el ángulo de oscilación de la onda electromagnética con respecto a la dirección de propagación de la energía-, el régimen de Rayleigh –que asume una determinada relación de tamaño entre la longitud de onda y el tamaño de la partícula-, la propagación isotrópica en el plano perpendicular a la dirección del fotón, se ignora la difracción de la luz –el cambio en la dirección de propagación cuando hay ausencia de colisión-, así como también se supone la función Henyey-Greenstein como la función de anisotropía, no se admiten fenómenos de fluorescencia (remisión a diferentes longitudes de onda) ni fosforescencia (remisión en un tiempo posterior), entre otras.

Este tipo de simuladores requieren de recursos de cómputo como memoria y procesamiento. En la medida de lo posible se optimizará el uso de memoria y procesamiento, sin embargo no se considera como un objetivo en este trabajo.

Respecto a las simulaciones biológicas para verificación y validación se realizó una discretización arbitraria del tejido, se tienen simulaciones estáticas (modelo steady state) y para la caracterización óptica se utilizaron coeficientes de atenuación obtenidos *in-vitro* reportados en la literatura, pero que pueden diferir de sus análogos *in-vivo*.

## 1.9 Contribuciones

- Una nueva herramienta de simulación de modelos directos de transferencia de la radiación con capacidad de expresar geometrías de sensado complejas.
- Facilitación de la reutilización del conocimiento, mediante la flexibilización de los formatos de entrada y salida de datos. Como ejemplo, una vez que se tuviese el modelo de la piel, este podría ser utilizado directamente como la capa superior de un modelo de la cabeza adulta para neuroimagen.
- Una propuesta de definición de tipo de documento (XML DTD) para la expresión de tejidos y simulaciones.

## 1.10 Presentaciones

- Cuervo-Soto, Bibiana; Herrera-Vega, Javier; Garcés-Báez, Alfonso; Sucar, Luis Enrique; Treviño-Palacios, Carlos G.; Orihuela-Espina, Felipe, “*Facilitating tissue specification and complex sensing geometries in Monte Carlo radiation transport simulations – application to functional Near Infrared Spectroscopy (fNIRS)*”, XVI Reunión de neuroimagen, CIMAT, 17 Oct. 2014, Guanajuato, Gto., México.

---

## 1.11 Publicaciones

- Bibiana Cuervo-Soto, Javier Herrera-Vega, J. Alfonso del C. Garcés-Báez, Carlos Treviño-Palacios, Felipe Orihuela-Espina, “ *MOCARTS: A lightweight radiation transport simulator for easy handling of complex sensing geometries*”, International Symposium on Biomedical Imaging (ISBI), Clarion Congress Hotel, 13-16 Abril 2016, Praga, República Checa.



## 2. Fundamentos teóricos

Para poder entender mejor las secciones que conforman este tema de investigación es necesario comprender algunos términos que se mencionarán durante éste.

La **energía** se define como una propiedad (característica medible) de un sistema físico –un conjunto de componentes interactuando- que puede ser transferida entre los objetos físicos, por ejemplo, la materia a través de las 4 interacciones fundamentales –gravitacional, electromagnética, nuclear fuerte y débil [16].

La **radiación** es el proceso en el cual la energía es emitida desde una fuente como partículas u ondas. También se puede referir a la transmisión de esa energía. La radiación puede ser ionizante y no-ionizante. En la primera, si la radiación lleva suficiente energía para liberar electrones de los átomos, éstos provocan daños en los tejidos vivos que pueden resultar en la mutación, enfermedad por radiación, cáncer y la muerte. En la no-ionizante, si no lleva suficiente energía para liberar electrones de los átomos es parcialmente seguro pero aún puede causar quemaduras [16].

La **densidad ( $\rho$ )** se define como el número de átomos o moléculas por unidad de volumen. Cuando una sustancia está compuesta por un solo tipo de átomos o moléculas se le denomina sustancia pura, pero cuando está compuesta por una mezcla de diferentes sustancias puras y se requiere saber la cantidad de una sola de ellas (soluto) en la cantidad total de la sustancia que la contiene (solvente) entonces se habla de **concentración ( $\rho_n$ )**, la cual es la relación que existe entre la cantidad de soluto y la cantidad de solvente. Mientras más moléculas tenga una solución más radiación absorberá y menos radiación transmitirá [16].

Una **onda** es una vibración que se propaga (no transporta la materia). Las ondas transportan energía de una ubicación a otra [16] mediante el llamado movimiento ondulatorio. Para comprender el término de movimiento ondulatorio podemos tener como ejemplo el lanzar una piedra a un lago; en principio la superficie de éste no muestra movimiento alguno, sin embargo al arrojar una piedra se produce una perturbación a la tranquilidad del líquido, lo cual se puede observar en el movimiento del agua en forma de ondas o vibraciones alrededor del punto en que la piedra entra en contacto con el agua, como se muestra en la Figura 2. En las ondas, las partículas no se desplazan solo oscilan, y es la energía la que se transmite. Dependiendo de la dirección de oscilación con respecto a la dirección de propagación estas ondas pueden ser transversales, longitudinales, elípticas o de superficie y de torsión [17].



Figura 2 Incidencia de una piedra sobre un lago. Figura obtenida de [18].

Quizás más relevante que esta clasificación por la dirección de oscilación es la clasificación respecto al medio de propagación. Las ondas pueden ser mecánicas o electromagnéticas. En las primeras para poder propagarse requieren de un medio elástico, mientras que las electromagnéticas no, pudiéndose propagar en el vacío. Aunque la imaginología médica puede estar basada en radiación mecánica (ultrasonido), en esta tesis nos centraremos exclusivamente en la radiación electromagnética.

En la Figura 3 se muestran los elementos principales de una onda:

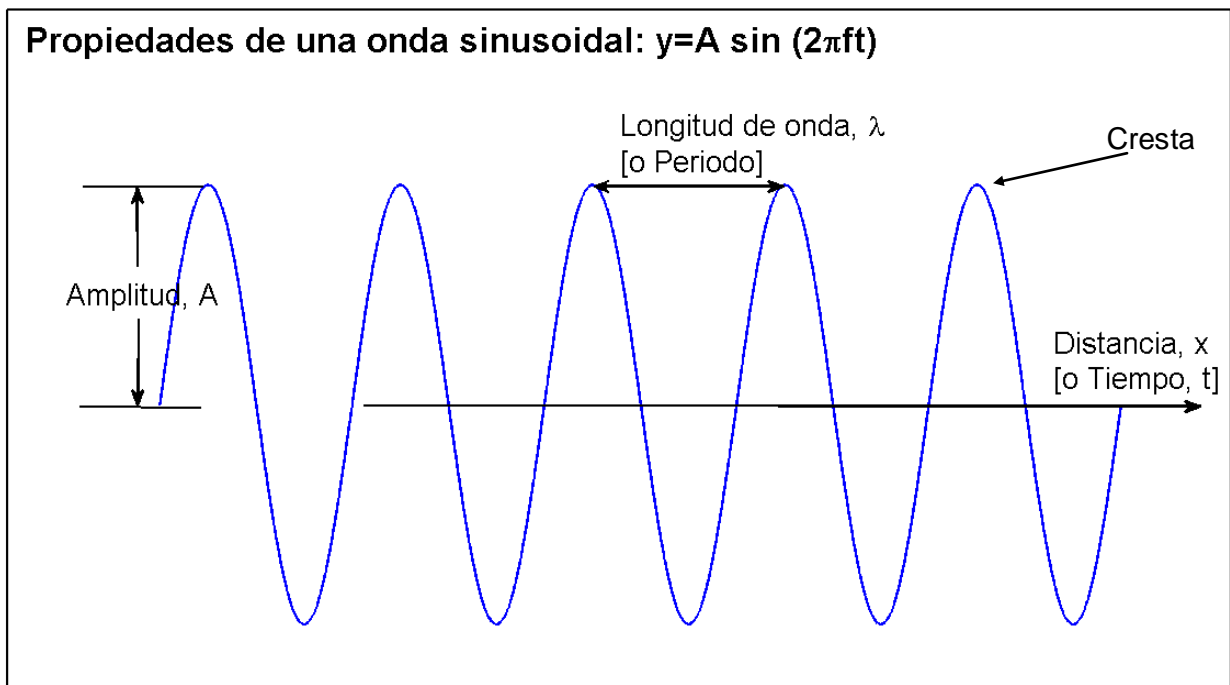


Figura 3 Propiedades de la onda Seno.

**Cresta (Peak):** es el punto de máxima amplitud de la onda.

**Amplitud (A):** es la distancia vertical y el punto medio de la onda.

**Periodo (T):** es el tiempo que tarda una onda en ir entre dos puntos de máxima amplitud.

**Frecuencia (f):** es el número de veces que se repite la onda por unidad de tiempo.

**Longitud de onda ( $\lambda$ ):** Es la distancia que hay entre dos crestas consecutivas.

**Velocidad de propagación (v):** es la velocidad a la que se desplaza el movimiento ondulatorio y depende de las propiedades del medio.

La ecuación (1) describe el comportamiento básico de una onda:

$$v = \lambda f \quad (1)$$

La cual relaciona tres de los parámetros anteriores; si consideramos la relación entre el periodo y la frecuencia (ecuación (2)):

$$T = 1 / f \quad (2)$$

## 2.1 Teoría de la luz y el color

La definición de **luz** no es trivial; a menudo se refiere a la región visible del espectro electromagnético [16]. Algunas veces es usada para referirse a otros rangos del espectro, por ejemplo, luz infrarroja, y por tanto se refiere a la radiación electromagnética [19]. La luz se comporta como una onda electromagnética (onda) y como una partícula (fotón).

Tradicionalmente, han existido dos teorías sobre la composición de la luz. La primera de ellas propuesta por Isaac Newton llamada teoría corpuscular que afirma que la luz está compuesta de pequeñas partículas o cuantums, denominados fotones, los cuáles son una cantidad de energía que puede ser absorbida por un electrón. Por otro lado, Christian Huygens considera que la luz se comportaba como un fenómeno ondulatorio [20]. Finalmente, Albert Einstein concilia ambas teorías demostrando la dualidad de la composición de la luz [19].

En un rayo de luz, los rayos pueden ser paralelos y se denomina luz colimada, o los rayos pueden tener diferentes direcciones a lo que se le conoce como luz difusa. La polarización de la luz se refiere a la dirección de vibración del rayo de luz con respecto a su dirección de propagación. La polarización puede ser plana, circular o elíptica. En este trabajo no se considera la polarización de la luz. Dada la naturaleza de la luz como una onda, esta es caracterizada por su longitud de onda en el espectro electromagnético. La Figura 4 muestra la conceptualización de las diferentes regiones del espectro.

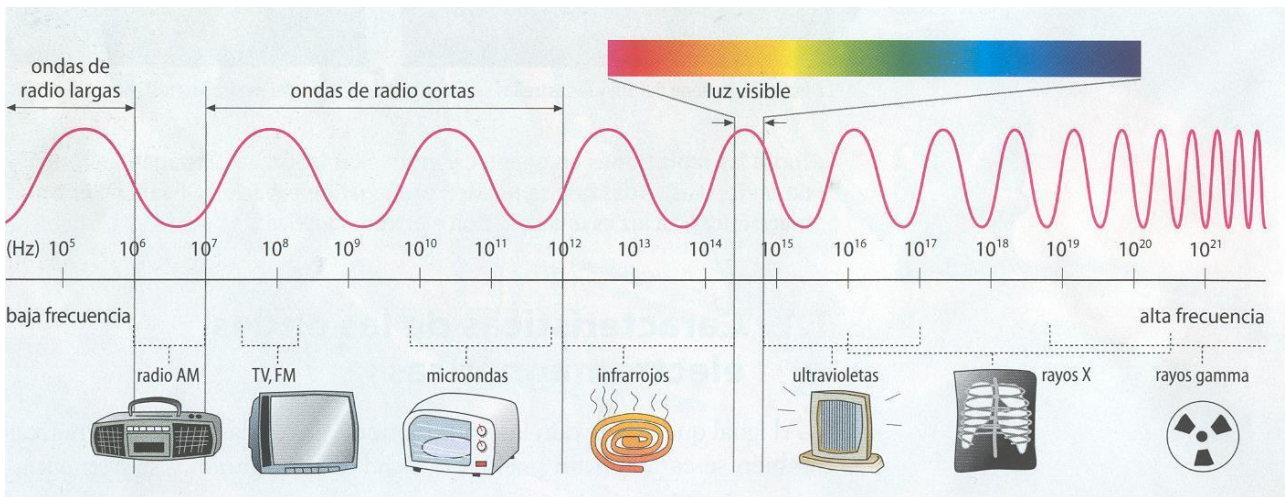


Figura 4 Espectro Electromagnético. Imagen extraída de [21]

La región del infrarrojo tiene un interés especial en biomedicina por una combinación de factores como ser una radiación no ionizante (léase no tiene suficiente energía como para liberar electrones), la reducida absorción y esparcimiento de la luz por los tejidos en la llamada **ventana óptica**, y la extrema sencillez de la instrumentación requerida para su sensorado.

Cuando la radiación toca sobre un medio se observan fenómenos como el de **absorción**, lo cual se entiende como la fracción de radiación depositada en el medio, la **reflexión**, que es la fracción de radiación reflejada en una dirección de retroceso con respecto a la superficie del medio, la **transmisión**, definida como la fracción de radiación que traspasa el medio y el **esparcimiento** que se entiende como la desviación de la onda en una diferente dirección y la cual está relacionada con la concentración de moléculas en el medio. Nótese que la reflexión es un caso particular de esparcimiento. La probabilidad de que el haz de luz tome otra dirección forma una distribución esférica que puede ser isotrópica o anisotrópica.

Cuando una onda choca con el límite de un medio diferente puede cambiar su velocidad y su dirección de propagación, si la onda pasa al siguiente medio, se le llama **refracción** y cuando esa onda es rebotada se le denomina reflexión. La ley de Snell gobierna la relación entre los ángulos de incidencia y la velocidad de las ondas como se ilustra en la Figura 5.

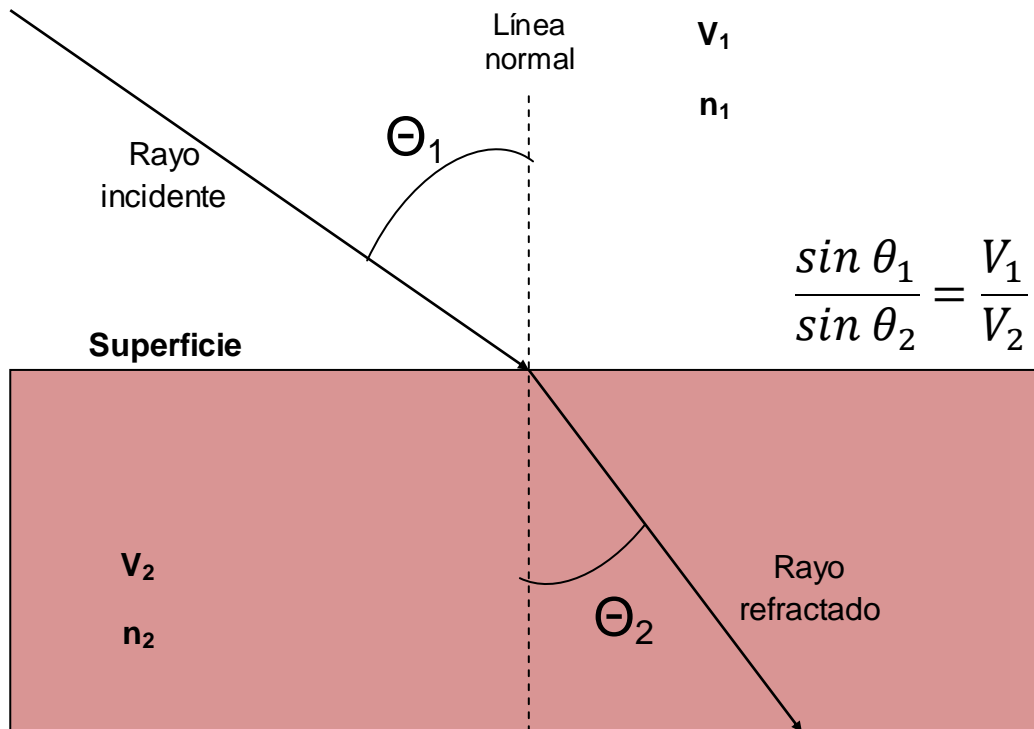


Figura 5 Ley de Snell.

Esta ley aplica tanto para las ondas mecánicas como para las electromagnéticas y determina el índice de refracción del medio. El índice de refracción  $n$  en un medio es la proporción de la velocidad de la luz en el vacío  $c$  sobre la velocidad de la luz en el medio  $v$ , tal como se muestra en la ecuación (3):

$$n = \frac{c}{v} \quad (3)$$

Cuando un haz de luz choca con un tejido y este es reflejado con el mismo ángulo de la luz incidente pero en una dirección complementaria sobre el vector normal a la superficie, se dice que es un fenómeno de reflexión especular. Cuando rebota en cualquier otra dirección se denomina reflexión difusa. Adelantándonos un poco, el término difuso en óptica indica que las direcciones de propagación de las distintas ondas de un haz no son paralelas.

La Figura 6 muestra las curvas de absorción de algunos compuestos del tejido biológico en función de la longitud de onda. Es de interés particular las mostradas en la ventana óptica en el Infrarrojo cercano. Se puede observar por ejemplo que la sangre oxigenada ( $O_2Hb$ ) incrementa su curva de absorción entre las longitudes de onda 600-1000nm mientras que la sangre desoxigenada

o reducida (HHb) decreta dicha curva en las mismas longitudes de onda. Estas dos especies de hemoglobina por ejemplo, nos permiten explorar la hemodinámica cerebral y a partir de ésta, la actividad cerebral subyacente [22].

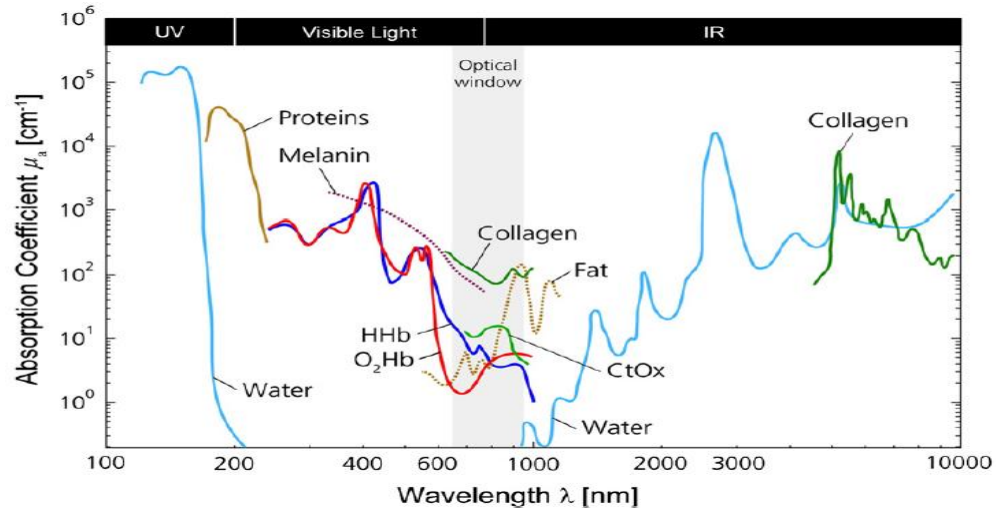


Figura 6 Curvas de absorción de algunos compuestos biológicos en la ventana óptica de IR. Figura extraída de [23].

## 2.2 Imagen Óptica Difusa (DOI)

En la interacción de la luz con la materia ocurren dos fenómenos principales que son la absorción y el esparcimiento. En la absorción, parte de la energía de la luz es absorbida por las moléculas que componen a la materia. La energía de la onda es reducida ya sea por un decremento en su velocidad y por tanto una alteración en su longitud de onda, o por un decremento en su intensidad o amplitud de la onda dual. El tejido puede disipar esa energía en forma calorífica, remitiéndola a diferente longitud de onda (fluorescencia) o posteriormente (fosforescencia). En el esparcimiento, cuando un fotón colisiona con una molécula ésta induce un cambio en la dirección de viaje del fotón [3]. La nueva dirección depende de varios parámetros físicos como la relación entre el tamaño de la molécula y la longitud de onda, lo cercano que se produzca la colisión con respecto a un núcleo atómico, y la particular composición electroquímica de la molécula. Esta nueva dirección es altamente impredecible como fenómeno aislado, existiendo modelos más o menos fidedignos (por ejemplo, la dispersión de Mie asume una forma de partícula esférica [24], pero su distribución integral sobre muchos eventos es característica, siendo los tejidos biológicos altamente dispersantes hacia adelante [25], lo que facilita su modelado mediante distribuciones y/o funciones conocidas como por ejemplo la distribución de Henyey-Greenstein.

En el caso particular de una dispersión en la frontera entre dos capas de tejido con diferente estructura histológica, y a menudo con diferentes propiedades ópticas el esparcimiento toma la forma de reflexión o difracción, y viene determinado por la ley de Snell anteriormente descrita. El

índice de refracción determina la velocidad de la luz en el medio y por ende, el ángulo de la nueva dirección de propagación.

Este comportamiento conceptualmente sencillo requiere de la formulación de modelos que expliquen dicha interacción. Se han propuesto diversos modelos matemáticos que intentan explicar el comportamiento de la luz con la materia, problema conocido como transporte de la radiación. El más general es el descrito por las leyes de Maxwell. Sin embargo, debido a la intractabilidad del problema, se han derivado modelos simplificados que omiten fenómenos con la finalidad de obtener una aproximación a la física de la interacción luz-materia. Estos modelos se pueden dividir en analíticos y estocásticos [8]. En los analíticos está la Ecuación de Transferencia de Radiación (RTE), sin embargo, ésta solo ha podido ser aplicada en casos muy simples. Por lo cual se han derivado de esta otros modelos aún más sencillos como la teoría de difusión [4], [5], [7], la teoría de Kubelka-Munk [5] o la ley de Beer-Lambert [3], [7], entre otros. Estas soluciones deterministas son exactas siempre y cuando se cumplan las suposiciones de las que parten, pero estas suposiciones, a menudo excesivamente restrictivas, no suelen coincidir con las de un tejido biológico.

Como alternativa surgen métodos probabilistas que sacrifican la exactitud de la respuesta a cambio de poder explorar escenarios menos restrictivos con la promesa de convergencia a la solución en simulaciones lo suficientemente densas. Dentro de los modelos estocásticos se encuentran las caminatas aleatorias y las simulaciones basadas en la técnica de Monte Carlo (MC) [10]. Este último, central a esta tesis, se basa en el muestreo de distribuciones de probabilidad para determinar el comportamiento de los fotones en el tejido. En general, las técnicas de MC diseñan una función probabilística que predice el estado siguiente más probable del sistema, de forma que en el límite, la probabilidad de visitar un estado coincida con la observada en la realidad. Cuando el modelado analítico de una determinada función es altamente complejo, las técnicas de MC pueden obtener aproximaciones sin necesidad de requerir una ecuación explícita subyacente.

## 2.3 Caracterización óptica de tejidos biológicos

Para ser funcionales, los modelos de transporte de la radiación requieren de la especificación de **parámetros ópticos** como los descritos a continuación:

- **Índice de refracción ( $n$ )** que indica el cambio de velocidad de la luz al pasar de un medio a otro.
- **Coefficiente de absorción ( $\mu_a$ )** que indica la eficiencia macroscópica del tejido para absorber energía de un haz de fotones (este puede ser a nivel macroscópico o a nivel molecular).
- **Coefficiente de esparcimiento ( $\mu_s$ )** que indica la capacidad macroscópica del tejido para dispersar un fotón (al igual que el coeficiente de absorción, este puede ser a nivel macroscópico o molecular).

- **Función de Anisotropía (g)** la cual indica la distribución de probabilidad de la nueva dirección del fotón en cada evento de esparcimiento. Bajo la suposición de distribución isotrópica en el plano perpendicular a la dirección de propagación del fotón, la función de anisotropía puede ser caracterizada por un único parámetro, con frecuencia el coseno del ángulo que forma la nueva dirección con respecto a la anterior ( $g = \cos \theta$ ).

Para ilustrar estos conceptos, en la Figura 7 se muestra de forma esquemática el comportamiento de la luz sobre un tejido que ha sido irradiado con una fuente de luz. En esta representación se ilustran los siguientes fenómenos [10]:

- 1.- Parte del haz de fotones se refleja, es decir, se remite desde la superficie del tejido o el medio, ya sea tras haber interactuado con el tejido (reflectancia difusa) o sin tener alguna interacción (reflectancia especular). Modelos como el de Fresnel [8] modelan este último. Ambos componentes especular y difuso pueden ser separados por modelos como el dicromático [26], permitiendo aislar estrictamente la luz que ha interactuado con el tejido.
- 2.- Parte del haz de fotones es absorbido como resultado de la interacción con las moléculas que componen el tejido. El análisis espectroscópico de la signatura de estas pérdidas permite inferir la composición del tejido.
- 3.- Parte del haz de fotones se esparce por el tejido, impidiendo que muchos de los fotones irradiados vuelvan a emerger en la superficie por lo que la intensidad de la radiación remitida en superficie es menor que la irradiada originalmente.
- 4.- Finalmente, parte del haz de fotones logra pasar todas las capas del modelo, entendiéndose este fenómeno como transmitancia.

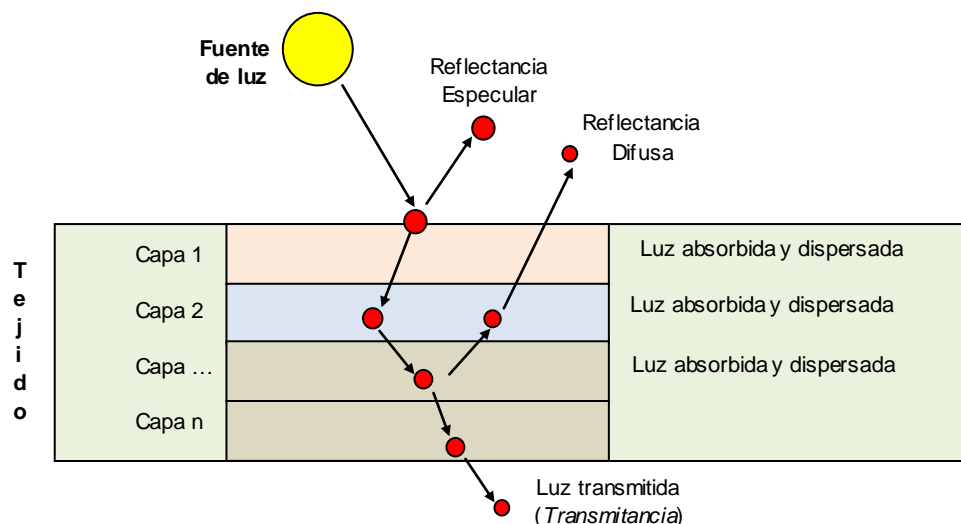


Figura 7 Comportamiento de un fotón en el tejido.

Además de los elementos anteriores, los modelos de transporte de radiación requieren de un quinto parámetro geométrico crítico; el grosor óptico de las diferentes capas de tejido. La ley de

Beer-Lambert (ecuación (4)) predice la atenuación de la luz al pasar por un tejido homogéneo de un determinado grosor.

$$I = I_0 e^{\varepsilon \cdot c \cdot l} \quad (4)$$

Donde  $I_0$  es la intensidad de luz irradiada al tejido,  $\varepsilon$  es el coeficiente de extinción molar,  $c$  la concentración del compuesto y  $l$  el grosor del tejido.

## 2.4 Método de Monte Carlo

El método de Monte Carlo es un método estadístico que proporciona soluciones aproximadas a problemas matemáticos. Este método debe su nombre a un casino de Monte Carlo en Mónaco por ser la capital del juego del azar, al tomar una ruleta como un generador de números aleatorios. Esta técnica toma una muestra del fenómeno estocástico y se estiman los valores promedio a partir de ésta. Así los valores muestreados se obtienen a partir de distribuciones de probabilidad de cada variable. Entre mayor sea el número de experimentos, es mayor la probabilidad de tener un análisis más preciso.

En este trabajo el método de Monte Carlo se usó para simular el transporte de la radiación en tejidos biológicos a través del muestreo de dos eventos relacionados con la trayectoria de los fotones:

- el tamaño del paso (el cual indica la longitud del desplazamiento del fotón), y
- la dirección del fotón después de un evento de esparcimiento.

Estos eventos se definen de acuerdo a distribuciones de probabilidad particulares. En el caso del tamaño del paso se utiliza una distribución de probabilidad exponencial como se muestra en la Figura 8, la cual indica que es más probable un tamaño del paso corto que un tamaño del paso largo.

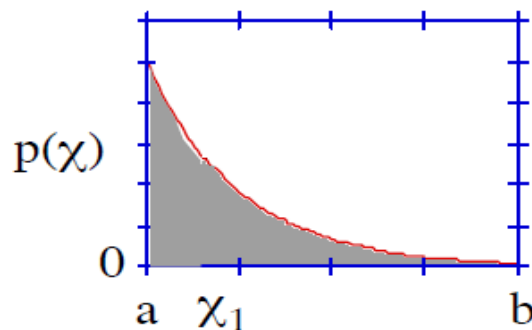


Figura 8 Distribución de probabilidad exponencial para el tamaño del paso, donde  $p(x)$  es la probabilidad del tamaño del paso,  $a$  y  $b$  son la distancia máxima y mínima (respectivamente) que el tamaño del paso puede tomar y  $x_1$  es un ejemplo de un valor muestreado. Figura extraída de [7].

En el caso de la dirección del fotón esquematizada en la Figura 9, se usa la distribución de probabilidad Henyey-Greenstein la cual estima el ángulo de deflexión de la partícula con respecto a su dirección de propagación, y que ya se conoce que es una buena aproximación a las observaciones en tejidos biológicos [5].

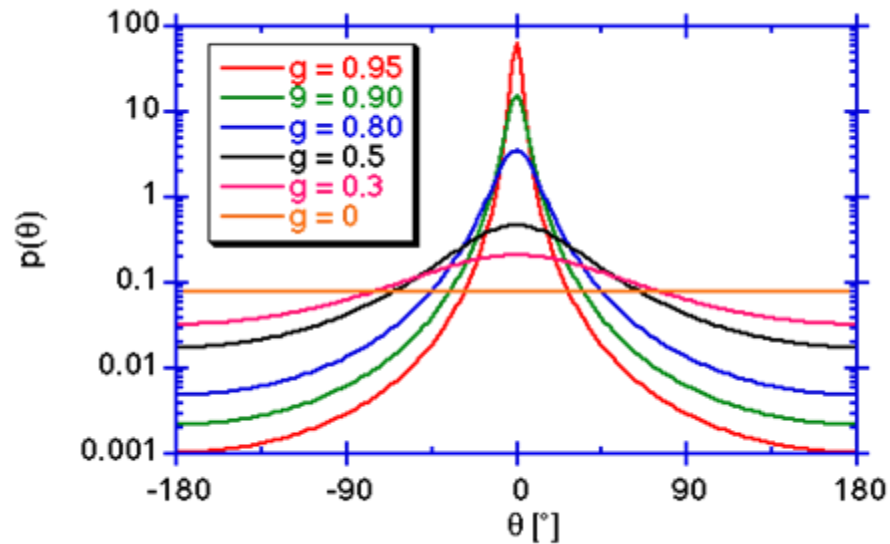


Figura 9 Distribución angular de probabilidad Henyey-Greenstein para obtener la dirección del fotón. En el eje de ordenadas se representa la probabilidad de tomar el valor del coseno de  $\theta$  indicado en la abscisa. Figura extraída de [27].



### 3. Software de Transporte de la radiación

Existen varios sistemas que permiten conocer el comportamiento de la luz con la materia. Sin ser exhaustivos, entre ellos se encuentran:

***mcmI***: Monte Carlo Modeling of Light Transport in Multi-layered Tissues, desarrollado en el Oregon Medical Laser Center [7]. Basado en modelos estocásticos, presenta entre sus limitaciones críticas un único punto de entrada de la irradiación, y entre otras no críticas, la rigidez de los formatos de entrada y salida de datos. Este software se ha convertido en un gold standard en las simulaciones de transporte de radiación en tejidos biológicos [28] y en esta tesis se toma como punto de partida por lo que lo detallamos más en la siguiente sección; aunque no obstante la solución de esta tesis es reimplementada como veremos en el capítulo 3 desde cero.

**NIR-FAST**: Near Infrared Fluorescence and Spectral Tomography, es un software que simula la propagación de la luz en tejidos biológicos basado en la solución de la ecuación de difusión [29] mediante el método de elementos finitos [8]. Fue desarrollado por el Optics in Medicine Laboratory at Dartmouth.

**TOAST**: Basado en modelado de elementos finitos. Fue desarrollado originalmente para el modelado y la reconstrucción de la Tomografía Óptica Difusa (Diffuse Optic Tomography, DOT por sus siglas en inglés), fue desarrollado en el University College London [30]. Una desventaja es que no hay medios para derivar trayectorias de fotones individuales.

**penMesh**: Simulación del transporte de la radiación en una malla de geometría triangular usando la técnica de Monte Carlo. Fue desarrollado por U. S. Food and Drug Administration (FDA), Center for Devices and Radiological Health, Office of Science and Engineering Laboratories, Division of Imaging and Applied Mathematics [31].

**MCX**: Monte Carlo eXtreme. Es un simulador que utiliza la técnica de Monte Carlo diseñado para la modalidad de espectroscopía de tiempo de vuelo del fotón. Usa una técnica de cómputo paralelo basado en una unidad gráfica de procesamiento GPU. Fue desarrollado en la División de óptica del Martinos Center for Biomedical Imaging, Massachusetts General Hospital/Harvard Medical School [32].

**CUDAMCML**: GPU Monte Carlo for Graphics Cards. Es una implementación paralelizada de *mcmI* que toma las ventajas de usar una tarjeta gráfica NVIDIA que permite hacer simulaciones mucho más rápidas. Fue desarrollado por el Department of physics Lund University, Sweden [33]. Este simulador mantiene invariantes la definición de modelos y datos de entrada/salida.

#### 3.1 *mcmI*

El software *mcmI* (Monte Carlo Multi-Layer) [7] utiliza una solución estocástica para resolver el problema de transporte de la radiación. Este software, que implementa el método de Monte Carlo, considera una fuente que emite un haz de fotones incidente perpendicularmente sobre un tejido

formado por múltiples capas homogéneas. Cada capa es descrita por sus parámetros ópticos como: índice de refracción ( $n$ ), coeficiente de absorción ( $\mu_a$ ), coeficiente de dispersión ( $\mu_s$ ), factor de anisotropía ( $g$ ) y geométricos como el espesor de la capa ( $d$ ). El tejido es modelado como una rejilla tridimensional en donde cada celda tiene asociada las propiedades ópticas de la capa correspondiente. En cada evento de extinción se determina la cantidad de energía que es absorbida por el tejido, el ángulo de la nueva trayectoria del fotón así como el tamaño del paso antes de sufrir un nuevo evento de extinción. Este proceso continúa hasta que el fotón pierde toda su energía o sale del tejido (reflectancia o transmitancia). La Figura 10 muestra el diagrama de flujo de este proceso.

1. Se emite un fotón
2. Se le asigna el desplazamiento que tendrá
3. Se analiza si el fotón llegó al límite de la capa
4. Se establece si hubo reflectancia o transmitancia.
5. Se evalúa el peso del fotón, si es demasiado pequeño en ese momento cesa su actividad, de lo contrario se le asigna otro desplazamiento.
6. Lo anterior se repite para todos los fotones que se simulan.

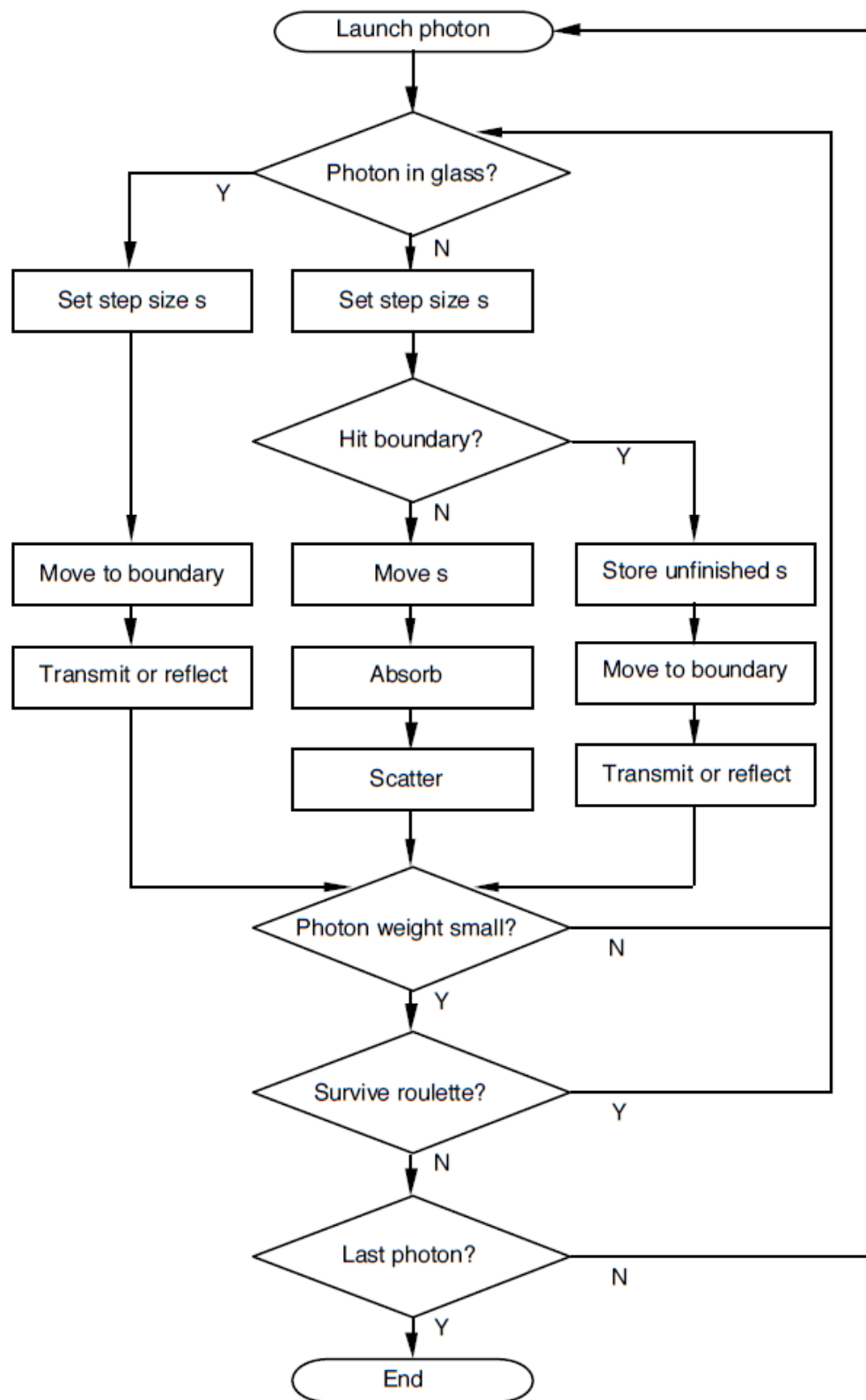


Figura 10 Diagrama de Flujo del simulador *mcm1*. Figura extraída de [8]

Este simulador está implementado en lenguaje ANSI C bajo el paradigma de programación estructurada. Recibe como entrada un archivo en texto plano con la definición de las capas del tejido a simular descritas mediante sus propiedades ópticas. Tras la ejecución de una simulación se obtienen como datos de salida la reflectancia (especular y difusa), la transmitancia y la absorción en cada celda de la rejilla definida. No obstante, de manera interna, *mcmI* lleva registro de toda la absorción y el flujo de fotones interno al tejido.

*mcmI* tiene sus inicios en 1989 en un código escrito en el lenguaje Pascal por Steven L. Jacques y Marleen Keijzer y ha pasado por diversas modificaciones hasta llegar a su versión más reciente en 1995 [7], [28], [34]- [35]. Desde entonces, se ha convertido en uno de los simuladores más populares en la comunidad científica, siendo citado en más de 600 publicaciones [36]. Actualmente, ya tiene una versión implementada sobre GPU para aumentar su rendimiento [33], pero para esta tesis partimos de la versión no paralelizada.



## 4. Metodología

Por otro lado, a través del tiempo, el software de computadora ha evolucionado. Ha tenido un cambio que se muestra, por ejemplo, cuando éste se adapta a un nuevo ambiente, cuando se corrigen errores, cuando se solicita adecuar nuevas funciones o cuando se requiere de una reingeniería para proporcionar beneficios de un contexto moderno.

La **Ingeniería del Software** establece los estándares y modelos que facilitan la comunicación entre el usuario final y el desarrollador, así como también provee los métodos, herramientas y procedimientos en la construcción del software. Los métodos indican cómo construir técnicamente el software, las herramientas proporcionan un soporte automático o semiautomático para los métodos (herramientas CASE) y los procedimientos son la manera en que los métodos y las herramientas se unen para facilitar el desarrollo del software. El término Ingeniería del Software nace para intentar solucionar la crisis del software alrededor de los años 70's, donde la calidad del software era mala y se excedían los tiempos de entrega y presupuestos, entre otros [37].

La **Programación Orientada a Objetos** (POO) [38] nace a finales de los 70's no para resolver la crisis del software pero sus características ayudaron a solventar ésta. Utiliza **objetos** como el elemento fundamental para el desarrollo de la solución. Se entiende el término objeto como una abstracción de algún hecho o constructo del mundo real, el cual tiene atributos que representan sus propiedades y métodos que emulan su comportamiento o actividad. Todas estas propiedades y métodos se agrupan en **clases**, se dice que un objeto es una instancia de una clase.

**XML** es un lenguaje de marcas utilizado para almacenar datos de forma legible [39]. Se propone como un estándar para el intercambio de información estructurada entre diferentes plataformas. Permite la compatibilidad entre sistemas para compartir la información de manera segura, fiable y fácil. Busca dar solución al problema de expresar información estructurada de la manera más abstracta y reutilizable posible. Decir que la información se muestra de manera estructurada quiere decir que se compone de partes bien definidas, y que esas partes se componen a su vez de otras partes. Entonces se dice que se elabora un árbol con trozos de información, denominados etiquetas que son los elementos del árbol.

### 4.1 Diseño e implementación de MOCARTS

La experimentación necesaria para soportar la hipótesis central de este trabajo conlleva como necesidad colateral la implementación de un software de simulación de transporte de la radiación que permita simulaciones con geometrías de sensado multicanal. La propuesta de esta tesis a esta necesidad es MOCARTS (**M**onte **C**arlo **R**adiation **T**ransport **S**imulation) que presentamos en esta sección.

Como se mencionó anteriormente, el desarrollo de MOCARTS tuvo como base el simulador *mcmI*. Este fue reprogramado bajo el paradigma de Programación Orientada a Objetos (POO) para

el lenguaje y plataforma Java, con la finalidad de proporcionar flexibilidad al nuevo simulador y permitir su crecimiento de forma estructurada en futuros proyectos.

Tres puntos importantes fueron considerados en el diseño e implementación de MOCARTS. Primeramente, éste debería ser capaz de simular múltiples fuentes de luz y múltiples detectores. Con la capacidad de poder ser ubicados libremente en cualquier punto de la superficie del tejido, como se describe en la sección 4.9. Geometrías de sensado multicanal. Por otro lado, se buscó un diseño que permitiera la flexibilización de la definición de tejidos biológicos. Haciendo la definición de un tejido de forma jerárquica y por ende más clara e intuitiva, tal como se muestra en la sección 4.2. Diseño del sistema. Finalmente, se buscó el incremento en la portabilidad de la herramienta así como de la información de entrada/salida usada por el simulador, la cual se detalla en la sección 4.7. Definición en XML de una simulación.

## 4.2 Diseño del sistema

Se ha diseñado una arquitectura orientada a objetos para conformar un software de simulación de transporte de la radiación electromagnética en tejidos biológicos.

Esta arquitectura se ha diseñado teniendo como base el software de simulación *mcmI* [7] reimplementando el motor en el lenguaje Java, adaptando la solución al paradigma orientado a objetos y haciéndole ampliaciones a dicho sistema, como la modularización con la finalidad de tener independencia y flexibilidad en el modelado y fácil mantenimiento del mismo. La Figura 11 muestra los paquetes de software que conforman la arquitectura del sistema.

Aunque no es objetivo central de esta tesis esta nueva pieza de software nace con la intención de ampliarse en trabajos posteriores, por lo que se dedicó un esfuerzo particular a obtener un diseño extendible. Por ejemplo, aunque no se llegó a implementar, pero se diseñaron varios elementos para la interfaz de usuario como veremos. Se favoreció una nomenclatura de clases y campos en inglés para facilitar su posterior publicación en los foros apropiados.

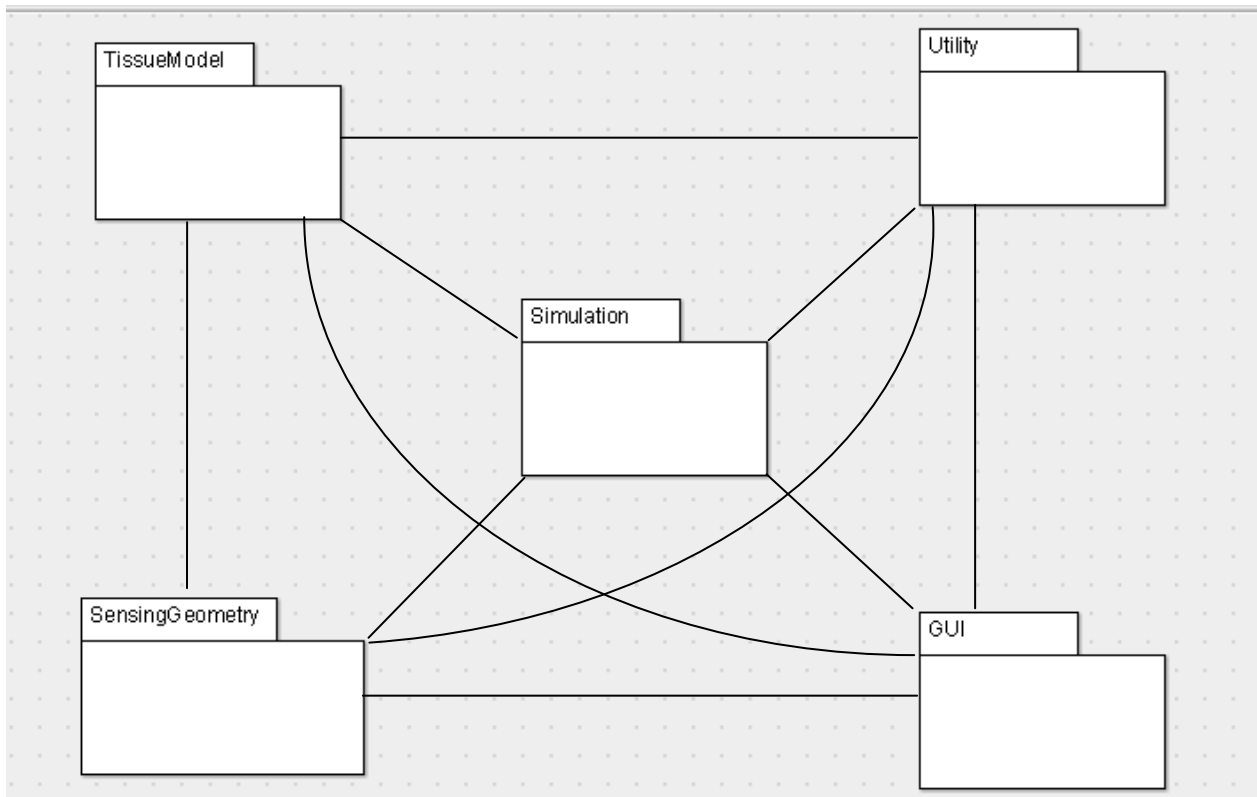


Figura 11 Modularización del sistema MOCARTS.

La descripción de los paquetes con que cuenta el sistema se detalla a continuación:

- TissueModel:** Este paquete incorpora todas las clases para el modelado del tejido, órganos y medios biológicos. El Transporte de la radiación se simula sobre un modelo de tejido de varias capas (con independencia de que estos tejidos provengan de un órgano determinado o varios órganos). Con respecto a la especificación del tejido que originalmente propone *mcmI*, este nuevo paquete mantiene por ahora las capas planas apiladas, pero ya pone los cimientos para la posterior ampliación a geometrías de tejido más complejas. En la Tabla 1 se describen Las clases que componen este paquete.

Tabla 1 Conjunto de clases que conforman el paquete TissueModel.

Clase	Descripción
BiologicalSlab	Es una colección de rebanadas biológicas (BiologicalSlice). Todas las simulaciones de transporte de la radiación se hacen sobre esta clase.
BiologicalSlice	Es la generalización de cualquier tipo de corte biológico, ya sea órgano, tejido, capa, etc.
Organ	Es una parte biológicamente diferenciada del cuerpo que cumple una función específica. A menudo está conformado por un conjunto de tejidos.
Tissue	Es una agregación de células morfológicamente similares. Se puede considerar que está conformada como un conjunto de capas.

Layer	Representa un bloque de tejidos. Para la simulación del transporte de la radiación este es el bloque básico de construcción.
HomogeneousLayer	Una capa básica (frecuentemente compuesta por un tipo de células simple) el cual para el propósito de simulación del transporte de la radiación sus propiedades ópticas son homogéneas.
HeterogeneousLayer	Esta clase fue definida dada la necesidad de tener una definición más realista de un tejido biológico. El objetivo de esta clase es el manejo de capas con propiedades ópticas no homogéneas. (No implementado en el alcance de esta tesis) <sup>1</sup>
OpticalProperties	Un conjunto básico de propiedades ópticas en una ubicación dada. Esta ubicación puede ser en realidad un punto simple en la capa, o referirse a toda una capa a la vez.
Coefficient	Un coeficiente representa un coeficiente de extinción, ya sea de absorción o dispersión, aplicable a una molécula o a un medio entero, independientemente de si es específico o macroscópico.
MolecularSpecificCoefficient	Generalización para el manejo de los coeficientes específicos.
MolecularSpecificAbsorptionCoefficient	Esta clase se encarga de la lectura, almacenamiento y manejo de los coeficientes de absorción específicos que se definan en un modelo.
MolecularSpecificScatteringCoefficient	.Esta clase se encarga de la lectura, almacenamiento y manejo de los coeficientes de esparcimiento específicos que se definan en un modelo.
MolecularMacroscopicCoefficient	Un coeficiente de extinción macroscópico de una molécula dentro de un medio, el cual es el producto de la concentración de una molécula multiplicado por el coeficiente de extinción específica de esa molécula.
MolecularMacroscopicAbsorptionCoefficient	Un coeficiente de absorción macroscópico de una molécula dentro de un medio.
MolecularMacroscopicScatteringCoefficient	Un coeficiente de dispersión macroscópico de una molécula dentro de un medio
BiologicalSliceMacroscopicCoefficient	Esta clase representa la extinción para toda una BiologicalSlice (ya sea órgano, tejido, o capa). En una rebanada biológica, puede haber varios cromóforos o pigmentos que contribuyen a la extinción, además de, por supuesto, una extinción global de fondo debido a los elementos modelados no explícitamente.
BiologicalSliceMacroscopicAbsorptionCoefficient	Representa el coeficiente de absorción macroscópico de una rebanada biológica (BiologicalSlice, ya sea órgano, tejido o capa).
BiologicalSliceMacroscopicScatteringCoefficient	Representa el coeficiente de dispersión macroscópico de una rebanada biológica (BiologicalSlice, ya sea órgano, tejido o capa).
AnisotropyFunction	La función de anisotropía representa la función de probabilidad angular para el ángulo de deflexión en un evento de dispersión.
HenyeGreensteinAnisotropyFunction	Esta clase implementa la función de anisotropía Henye-Greenstein para calcular el ángulo de deflexión en un evento de dispersión. Esta función es bien conocida pues solo depende de un simple parámetro, el factor de anisotropía $g$ .

<sup>1</sup> Posterior al desarrollo de esta tesis (y previo a la evaluación) esta clase ya ha sido implementada por Javier Herrera Vega y constituye la primera ampliación de MOCARTS.

- **Simulation:** Es el paquete central para MOCARTS. Este paquete incorpora todas las clases necesarias para llevar a cabo una simulación del transporte de radiación en los tejidos. Este paquete es responsable de la gestión de toda la información sobre la red de simulación y los modelos físicos de la radiación, así como propiamente de la ejecución de la simulación del proceso físico. Este paquete reproduce el funcionamiento de la herramienta original *mcmI*. No obstante también presenta mejoras sustanciales. El encapsulamiento de determinadas funciones permite que en ampliaciones futuras estas puedan ser reemplazadas con facilidad para explorar modelos más genéricos de simulación. La Tabla 2 muestra las clases que integran este paquete.

Tabla 2 Conjunto de clases que conforman el paquete Simulation

Clase	Descripción
InParams	Clase que almacena los objetos (datos) necesarios para la realización de una simulación: Modelo de tejido, Geometría de Sensado y Parámetros de simulación.
OutParams	Esta clase mantiene las estructuras de datos de salida como: matrices de absorción, reflectancia, transmitancia, entre otras.
Photon	Un solo fotón. Esta clase está inspirada en la estructura de datos de <i>mcmI</i> : PhotonStruct. Lleva un registro de la energía restante del fotón, de su posición actual, su dirección de propagación, así como también de la fuente de luz de la que proviene.
Simulation	Esta clase representa una simulación del transporte de la radiación en tejido biológico. Es la parte central del paquete Simulation puesto que es el responsable de ejecutar el proceso de simulación con la técnica de Monte Carlo dando la información correspondiente.
SimulationGrid	Esta clase representa la estructura de la rejilla para guardar los resultados de la simulación.
PolarSimulationGrid	Implementa una rejilla usando un sistema de coordenadas polares, tal es el caso del software de simulación <i>mcmI</i> .
CartesianSimulationGrid	Esta clase implementa una rejilla usando un sistema de coordenadas cartesianas. Tal es el caso de MOCARTS.
ReflectanceModel	Es la clase que asume el modelo de transmisión de la luz (reflexión o refracción) que rige el comportamiento de la luz entre los límites entre capas de tejido.
Fresnel	Implementa la ley de Fresnel de acuerdo al siguiente comportamiento: cuando un rayo de luz viaja a través de un medio caracterizado por un índice de refracción $n_1$ y choca con otro medio de índice de refracción $n_2$ , una parte del rayo es reflejado y otra parte transmitido a otro medio.

- SensingGeometry:** Este paquete encapsula las clases necesarias para el modelado de las fuentes de radiación y fotodetectores y de las geometrías de sensado –el emparejamiento de los anteriores-. Las capacidades de *mcm1* a este respecto son rudimentarias, y esta contribución es central para esta tesis. En *mcm1* original se supone una única fuente (situada en (0,0,0) en la cuadrícula de simulación), no define ninguna localización para el fotosensor, además que colapsa en su salida la respuesta polar (integración angular). Este paquete generaliza las geometrías de detección; (i) facilitando simulaciones de varios canales con diferentes combinaciones de iluminantes y fuentes, (ii) elimina la restricción de la localización del iluminante, y (iii) no integra la salida. La Tabla 3 describe las clases que forman este paquete.

Tabla 3 Conjunto de clases que conforman el paquete SensingGeometry

Clase	Descripción
SensingGeometry	Esta clase encapsula una geometría de sensado particular, la cual es una colección de fuentes de luz o emisores y fotodetectores o simplemente detectores con una cierta ubicación y orientación y sus pares. Esta es la clase que actúa como un punto de entrada del paquete.
LightSource	Esta clase representa una fuente de luz ubicada en una posición determinada en el plano X, Y.
Channel	Un canal es un par entre la fuente de luz o emisor y el fotodetector.
Photodetector	Esta clase representa un fotodetector (tubos fotomultiplicadores, fotodiodos de avalancha, CCD, etc.)

- Utility:** Este paquete contiene una miscelánea de clases básicas que proveen soporte genérico rudimentario. Puede ser usado por todos los paquetes a conveniencia. Además y como se describe más detalladamente abajo, MOCARTS se equipa con un parser para el formato de entrada y salida de datos en XML que facilita la definición, lectura y validación de las entradas y salidas. En la Tabla 4 se describen las clases que integran este paquete.

Tabla 4 Conjunto de clases que conforman el paquete Utility

Clase	Descripción
Wavelength	Representa una longitud de onda expresada en [nm]
Gaussian	Esta clase define una función de distribución Gaussiana.
Point3D	Un punto tridimensional que es representado por las coordenada x, y, z.
Utils	Contiene una miscelánea de métodos para búsquedas en arreglos, o cálculos genéricos entre otros.
XMLModelReader	Esta clase se encarga de la lectura de un fichero de entrada y crear los objetos necesarios para llevar a cabo las simulaciones.
XMLModelWriter	Esta clase se encarga de la escritura a un fichero de las matrices de Reflectancia, Transmitancia, Absorción y valores totales de éstos.

El paquete de la **Interfaz Gráfica de Usuario** (GUI por sus siglas en inglés) (Tabla 5) se ha diseñado pero no implementado, sin embargo de manera informativa se presenta en esta sección para ser llevada a la sección de trabajos futuros.

Tabla 5 Conjunto de clases que conforman el paquete Interfaz Gráfica de Usuario

<b>Clase</b>	<b>Descripción</b>
Main	Clase principal para la ejecución de la aplicación.
TreeView	Es una vista de árbol de los datos a simular.
ModelView	Maneja las peticiones del controlador utilizando o notificando a las respectivas clases en el modelo.
LayerView	Ofrece la posibilidad de mostrar las capas del modelo definido
TissueView	Muestra el modelo biológico en el nivel superior de la jerarquía (agrupado por tejidos - Tissues )
OrganView	Muestra el modelo biológico en el siguiente nivel superior de la jerarquía (agrupado por Organos-Organ)
SensingGeometryView	Maneja las peticiones del controlador utilizando o notificando a las respectivas clases en el modelo. Permite observar la geometría de sentido definida.
SimulationParamsView	Maneja las peticiones del controlador utilizando o notificando a las respectivas clases en el modelo. Permite observar los parámetros de la simulación.
ResultsView	Es la forma de mostrar los resultados de la simulación.
RATView	Muestra los resultados de la Reflectancia, Absorción y Transmitancia.
GraphicResults	Es la manera de mostrar los resultados gráficos de la simulación.
Image	El resultado de la simulación se puede mostrar mediante imágenes (Cortes de la matriz de absorción, reflectancia o transmitancia)
Bar	El resultado de la simulación se puede mostrar mediante gráficas de barras.
SerieXY	El resultado de la simulación se puede mostrar mediante gráficas de dispersión XY.

### 4.3 Diagrama de Clases

La Figura 12 tiene la representación del diagrama de clases del paquete TissueModel.

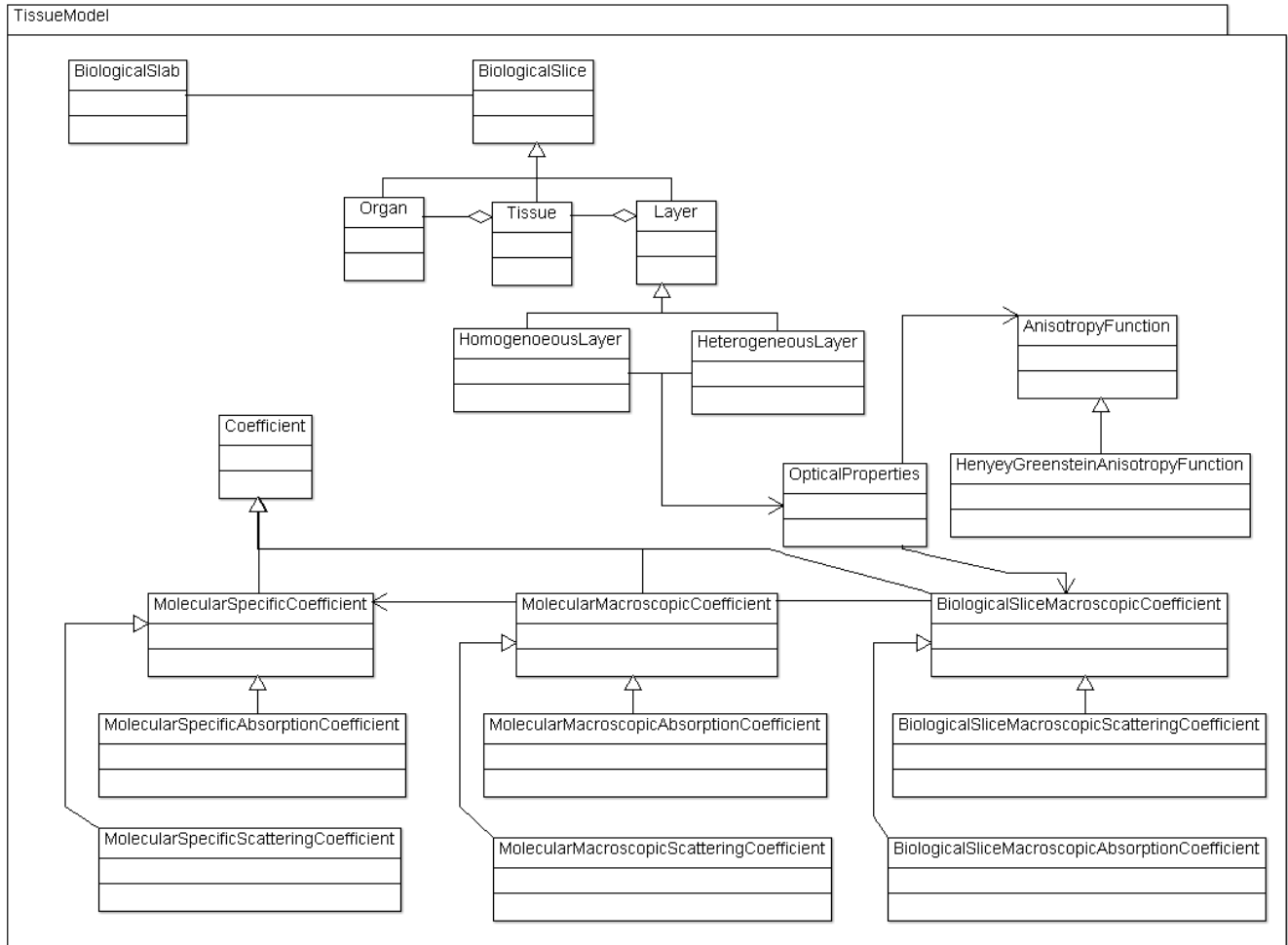


Figura 12 Diagrama de clases del paquete TissueModel.

La Figura 13 muestra el diagrama de clases para el paquete Simulation.

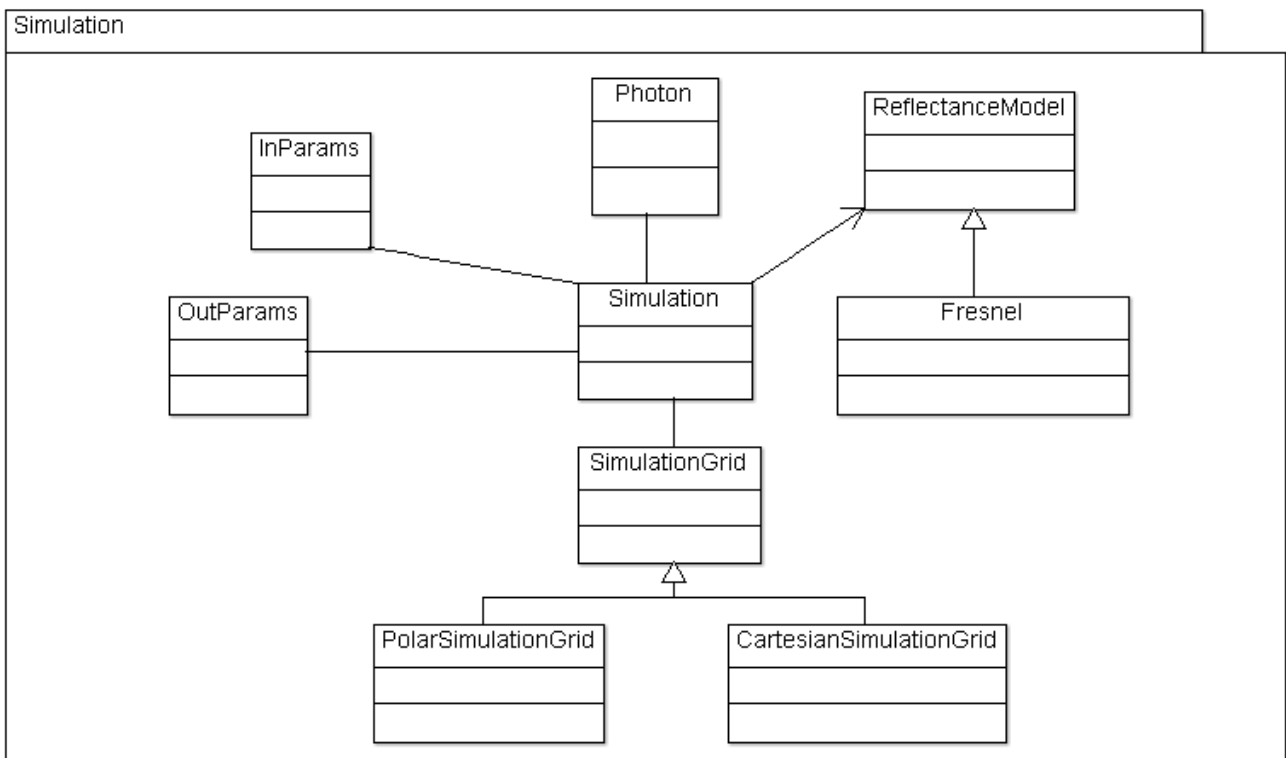


Figura 13 Diagrama de clases del paquete Simulation.

El diagrama de clases del paquete SensingGeometry se muestra en la Figura 14.

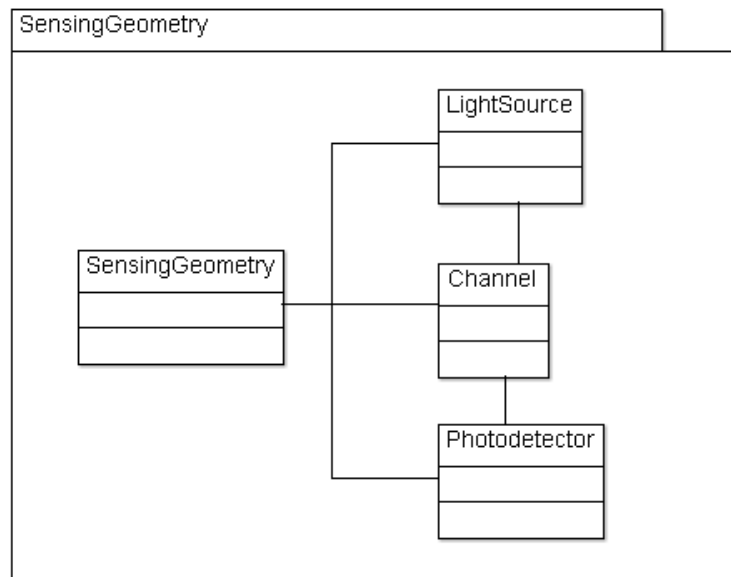


Figura 14 Diagrama de clases del paquete SensingGeometry.

#### 4.4 Diagrama de Casos de Uso

En la Figura 15 se muestra un ejemplo sencillo que modela un diagrama de Casos de Uso del sistema.

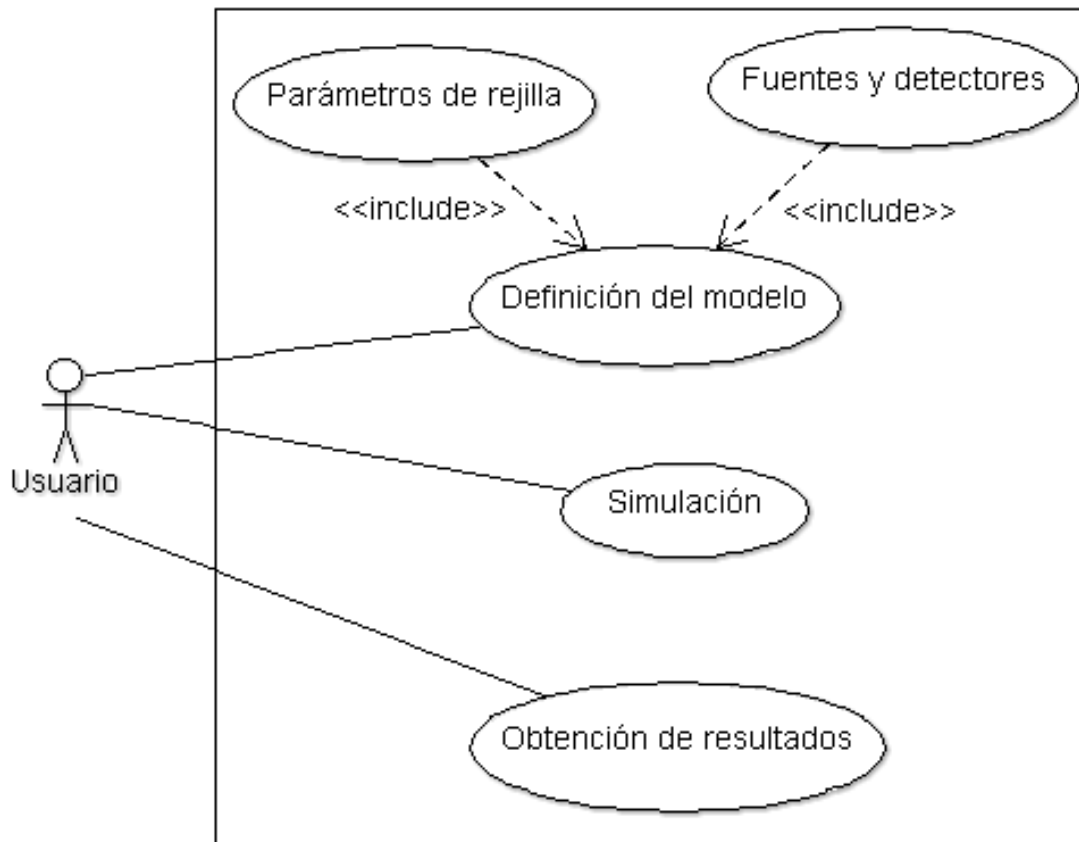


Figura 15 Diagrama de Casos de Uso de MOCARTS.

El **Actor "Usuario"** es la persona que llevará a cabo la simulación, es quien generará el archivo XML con la definición del modelo para posteriormente realizar la simulación y obtener los resultados de ésta.

El **Caso de Uso "Definición del modelo"** permite describir las características ópticas y geométricas de la rebanada de tejido biológico donde se requiere simular el transporte de la radiación. Ésta descripción se realiza en un documento XML.

El **Caso de Uso "Simulación"** es el responsable de llevar a cabo el proceso de simulación del sistema.

El **Caso de Uso "Obtención de Resultados"** permite al usuario contar con la información resultante de la simulación, esto es: matriz de absorción, matriz de reflectancia difusa, matriz de transmitancia y todo lo obtiene en el archivo XML resultado de esta simulación.

### 4.5 Diagrama de flujo

El siguiente diagrama (Figura 16) muestra el flujo del proceso de simulación en MOCARTS.

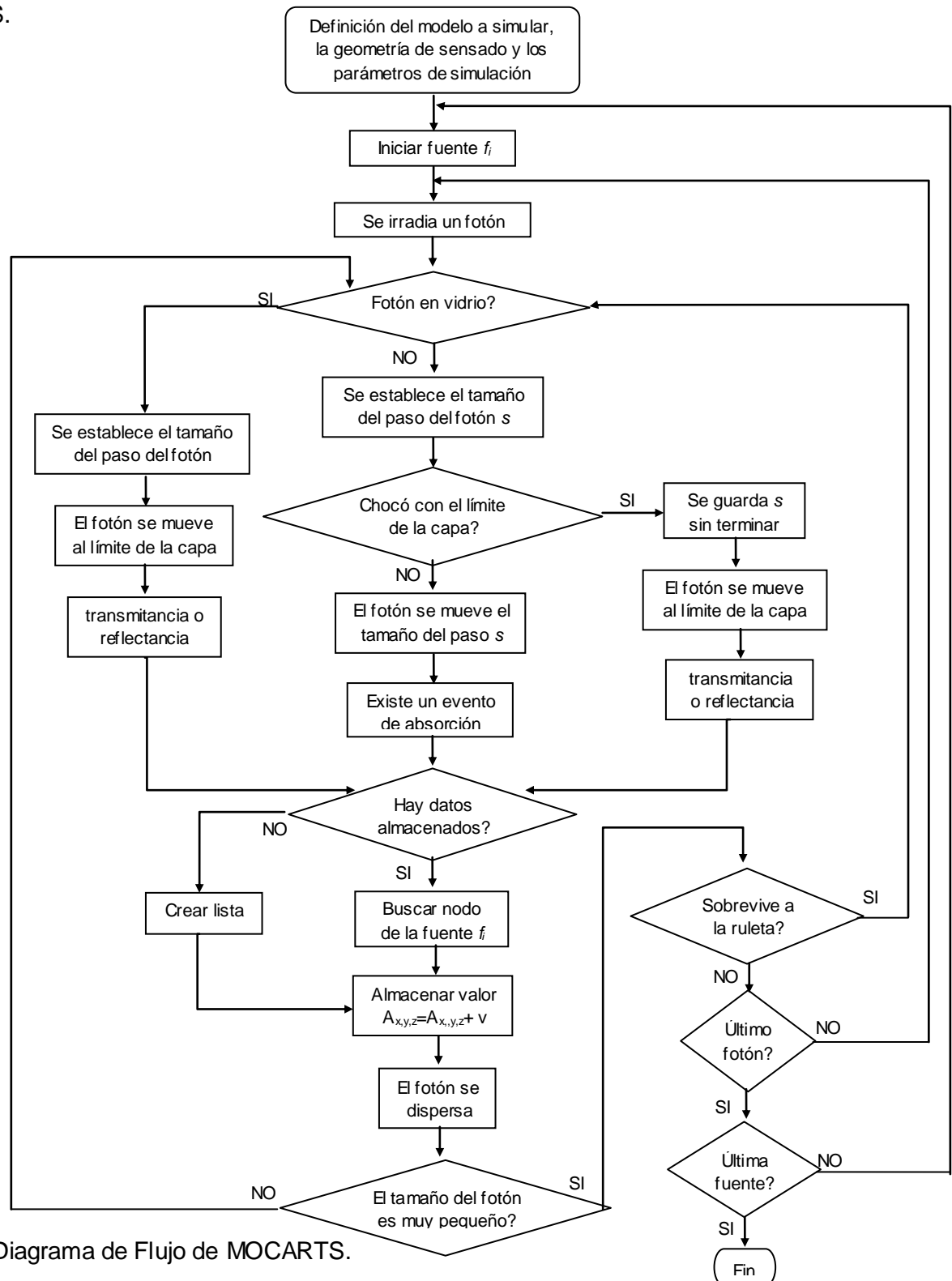


Figura 16 Diagrama de Flujo de MOCARTS.

## 4.6 Abstracción del Tejido

La disciplina que estudia todo lo relacionado con los tejidos orgánicos se llama histología. Para poder comprender como se realizó la abstracción del tejido y para poder modelarlo computacionalmente, es necesario definir algunos términos. Partiendo de la unidad más pequeña, tenemos que una **célula** (Figura 17) es la unidad anatómica fundamental de los organismos vivos [40].



Figura 17 Imagen representativa de una célula. Imagen extraída de: [41]

Los **tejidos (tissues)** son aquellas estructuras formadas por un grupo de células que pudieran ser iguales o con muy pocas diferencias entre ellas, organizadas y distribuidas de manera regular con un comportamiento fisiológico coordinado y un origen embrionario común [40]. La Figura 18 muestra una fotografía de tejido nervioso obtenida a partir de una imagen de un microscopio óptico.

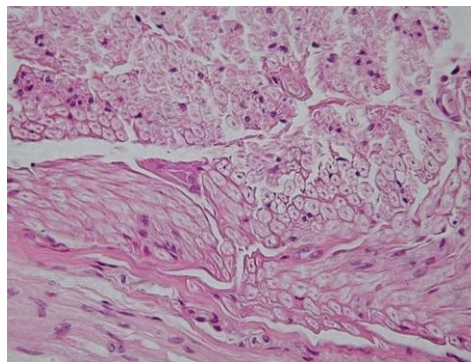


Figura 18 Tejido nervioso. Figura extraída de: [42]

Un **órgano (organ)** es una agrupación de diversos tejidos que forman una unidad estructural encargada del cumplimiento de una función determinada. Algunos órganos del cuerpo humano son: corazón, estómago, piel, pulmón y ojos por mencionar algunos.

Para realizar una simulación se toma una sección del cuerpo, la cual puede estar formada por varios órganos y tejidos de un área del cuerpo determinada, a esta agrupación la denominaremos corte biológico (**BiologicalSlice**), en el caso de la Figura 19 el recuadro en el área de la cabeza representa dicho corte biológico formado por una capa de cuero cabelludo, otra capa denominada cráneo, y por último la capa de la materia gris; entendiéndose por capa (**layer**) una zona superpuesta a otra u otras con las que se forma un todo [43] y que representa un bloque de células. En términos histológicos una capa es la agrupación más sencilla pero en términos **ópticos** puede ser **homogénea**, es decir, se considera que las células que la integran son completamente iguales, o **heterogénea**, en donde existen células diferentes, por ejemplo el cuero cabelludo puede estar densamente poblado de cabello o por el contrario ser escasa la cantidad de cabello que la persona posea y esos folículos capilares hacen que esa capa deje de ser homogénea, así como también tener ramificaciones arteriales. Sin embargo para el desarrollo de MOCARTS las capas a simular se consideran homogéneas. La agrupación de varios cortes biológicos representan una rebanada biológica (**BiologicalSlab**).

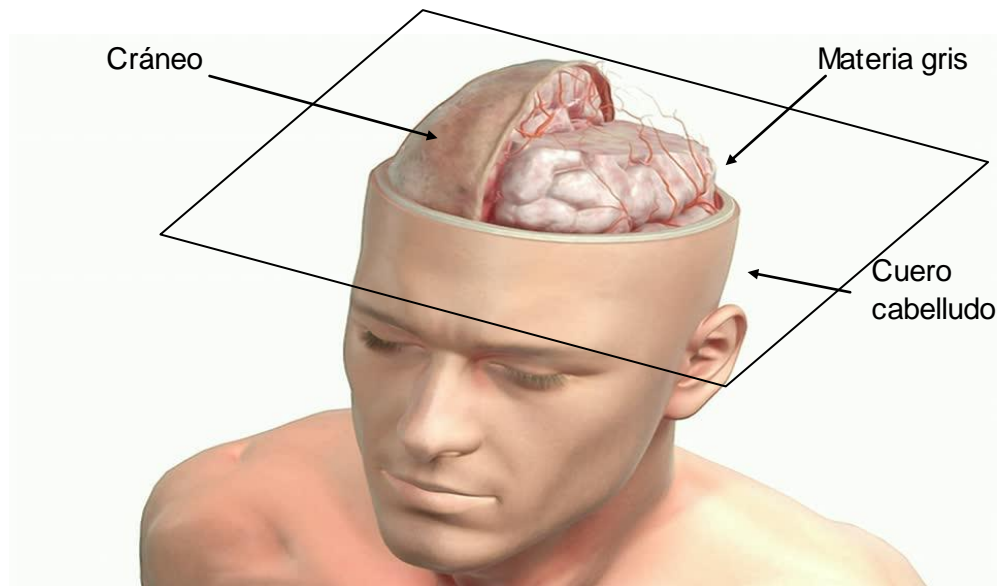


Figura 19 Ejemplo de rebanada biológica. Figura extraída y modificada de: [44]

#### 4.7 Definición en XML de una simulación

Otra de las contribuciones de esta tesis es la definición de una estructura XML para la codificación de la entrada y la salida de la simulación. El formato XML es un formato de fichero donde la información se almacena en texto plano, en principio no encriptado, que facilita la lectura por un humano y la validación de su contenido [39]. El precio a pagar es frecuentemente una representación menos eficiente en términos espaciales, con ficheros de mayor tamaño que los no estructurados.

La Figura 20 muestra la estructura de árbol implementada en un archivo XML para la definición de un modelo a simular (fichero de entrada). Este archivo está compuesto por 3 partes principales que definen la simulación:

- **El modelo de tejido biológico**, el cual se define de forma jerárquica; desde la rebanada biológica hasta la definición de las propiedades ópticas de cada capa. En esta estructura cada nivel requiere de la definición del atributo “name”.
- La definición de la **geometría de sensado**, la cual define el número de iluminantes y fotodetectores así como su localización y la configuración de canales de comunicación entre ambos.
- Los **parámetros de simulación** especifican los detalles de la configuración de la rejilla de simulación, el número de fotones, entre otros.

Un ejemplo de este archivo de entrada se puede encontrar en el Anexo 1 Archivo de definición de modelo de simulación.

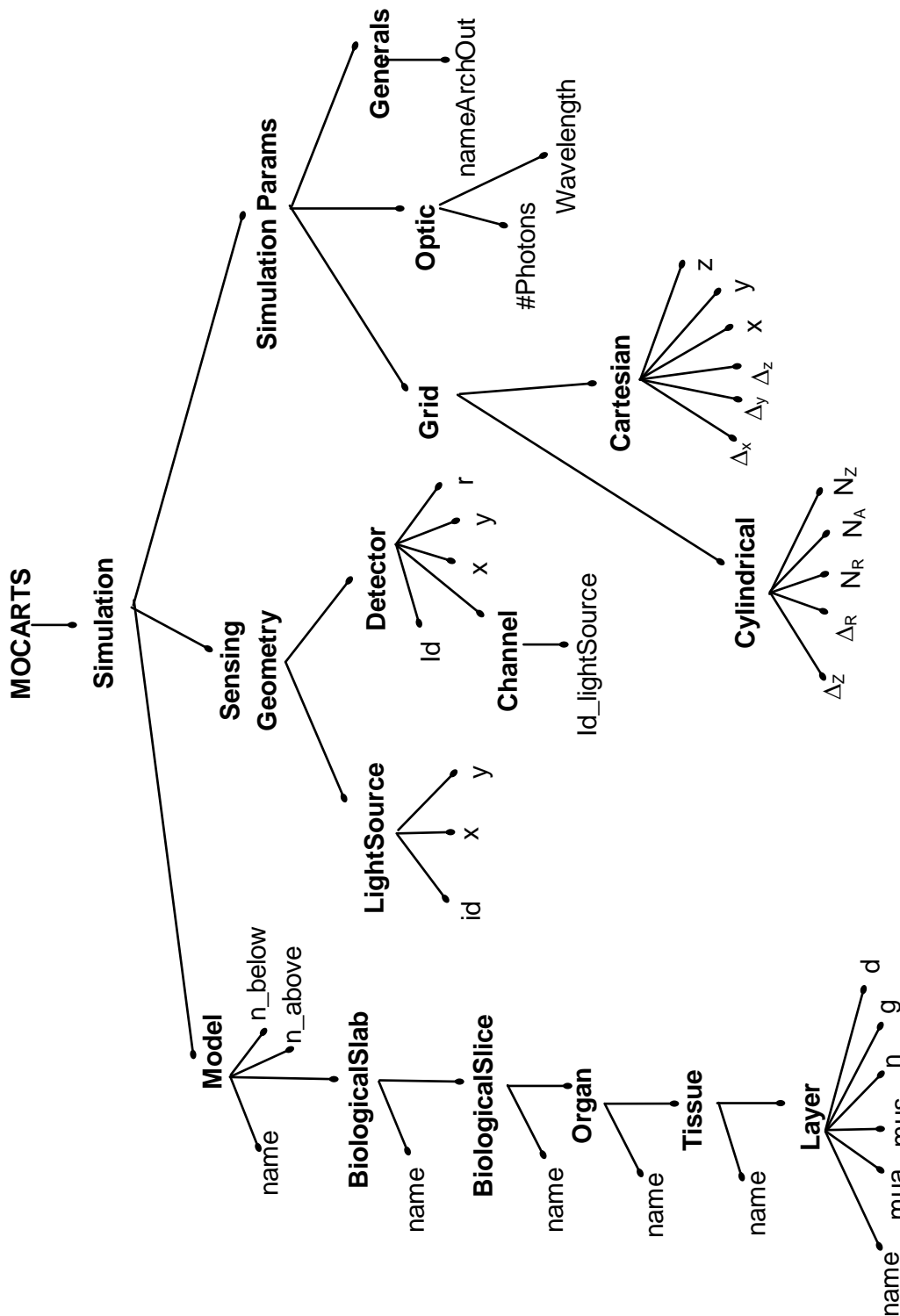


Figura 20 Definición del DTD (Document Type Definition) de XML para la definición de un modelo de simulación.

El fichero de salida con los valores resultantes del cómputo de la información está diseñado modelando el siguiente árbol XML que se ilustra en la Figura 21.

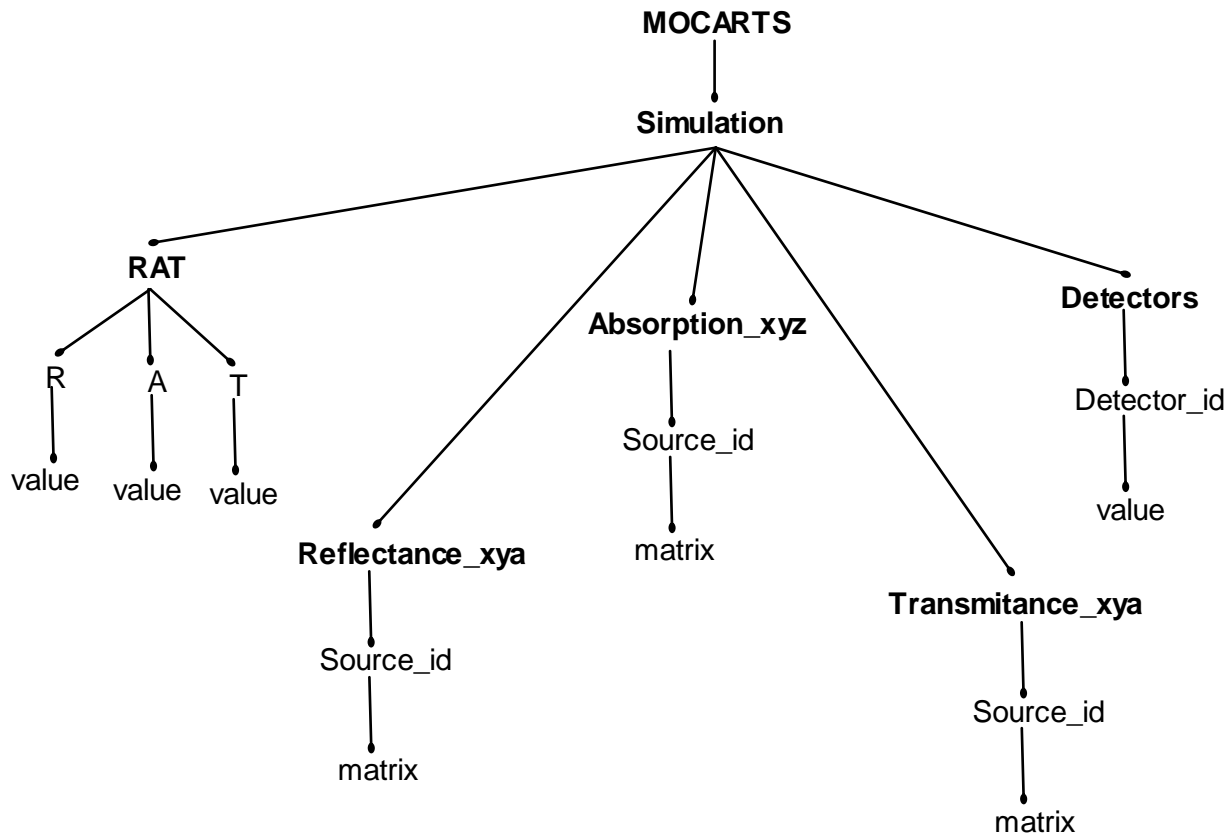


Figura 21 Definición del DTD (Document Type Definition) de XML para el fichero de salida del modelo de simulación.

El Anexo 2 Fichero de salida del modelo simulado, muestra un ejemplo del formato del archivo de salida.

## 4.8 Rejillas

Tanto en *mcmI* como en MOCARTS, la simulación de propagación de fotones se realiza de manera natural en un espacio “continuo” tridimensional cartesiano (X,Y,Z), así, los eventos de absorción y esparcimiento ocurren en cualquier punto en este espacio. Sin embargo, para fines prácticos, dadas las limitaciones computacionales de representación y almacenamiento, la energía del fotón absorbida por el tejido es acumulada en las celdas de una rejilla definida previamente en los parámetros del modelo de simulación. Por lo cual se obtiene una discretización del espacio.

En *mcmI* se usa una rejilla cilíndrica (Figura 22 (a)) dónde la absorción es acumulada en un plano (r,z) del cilindro bajo la suposición de isotropía perpendicular a Z. Esto es, cada evento de absorción en la coordenada (x,y,z) durante la simulación es colapsado sobre el plano (r,z) donde la absorción se va acumulando en cada celda de dicha matriz. En MOCARTS se ha utilizado una rejilla rectangular (Figura 22 (b)) para mantener en cada posición (X,Y,Z) la absorción correspondiente sin la suposición de isotropía perpendicular a Z. Esto ofrece una ventaja ya que permite obtener los resultados de absorción (o reflectancia/transmitancia) más cercanos a la realidad. Por otro lado, la desventaja del uso de la rejilla rectangular repercute en un incremento de espacio de almacenamiento.

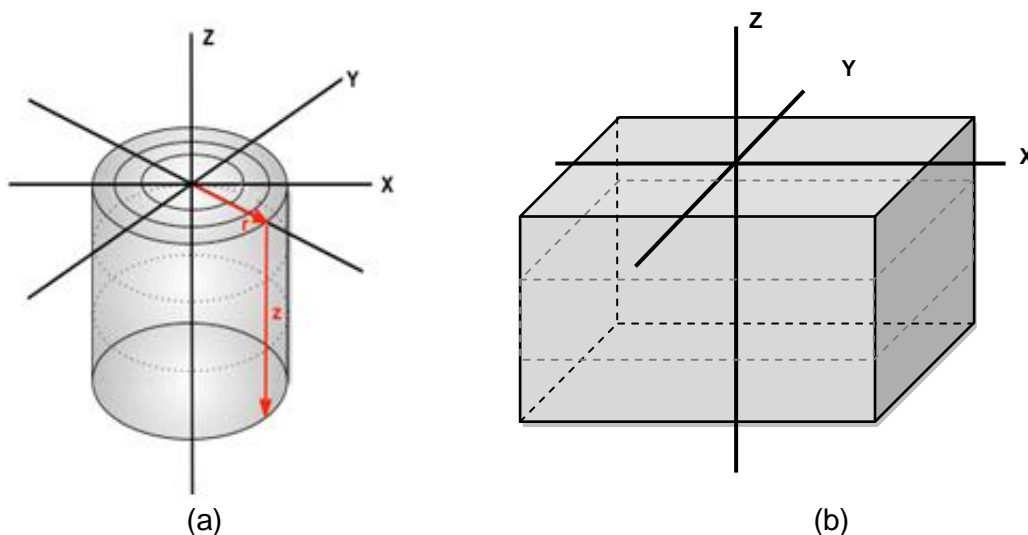


Figura 22 Rejillas de simulación a) *mcmI*, b) MOCARTS.

Al igual que en *mcmI*, la definición del tamaño y resolución de la rejilla se deja al usuario para adecuarla a sus requerimientos. Se debe tener en cuenta el compromiso entre resolución y espacio.

## 4.9 Geometrías de sensado multicanal

La implementación de múltiples fuentes de irradiación y múltiples detectores es una característica principal en MOCARTS. Ésta, amplía la funcionalidad y flexibilidad ofrecida por

*mcmI*. Para el desarrollo de esta característica no se consideró el efecto de interacción entre fotones que pudieran alterar su ruta de propagación. Por lo tanto, la propagación de los fotones de cada fuente de iluminación se considera independiente. Sin embargo, la energía depositada en el tejido (absorción) o la luz que sale del tejido (reflectancia/transmitancia) es acumulada tanto por cada fuente de iluminación como en su totalidad.

Bajo este supuesto, las simulaciones se realizan de forma secuencial, es decir, cada fuente simula los  $N$  fotones definidos en el modelo y tras terminar con el último fotón se inicia una nueva simulación con la siguiente fuente de iluminación. Este proceso continúa para el número de iluminantes definidos.

La estructura de datos utilizada para almacenar la información correspondiente de la simulación consistió en una matriz tridimensional en donde cada celda contiene una lista ligada (Figura 23). Cada nodo de la lista almacena la absorción/reflectancia/transmitancia correspondiente a una determinada fuente de iluminación. Esta estructura fue propuesta con la intención de optimizar el uso de memoria, ya que los nodos de las listas son creados dinámicamente sólo en aquellas celdas donde ha ocurrido algún evento de absorción/reflectancia/transmitancia. El siguiente pseudocódigo explica el algoritmo para la utilización de esta estructura en un evento de interacción.

Algoritmo 1. Asignación de valores en la estructura de almacenamiento

---

**Entrada:** Posición  $p$  donde ocurre un evento de interacción  
Identificador  $F_i$  de la fuente de iluminación  
Valor  $V$  a almacenar (resultado de la interacción)  
Matriz de almacenamiento  $M$   
**Salida:** Estructura de datos con el dato almacenado en la posición correspondiente

Si  $M[p]$  es nulo (la celda está vacía, es el primer evento que ocurre en esta posición)

    Crear nodo inicial  $N$  de la lista ligada

    Almacenar los datos correspondientes  $M[p].N \leftarrow (F_i, V)$

Sino (Ya hay datos almacenados en la posición  $p$ )

    Para cada nodo  $N$  en la lista (buscar nodo)

        Si  $N.Fid = F_i$

$N.valor \leftarrow N.valor + V$

    Si no se encontró ningún nodo en la lista con identificador  $F_i$

        Agregar nuevo nodo  $N$  al final de la lista

        Almacenar los datos correspondientes  $M[p].N \leftarrow (F_i, V)$

---

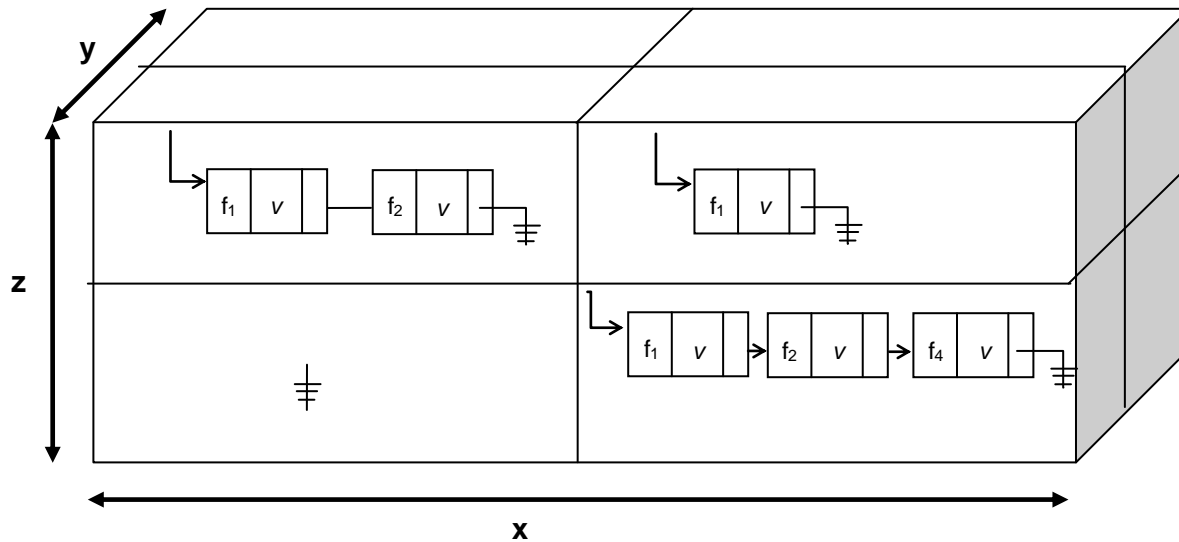


Figura 23 Matriz de listas ligadas para almacenar el valor de la energía ( $v$ ) proveniente de determinada fuente de luz ( $f_i$ ).

Esta estructura y el algoritmo 1 se aplican para el almacenamiento de la absorción, reflectancia y transmitancia. Sin embargo, existe una diferencia entre las matrices de reflectancia y transmitancia con respecto a la de absorción. Estas matrices que también son tridimensionales almacenan información referente al ángulo de salida de los fotones en la dimensión correspondiente al eje Z de la matriz de absorción. La Tabla 6 resume la estructura y objetivos de estas matrices.

Matriz	Dimensión	Objetivo
Absorción	3D-(x,y,z)	Almacenar la absorción del tejido en cada posición (x,y,z)
Reflectancia	3D-(x,y,a)	Almacenar la reflectancia en la superficie (x,y,0) y en el ángulo de salida del fotón (a)
Transmitancia	3D-(x,y,a)	Almacenar la transmitancia en la base del tejido (x,y,max(z)) y en el ángulo de salida del fotón (a)

Tabla 6 Descripción de las Matrices de salida

## 5. Experimentos y resultados

En este capítulo se describen las pruebas realizadas y los resultados de éstas para llevar a cabo la verificación de MOCARTS contra *mcm1*, el cual se ha tomado como “gold standard” en este trabajo, así como la validación de la hipótesis planteada.

El desarrollo y pruebas del sistema se llevó a cabo en una computadora portátil HP ProBook 4520s con procesador Intel Core i3 CPU, M 330, 2.13GHz, Memoria RAM de 2.0 GB y Sistema Operativo de 32 bits.

### 5.1 fNIRS

La espectroscopía funcional del infrarrojo cercano (functional Near InfraRed Spectroscopy, fNIRS, por sus siglas en inglés) es una modalidad no invasiva de neuroimagen que permite el monitoreo de la actividad cerebral. Este monitoreo se puede conseguir de forma directa, observando la llamada señal óptica rápida que surge de los cambios en la dispersión de la luz en su viaje por el tejido, o bien de forma indirecta mediante la observación de los cambios hemodinámicos asociados a la actividad cerebral mediante el llamado acoplamiento neurovascular.

Esta técnica está basada en la espectroscopía en el rango de la radiación infrarroja, ya que en este rango de longitudes de onda, se encuentra la “ventana óptica” descrita anteriormente con longitudes de onda entre 600 y 1000nm. Su funcionamiento es engañosamente sencillo; se irradia luz infrarroja sobre el cuero cabelludo, esta luz viaja a través del cráneo, y las meninges y alcanza la corteza cerebral. La parte de la luz retrodispersada o transiluminada que abandonan el tejido conlleva información sobre la estructura o función del mismo, como se observa en la Figura 24.

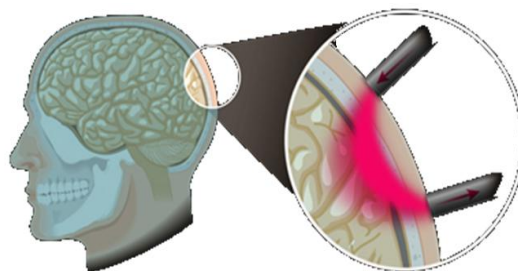


Figura 24 Principios de la interacción de la luz con la material. Figura extraída de [45].

La neuroimagen óptica es un ejemplo práctico donde se obtiene información del tejido de interés (cerebral) a partir de mediciones de radiación retrodispersada en superficie; en este caso el cuero cabelludo, y donde las geometrías de sensado complejas son comunes. Sin otro motivo en particular (se podría haber utilizado otros tejidos), en esta tesis se verificará la herramienta y validará la hipótesis partiendo de un modelo de la cabeza humana adulta.

## 5.2 Verificación de MOCARTS

Para comprobar el correcto funcionamiento del sistema MOCARTS se han hecho varias simulaciones de un modelo simplificado de la cabeza humana. La verificación consiste en la comparación de la salida de MOCARTS con respecto a la de *mcmI* y la medición del error obtenido. Las características ópticas de los tejidos simulados se han obtenido del capítulo 2: Optical Properties of Tissue del libro Biomedical Photonics Handbook [46] para generar un modelo y con el cual se estarán haciendo las simulaciones para llevar a cabo la verificación y validación del sistema.

La primera simulación consistió en observar la trayectoria de un sólo fotón, para lo cual se modeló una rebanada de tejido biológico formado por 4 capas. Esta secuencia de capas representa la cabeza humana adulta. Para la representación de la información y posterior cómputo se definen los parámetros de la rejilla de simulación como se muestra en la Tabla 7 y las propiedades ópticas de cada capa para una longitud de onda de 800nm se indican en la Tabla 8.

Parámetro	Valor
Separación entre las líneas en r (radio)	0.01 cm
Separación entre las líneas en z (profundidad)	0.01 cm
Número de elementos en r	500
Número de elementos en z	240
Separación de salida angular de fotones	90
Tamaño de la rejilla (r x z)	5 x 2.4 cm
Número de fotones	100,000

Tabla 7 Parámetros de la rejilla

Capa	Índice de Refracción [-]	Coefficiente de Absorción Macroscópico [cm <sup>-1</sup> ]	Coefficiente de Esparcimiento Macroscópico [cm <sup>-1</sup> ]	Factor de Anisotropía	Ancho de la capa [cm]
Medio superior	1	-	-	-	-
Cuero cabelludo	1.42	0.127	190.376	0.900	0.3
Cráneo	1.555	0.147	161.245	0.900	0.5
Materia gris	1.360	0.270	75.157	0.899	0.4
Materia blanca	1.380	0.931	372.501	0.870	0.2
Medio inferior	1	-	-	-	-

Tabla 8 Propiedades ópticas del modelo de tejido simulado.

De manera gráfica se puede ver en la Figura 25 las 4 capas simuladas: el cuero cabelludo, el cráneo, la primera capa del cerebro (materia gris) y la materia blanca. El modelo anterior, asume una geometría de tejido plana y homogénea intracapa. Aunque son suposiciones alejadas de la realidad, ambas son comúnmente utilizadas en la literatura de neuroimagen [2].

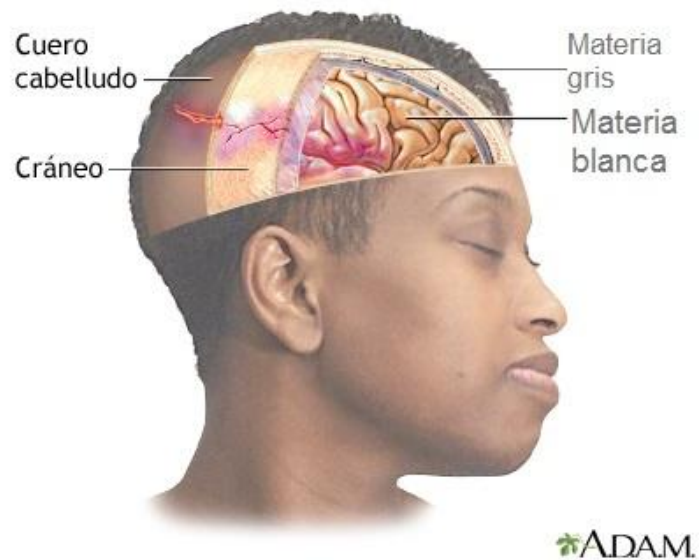


Figura 25 Las 4 capas del tejido simulado. Aunque de forma didáctica, la imagen idealiza el modelo. La geometría del tejido simulado es plana. Imagen extraída y modificada de [47].

Una sola fuente de luz fue simulada irradiando un fotón desde el origen (0,0,0) en aras de facilitar la comparación con *mcmI*. El mismo modelo fue simulado con *mcmI* y MOCARTS y a ambos sistemas se les fijó un mismo valor para la semilla generadora (*seed=-1*) de números aleatorios.

Para la visualización gráfica de las trayectorias de estos fotones se hizo uso de la herramienta MATLAB para generar las figuras, ya que MOCARTS, al igual que *mcmI*, aún carece de capacidades gráficas. La Figura 26 muestra la trayectoria de un fotón con el sistema *mcmI* y MOCARTS.

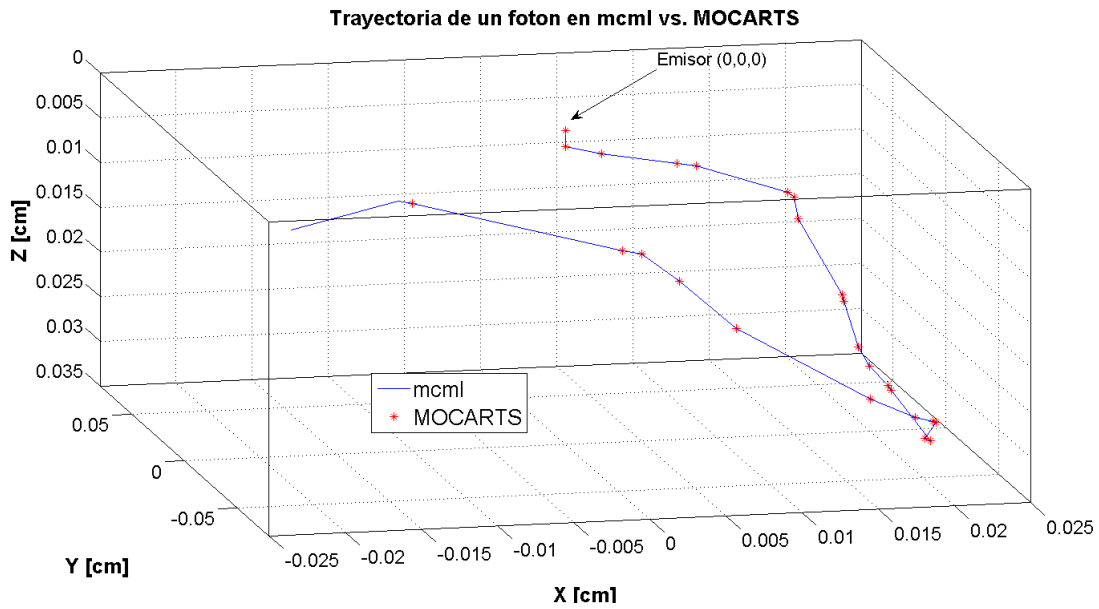


Figura 26 Trayectoria de un fotón con *mcml* (azul) y la nueva herramienta de software (rojo). La secuencia acaba con la absorción del paquete fotónico.

El error cuadrático medio (medido con la ecuación (5)) de la trayectoria de estos fotones se muestra en la Tabla 9.

	Raíz del Error Cuadrático Medio (RMSE)
Trayectoria de 1 fotón	4.0335e-009

Tabla 9 Error cuadrático medio de la Trayectoria de 1 fotón

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (\hat{Y}_i - Y_i)^2} \tag{6}$$

Se observó que el error encontrado entre las trayectorias de los fotones es debido a la precisión en el cálculo de la función logaritmo ente los compiladores C y java. Estos muestran una diferencia del orden de  $10^{-6}$ .

Una vez verificada la trayectoria del fotón, se procedió a una simulación más elaborada. En esta segunda simulación, se lanzaron 100,000 fotones sobre el modelo de 4 capas de la cabeza humana adulta anteriormente descrito. De nuevo, se simuló una única fuente de irradiación localizada en (0,0,0). Pero en esta ocasión, se observaron las matrices de absorción, reflectancia y transmitancia en ambos sistemas.

La Figura 27 muestra la matriz de absorción en el tejido para la simulación del modelo definido. Se indican las 4 capas que forman al modelo y el origen donde se irradia la fuente de luz. Se puede observar que después de 1.4cm de profundidad del tejido la gráfica ya no muestra diferentes intensidades de color, esto se debe a que nuestro modelo simulado es de un grosor total de 1.4cm, pero la rejilla de simulación en este ejemplo es más profunda.

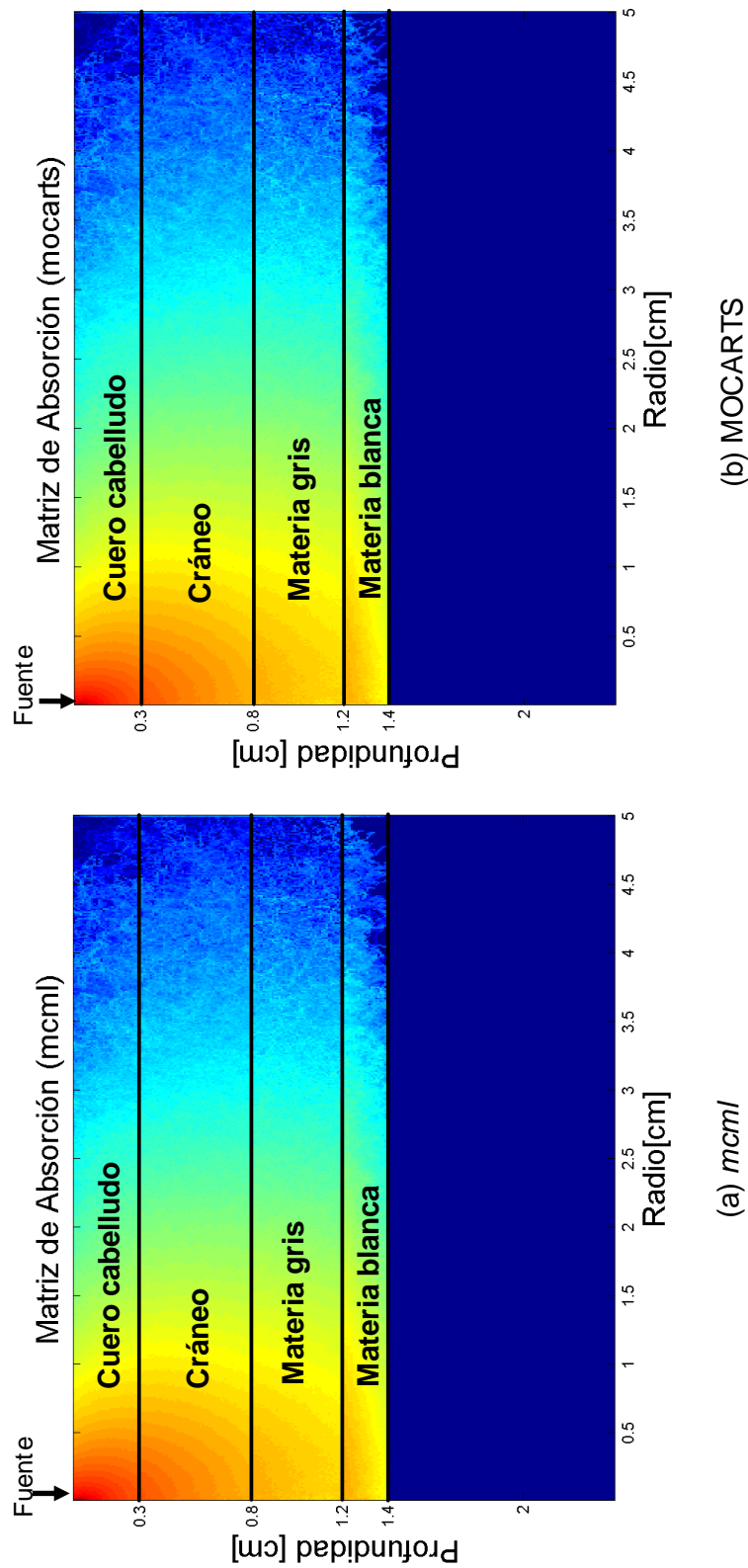
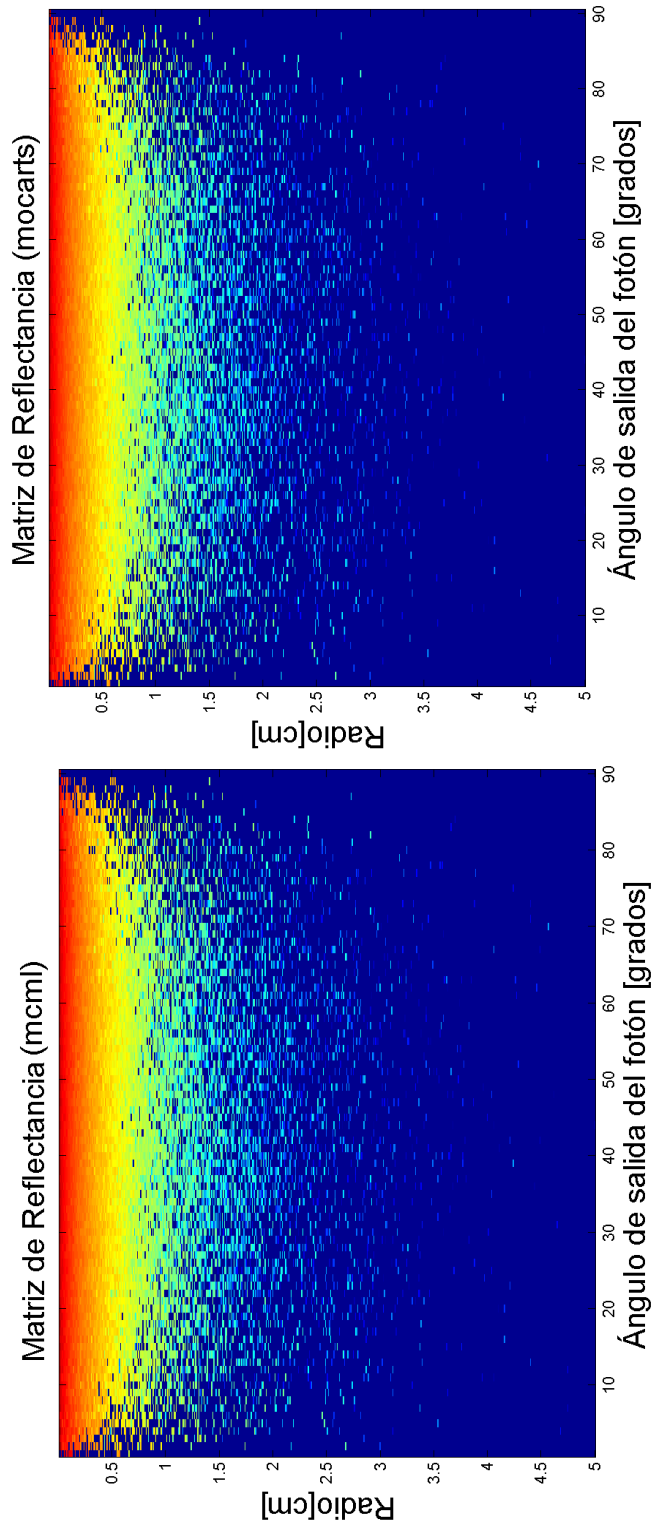


Figura 27 Matriz de absorción en ambos simuladores.

La Figura 28 muestra la matriz de reflectancia generada con el modelo e indica la cantidad de fotones que han logrado ser remitidos en diferentes direcciones (de 0° a 90°), la discretización de este intervalo se determina en los parámetros de entrada de la simulación. En este caso en incrementos de un grado.

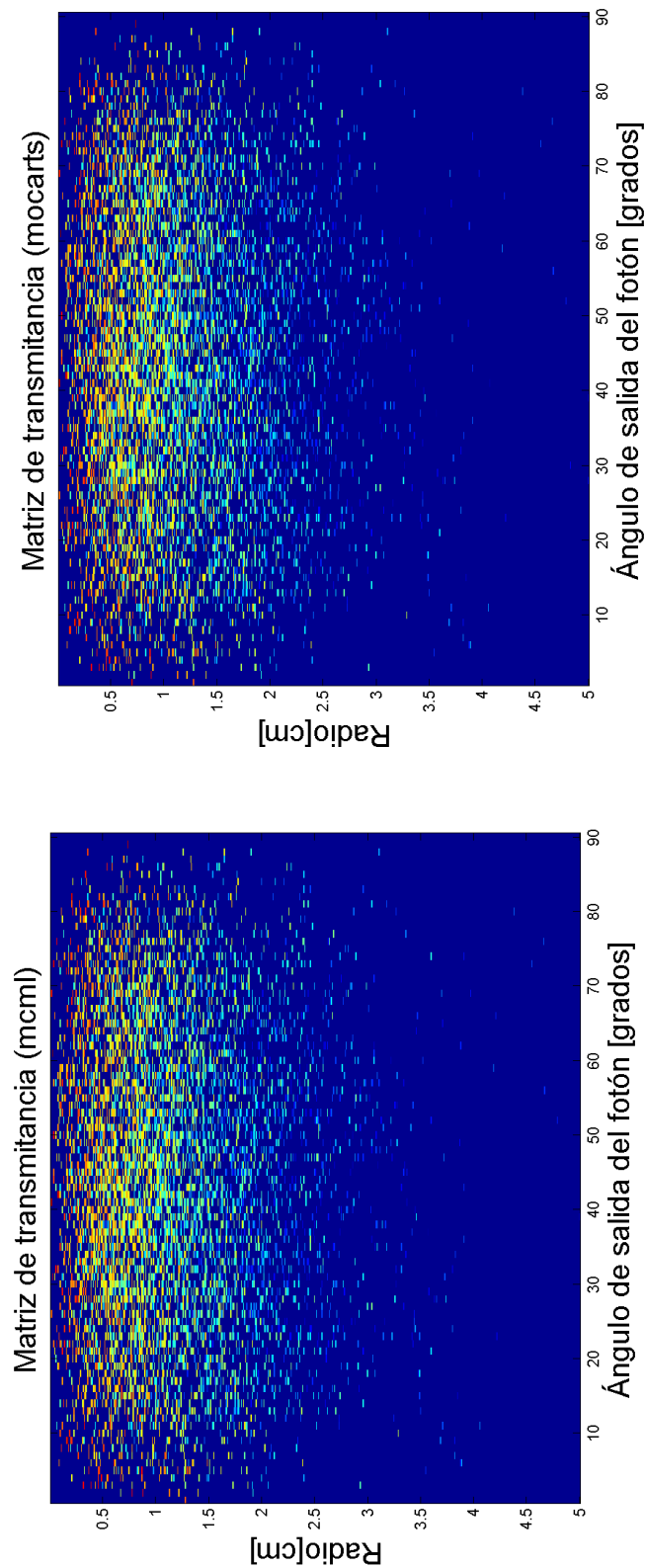


(a) *mcmI*

(b) MOCARTS

Figura 28 Matriz de reflectancia difusa en ambos sistemas.

En la Figura 29 se puede apreciar la matriz de transmitancia generada en el modelo, es decir, la fracción de radiación que traspasó las 4 capas del modelo definido en diferentes direcciones (de  $0^\circ$  a  $90^\circ$ ), al igual que la reflectancia, la discretización de este intervalo se determina en los parámetros de entrada de la simulación. Para este caso en incrementos de un grado.



(a) *mcmI*

(b) MOCARTS

Figura 29 Matriz de transmitancia en ambos sistemas.

Una vez computados los datos y realizado 3 simulaciones con el mismo modelo pero variando el número de fotones irradiados, se calculó el error cuadrático medio de los datos de salida de *mcml* y MOCARTS, donde se puede apreciar que a menor cantidad de fotones simulados el error de los datos es mayor, a diferencia que si la cantidad de fotones es mayor, el error tiende a ser menor, como era de esperar en una técnica probabilística como MC. Lo anterior se ilustra en la Tabla 10.

Datos de salida	Raíz del Error Cuadrático Medio (RMSE)		
	10,000 fotones	100,000 fotones	1,000,000 fotones
Matriz de absorción	0.0136	0.0026	0.0011
Matriz de reflectancia	0.2731	0.0728	0.0293
Matriz de transmitancia	4.7207e-004	1.1360e-004	7.3175e-005

Tabla 10 Raíz del error cuadrático medio entre los datos de salida de *mcml* y MOCARTS

### 5.3 Validación

Una vez verificada la herramienta con respecto al estándar, planteamos un experimento sencillo para la validación de la hipótesis. En este experimento clásico del tipo control vs intervención, nuestro único factor es la herramienta de simulación con dos tratamientos; una herramienta que admite geometrías de sensado multicanal (MOCARTS –intervención-) contra otra que no las admite (-control-). La unidad experimental es el tejido simulado. Se controlan el escenario de simulación (propiedades ópticas del tejido así como el número y posiciones de las fuentes de irradiación). Como el control, sólo admite la simulación de una fuente de irradiación, el número de estas debe ser necesariamente controlado, aunque en el Capítulo 6 Simulador MOCARTS se muestran simulaciones con varias fuentes. Si la hipótesis es correcta, no debiera haber diferencias significativas estadísticamente entrabos tratamientos.

Nótese la aparente similaridad de la prueba con respecto a la verificación; la diferencia es conceptual más que práctica en este caso particular. En la verificación se busca que el error con respecto al gold estándar sea mínimo; aquí se busca que no haya diferencia significativa entre los resultados. En la verificación, sólo se lanza 1 simulación (no hay replicación), mientras que en el experimento hay un proceso de varias simulaciones con diferentes semillas aleatorias –las mismas para ambos tratamientos- y diferente número de fotones (replicación). En la verificación, no hay factores; la respuesta de *mcml* se asume como la verdad, mientras que en la validación la verdad (desconocida) es la distribución espacial de la radiación en el tejido, y lo que se pretende es que ambas herramientas, dado que asumen el mismo modelo físico, lleguen a la misma conclusión (sin diferencia significativa) llamada validación concurrente.

Para las pruebas descritas en este capítulo se utilizó el mismo modelo descrito en la sección anterior simplemente que para este experimento no se asignaron semillas para la generación de números aleatorios.

### 5.3.1 Medición del error cuadrático medio entre *mcm* y MOCARTS

En la Figura 30 se ha realizado la sumatoria de las matrices de absorción para cada simulación efectuada con diferente cantidad de fotones ( $10$ ,  $10^2$ ,  $10^3$ ,  $10^4$ ,  $10^5$ ,  $10^6$ ) en ambos simuladores, obteniendo así que a mayor cantidad de fotones la tendencia global converge al mismo valor. En estas simulaciones ya no se estableció ninguna semilla fija para la generación de números aleatorios.

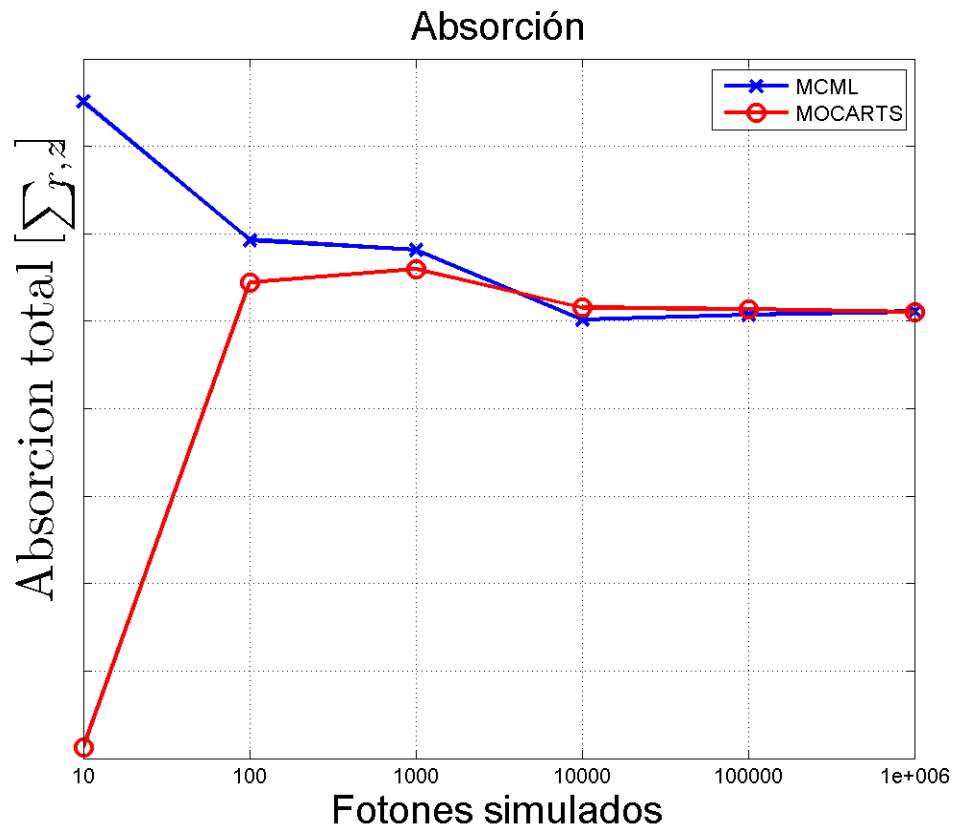


Figura 30 Comparación de Absorción total entre *mcm* y MOCARTS.

En la Figura 31 se ha realizado la sumatoria de las matrices de reflectancia para cada simulación efectuada con diferente cantidad de fotones ( $10$ ,  $10^2$ ,  $10^3$ ,  $10^4$ ,  $10^5$ ,  $10^6$ ), obteniéndose al igual que en la gráfica anterior una convergencia al mismo valor a mayor cantidad de fotones. Nuevamente en estas simulaciones ya no se estableció ninguna semilla para la generación de números aleatorios.

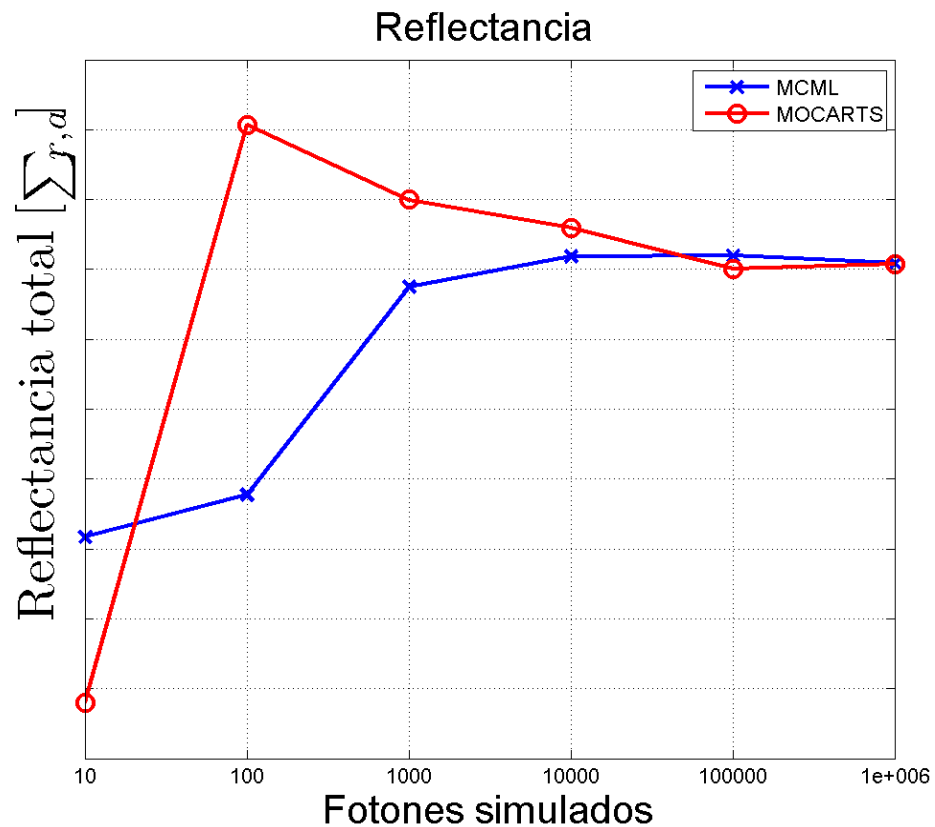


Figura 31 Comparación de Reflectancia total entre *mcmI* y MOCARTS.

En la Figura 32 se ha realizado la sumatoria de las matrices de transmitancia en simulaciones con diferente cantidad de fotones ( $10$ ,  $10^2$ ,  $10^3$ ,  $10^4$ ,  $10^5$ ,  $10^6$ ), y logrando una tendencia global hacia un mismo valor. De igual manera en estas simulaciones ya no se estableció ninguna semilla para la generación de números aleatorios.

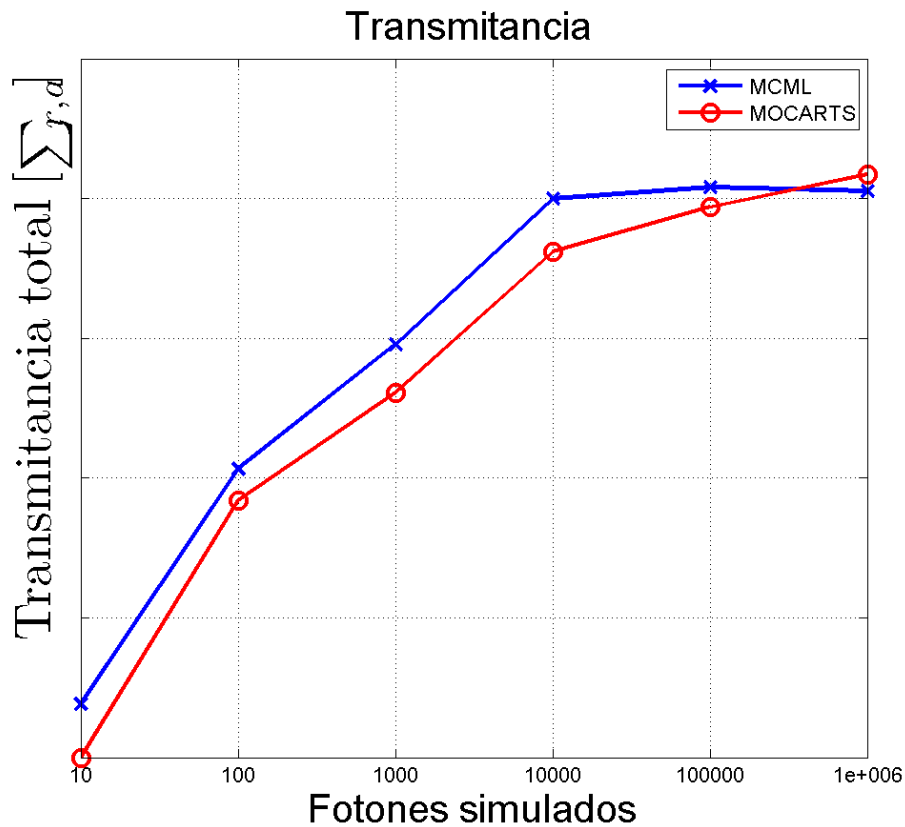


Figura 32 Comparación de Transmitancia total entre *mcml* y MOCARTS.

En la Figura 33 se grafican los errores reportados en las 3 gráficas anteriores que comparan a *mcmI* con MOCARTS unificándolos en un solo reporte indicando el error cuadrático medio de la reflectancia difusa, absorción y transmitancia y dejando observar que a mayor cantidad de fotones simulados, el error tiende a ser cercano a cero.

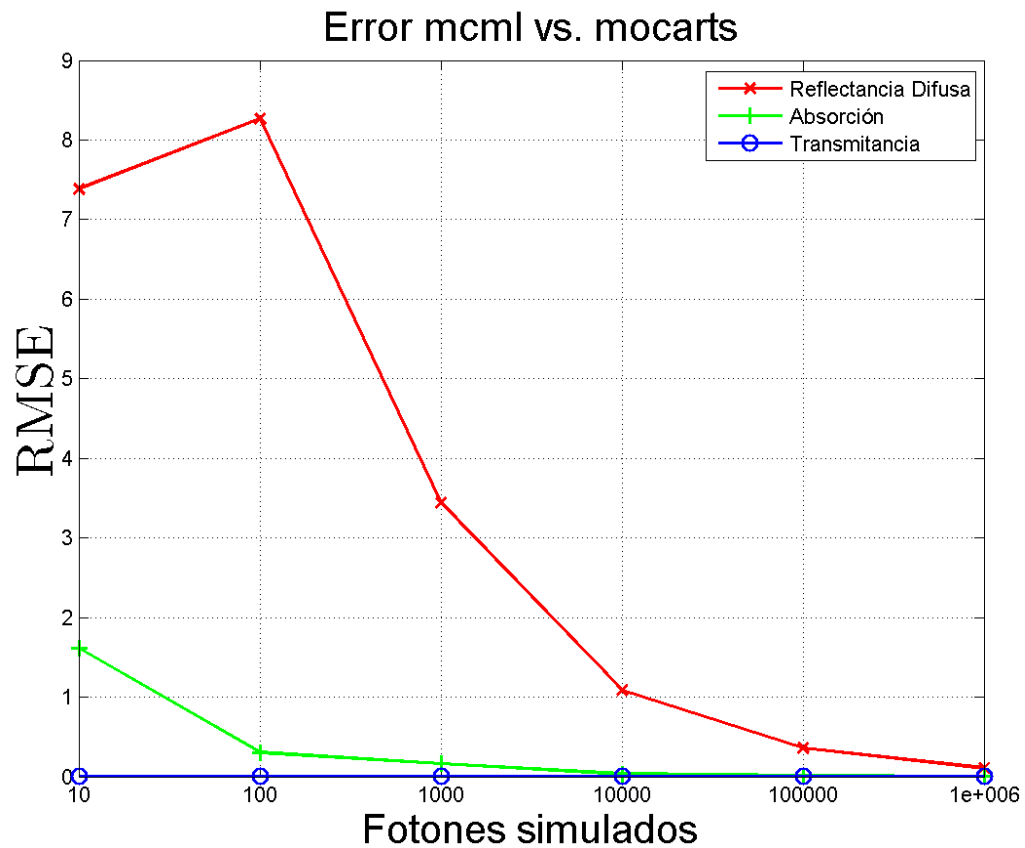


Figura 33 Error cuadrático medio entre las simulaciones de *mcmI* y MOCARTS.

## 6. Simulador MOCARTS

En este capítulo se pretende mostrar las bondades que nos permite MOCARTS, entre ellas y la principal aportación de este sistema, la inclusión de múltiples fuentes de iluminación.

Para la realización de las gráficas siguientes se hizo uso del software MATLAB, el cual nos permitió manejar la información resultante de MOCARTS para usarlos por ejemplo en la Figura 34 donde se muestra la representación 3D del modelo de tejido simulado con 24 capas en el eje Z y con cortes en los ejes ZX y ZY, los cuales se intersectan en el origen.

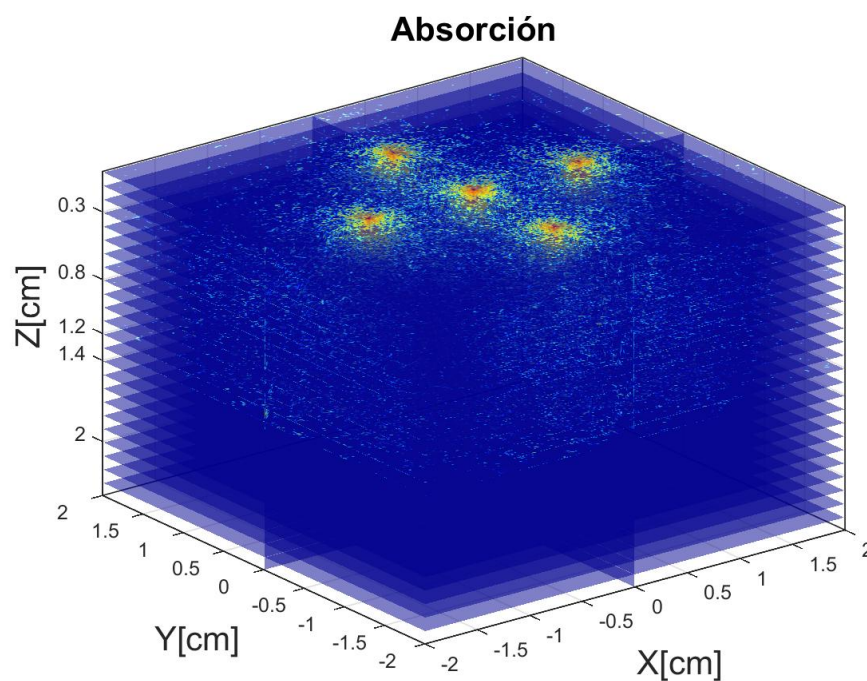


Figura 34 Representación 3D del modelo de tejido simulado.

En la Figura 35 se muestra el plano XY del tejido con cortes en ZX y ZY que se intersectan en el origen, además incluye la posición de las fuentes de iluminación (puntos en color azul) y los detectores (puntos en color rojo).

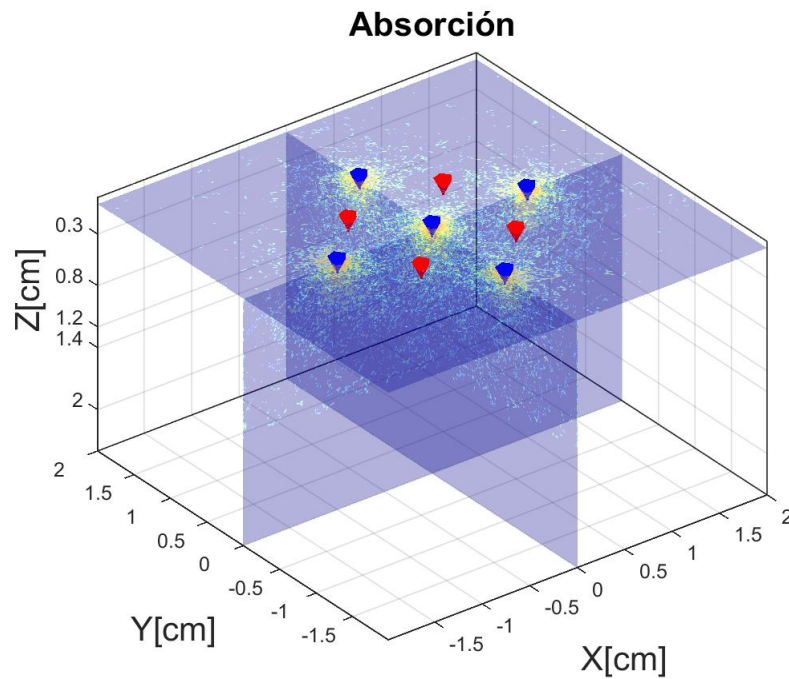


Figura 35 Superficie XY del tejido simulado con cortes en los ejes ZX y ZY, los cuales se intersectan en el origen.

En la Figura 36 se muestran cortes en ZX y ZY intersectándose en el origen y dejando observar la absorción que existe en el tejido debido a las 5 fuentes de iluminación definidas en la simulación.

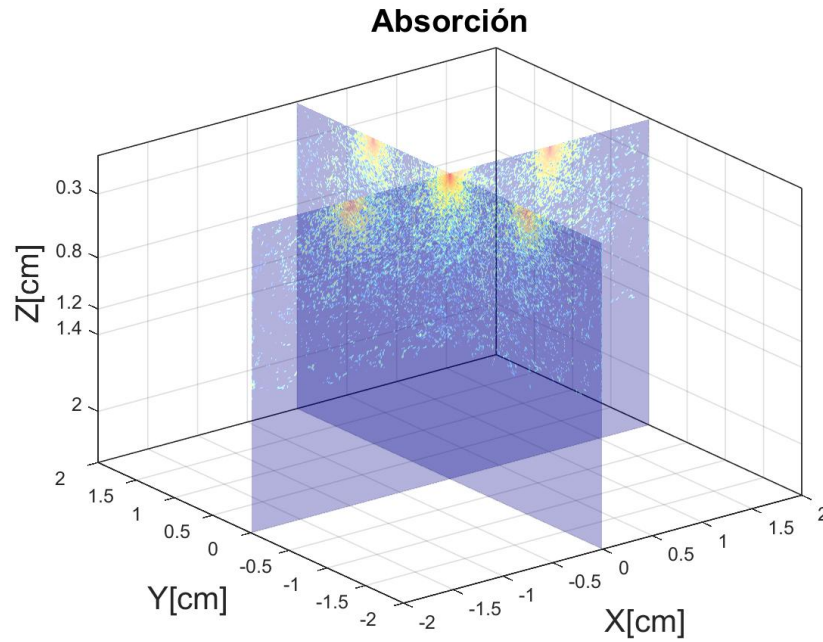


Figura 36 Corte del volumen de tejido simulado donde se muestran 5 fuentes.



## 7. Conclusiones y trabajos futuros

### 7.1 Conclusiones

En esta tesis se presentó la investigación de transporte de la radiación electromagnética en geometrías de sensado multicanal. Para lo cual fue necesario el diseño, desarrollo e implementación de un sistema de transporte de la radiación basado en la técnica de Monte Carlo – MOCARTS - que ha teniendo como base el simulador mcml del cual se han extendido sus capacidades en los siguientes puntos:

➤ Geometrías de sensado multicanal

El simulador MOCARTS permite la definición de múltiples canales de sensado. Esto se logró mediante el diseño de una arquitectura de software y la implementación de las estructuras de datos necesarias que almacenaran la información de cada fuente de iluminación en los eventos de interacción luz-materia. Por otro lado, MOCARTS realiza simulaciones en una rejilla rectangular lo que permite realizar simulaciones sin la suposición hecha en mcml sobre tejidos homogéneos. En general, estos avances proporcionan a la comunidad de investigación en el área de *imagen óptica difusa* el estudio de sistemas más complejos y más cercanos a la realidad como por ejemplo los sistemas de varios canales o incluso posiblemente los fNIRS High-Density.

➤ Flexibilización de la especificación del tejido

Se ha desarrollado una jerarquía de componentes biológicos para la especificación de tejido que se usa en una simulación. Esto, a través de la definición en un formato XML en el cual se definen los elementos abstractos: **rebanada biológica**, la cual está formada por un conjunto de **órganos** y estos a su vez están compuestos por **tejidos** que finalmente son descritos por un conjunto de **capas** las cuales especifican sus propiedades ópticas. De este modo, se permite una definición más intuitiva del modelo biológico a simular y permite su reutilización.

➤ Flexibilización de los datos de entrada/salida

La definición de parámetros de entrada y los datos de salida de una simulación se realizaron mediante la especificación de un archivo en formato XML que permitiera mayor legibilidad y una definición más intuitiva. Se analizó la información de entrada requerida para la simulación la cual fue dividida en tres partes: parámetros de simulación, modelo biológico y geometrías de sensado. En el caso de los datos de salida, ésta fue organizada en 4 secciones, primeramente los datos totales de reflectancia (R), transmitancia (T) y absorción (A), posteriormente las matrices: R, T, A asociadas a cada fuente de iluminación. Y finalmente la reflectancia medida por los sensores que hayan sido definidos. El formato y estructura de los archivos de entrada y salida simplifica su lectura ya sea de forma visual o mediante rutinas para ser cargados en otros sistemas para analizar la información. Esto

evita errores en la lectura de estos archivos en comparación con simuladores que usan archivos de texto plano.

## 7.2 Trabajos Futuros

Debido a las limitaciones definidas en este proyecto de tesis y aquellos puntos que fueron detectados en el transcurso de su desarrollo se proponen los siguientes trabajos futuros que se consideran importantes para la extensión y mejora de la herramienta de simulación:

- Se propone la inclusión de álgebra “sparse” que permita optimizar el espacio de memoria utilizado en el almacenamiento de los datos de salida de una simulación.
- Se plantea la paralelización del código para acelerar el proceso en el tiempo de ejecución del sistema. La opción para este punto es CUDA.
- En este trabajo se establecen las posiciones de las fuentes de iluminación en la superficie del modelo del tejido (pudiéndose posicionar en los ejes (x,y) sin ninguna restricción), sin embargo, una mejora a este punto sería poder localizar la fuente de luz en cualquier punto tridimensional (x,y,z). Permitiendo por ejemplo simulaciones para fusión de imágenes de endoscopía. Así como también su utilización en bioiluminiscencia.
- Desarrollo de una interfaz de usuario que permita mejorar la experiencia en el uso del sistema. En este trabajo se definió el paquete Interfaz Gráfica de Usuario (GUI) sin llegar a la implementación. Se desarrolló un bosquejo de dicha interfaz, la cual se muestra en la Figura 37.

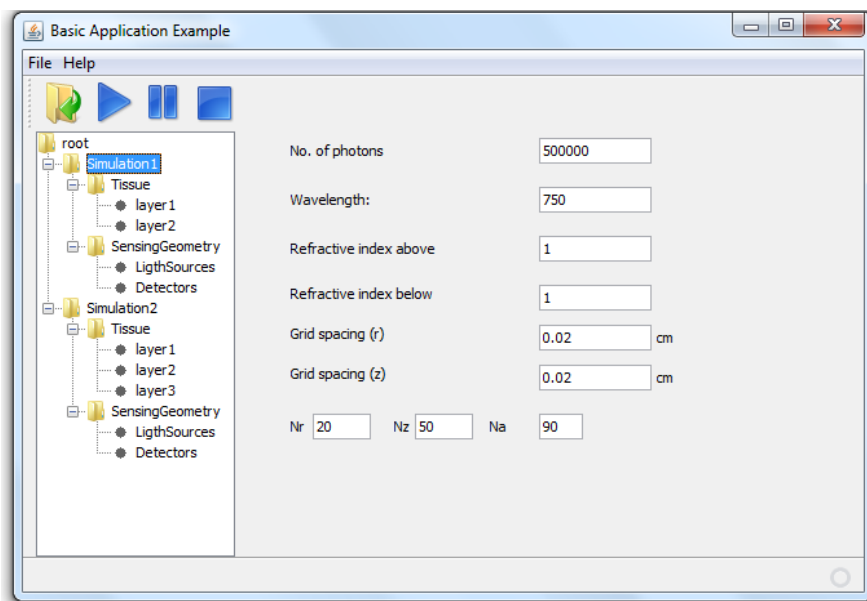


Figura 37 Interfaz Gráfica de Usuario de MOCARTS.

- En términos de la simulación, el desarrollo de los métodos necesarios que implementen la física necesaria para considerar fenómenos como fluorescencia y fosforescencia.

- Si bien para la simulación en MOCARTS se captura la luz en la superficie y debajo del modelo de tejido simulado, un trabajo futuro pudiera ser capturar la luz que sale por los lados del modelo de tejido simulado.
- La extensión a simular tejidos no homogéneos y geometrías complejas que permita que el resultado de una simulación se asemeje a mediciones reales<sup>2</sup>.

---

<sup>2</sup> Posterior al desarrollo de esta tesis (y previo a la evaluación) esta clase ya ha sido implementada por Javier Herrera Vega y constituye la primera ampliación a MOCARTS.



## Anexos

### Anexo 1 Archivo de definición de modelo de simulación

El archivo XML que se define a continuación corresponde a la descripción detallada de la Tabla 7 Parámetros de la rejilla y la Tabla 8 Propiedades ópticas del modelo de tejido simulado.

```
<?xml version="1.0" encoding="UTF-8" ?>
<MOCARTS>
  <Simulation name="TestSimulation">
    <Model name="BrainModel">
      <nabove>1.0</nabove>
      <nbelow>1.0</nbelow>
      <Organ name="Head">
        <Tissue name="Scalp">
          <Layer name="Dermis">
            <n>1.42</n>
            <mua>0.127</mua>
            <mus>190.376</mus>
            <g>0.900</g>
            <d>0.3</d>
          </Layer>
          <Tissue name="Skull">
            <Layer name="Bone">
              <n>1.555</n>
              <mua>0.147</mua>
              <mus>161.245</mus>
              <g>0.900</g>
              <d>0.500</d>
            </Layer>
            <Tissue name="Brain">
              <Layer name="GrayMatter">
                <n>1.360</n>
                <mua>0.270</mua>
                <mus>75.157</mus>
                <g>0.899</g>
                <d>0.4</d>
              </Layer>
              <Layer name="WhiteMatter">
                <n>1.380</n>
                <mua>0.931</mua>
                <mus>372.501</mus>
                <g>0.870</g>
                <d>0.2</d>
              </Layer>
            </Tissue>
          </Organ>
        </Model>
```

```
<SensingGeometry>
  <lightsource id="1">
    <x>0</x>
    <y>1</y>
  </lightsource>
  <lightsource id="2">
    <x>1</x>
    <y>0</y>
  </lightsource>
  <lightsource id="3">
    <x>0</x>
    <y>-1</y>
  </lightsource>
  <lightsource id="4">
    <x>-1</x>
    <y>0</y>
  </lightsource>
  <lightsource id="5">
    <x>0</x>
    <y>0</y>
  </lightsource>
  <detector id="1">
    <x>0.5</x>
    <y>0.5</y>
    <r>0.03</r>
    <Chanel>1</Chanel>
  </detector>
  <detector id="2">
    <x>0.5</x>
    <y>-0.5</y>
    <r>0.03</r>
    <Chanel>2</Chanel>
  </detector>
  <detector id="3">
    <x>-0.5</x>
    <y>-0.5</y>
    <r>0.03</r>
    <Chanel>3</Chanel>
  </detector>
  <detector id="4">
    <x>-0.5</x>
    <y>0.5</y>
    <r>0.03</r>
    <Chanel>4</Chanel>
  </detector>
</SensingGeometry>
<SimulationParams>
  <Grid type="Cartesian">
    <deltax>1</deltax>
    <deltay>1</deltay>
```

```
<deltaz>1</deltaz>
<Nx>40</Nx>
<Ny>40</Ny>
<Nz>24</Nz>
</Grid>
<optic>
  <photons>1000</photons>
  <wavelength units="nm">750</wavelength>
</optic>
<general>
  <fileOut>result.xml</fileOut>
</general>
</SimulationParams>
</Simulation>
</MOCARTS>
```

## Anexo 2 Fichero de salida del modelo simulado

```
<?xml version="1.0"?>
<Simulation>
  <RAT>
    <R> Value </R>
    <A> Value </A>
    <T> Value </T>
  </RAT>
  <Reflectance_xya>
    <Source id="1">Matrix</Source>
    <Source id="2">Matrix</Source>
    <Source id="3">Matrix</Source>
    <Source id="4">Matrix</Source>
    <Source id="5">Matrix</Source>
  </Reflectance_xya>
  <Absorption_xyz>
    <Source id="1">Matrix</Source>
    <Source id="2">Matrix</Source>
    <Source id="3">Matrix</Source>
    <Source id="4">Matrix</Source>
    <Source id="5">Matrix</Source>
  </Absorption_xyz>
  <Transmittance_xya>
    <Source id="1">Matrix</Source>
    <Source id="2">Matrix</Source>
    <Source id="3">Matrix</Source>
    <Source id="4">Matrix</Source>
    <Source id="5">Matrix</Source>
  </Transmittance_xya>
  <Detectors>
    <Detector id="1"> Value </Detector>
    <Detector id="2"> Value </Detector>
    <Detector id="3"> Value </Detector>
    <Detector id="4"> Value </Detector>
  </Detectors>
</Simulation>
```



## Anexo 3 Código fuente de MOCARTS

A continuación se lista el código fuente de las clases más importantes del simulador MOCARTS como la clase Simulation que es el núcleo del sistema y la cual está basada en el código original de *mcml* escrito en lenguaje C. Se lista también el código correspondiente a las clases relacionadas a la definición de las geometrías de sensado multicanal así como las estructuras de datos para la definición de las rejillas de simulación. El código completo y ejecutable de MOCARTS está disponible bajo petición en el sitio: <http://ccc.inaoep.mx/~f.oriuela-espina/Src/MOCARTS/>

```
package Simulation;

import SensingGeometry.LightSource;
import TissueModel.BiologicalSlab;
import TissueModel.HomogeneousLayer;
import Utility.Point3D;
import Utility.Utils;

/**
 * @author Bibiana
 */
public class Simulation {

    InParams inparams;
    OutParams outparams;
    long i_photon;
    // Declarar estructura de variables de salida
    Photon photon;
    long num_photons;
    long photon_rep = 10;
    short num_runs;
    static double COSZERO = (1.0 - 1.0E-12);
    static double CHANCE = 0.1;
    static double PI = 3.1415926;
    Grid myGrid = new Grid();

    public Simulation(InParams inparams, short num_runs) {
        this.inparams = inparams;
        this.num_runs = num_runs;
        this.num_photons = inparams.getNum_photons();
        outparams = new OutParams(this.inparams);
    }

    public void mainSimulationLoop() {
        while ((num_runs--) > 0) {
```

```
        DoOneRun(num_runs);
    }
}

/**
 * Implementation based in the original function DoOneRun in mcmlmain.c
 */
public void DoOneRun(short NumRuns) {
    for (int ns = 0; ns < inparams.getSourcesPositions().size(); ns++) {
        photon_rep = 10;
        CriticalAngle(inparams.getNum_layers(), inparams.getBiologicalSlab());
        outparams.setRsp(Rspecular(inparams.getBiologicalSlab()));
        i_photon = num_photons;
        long startTime = System.nanoTime();
        System.out.println("Light Source: " + ns);
        do {
            if (num_photons - i_photon == photon_rep) {
                System.out.println(i_photon + " photons & " + NumRuns + " runs left, ");
                photon_rep *= 10;
            }
            photon = LaunchPhoton(outparams.Rsp, inparams.getBiologicalSlab(), ((LightSource)
inparams.getSourcesPositions().get(ns)).getPosition());
            do {
                HopDropSpin(inparams, photon, outparams, ns);
            } while (!photon.isDead());
        } while ((--i_photon) > 0);
        long stopTime = System.nanoTime();
        long elapsedTime = stopTime - startTime;
        System.out.println("Time: " + elapsedTime * 0.000000001 + " Seconds");
    }
}

private Photon LaunchPhoton(double Rsp, BiologicalSlab slab, Point3D sourceLocation) {
    Photon photon = new Photon();
    photon.setWeight(1.0 - Rsp);
    photon.setDead(false);
    photon.setCurrLayer(slab.getLayers().get(1));
    photon.setS(0);
    photon.setRemainingStep(0);
    photon.setLocation(new Point3D(sourceLocation.getX(), sourceLocation.getY(),
sourceLocation.getZ()));
    photon.setDirection(0, 0, 1);

    if ((slab.getLayers().get(1).getMua() == 0.0)
        && (slab.getLayers().get(1).getMus() == 0.0)) {
        photon.setCurrLayer(slab.getLayers().get(2));
        photon.setLocation(new Point3D(photon.getLocation().getX(), photon.getLocation().getY(),
slab.getLayers().get(2).getZ0()));
    }
}
```

```

    }
    return photon;
}

private void HopDropSpin(InParams inparams, Photon photon, OutParams outparams, int sld) {

    if ((photon.getCurrLayer().getMua() == 0.0) && (photon.getCurrLayer().getMus() == 0.0)) {
        ;
        HopInGlass(inparams, photon, outparams, sld);
    } else {
        HopDropSpinInTissue(inparams, photon, outparams, sld);
    }

    if (photon.getWeight() < inparams.getWth() && !photon.isDead()) {
        Roulette(photon);
    }
}

/**
 * The implementation of this function is based on the same mcml code
 *
 * @param Layerspecs Struct that holds the layers
 * @return The specular reflectance computed accordingly to the refractive
 * indexes of each layer
 */
private double Rspecular(BiologicalSlab Layerspecs) {
    double r1, r2;
    double temp;
    double nAbove = Layerspecs.getLayers().get(0).getRefractiveIndex(null);
    double nfirstLayer = Layerspecs.getLayers().get(1).getRefractiveIndex(null);

    temp = (nAbove - nfirstLayer) / (nAbove + nfirstLayer);
    r1 = temp * temp;

    /*
     * Check if the first layers is a glass layer
     */
    if ((Layerspecs.getLayers().get(1).getMua() == 0.0)
        && (Layerspecs.getLayers().get(1).getMus() == 0.0)) {
        double nNextLayer = Layerspecs.getLayers().get(2).getRefractiveIndex(null);
        temp = (nfirstLayer - nNextLayer)
            / (nfirstLayer + nNextLayer);
        r2 = temp * temp;
        r1 = r1 + (1 - r1) * (1 - r1) * r2 / (1 - r1 * r2);
    }
    return r1;
}

```

```
public void CriticalAngle(short numLayers, BiologicalSlab slab) {
    short i = 0;
    double n1, n2;
    for (i = 1; i <= numLayers - 2; i++) {
        n1 = slab.getLayers().get(i).getRefractiveIndex(null);
        n2 = slab.getLayers().get(i - 1).getRefractiveIndex(null);
        ((HomogeneousLayer) slab.getLayers().get(i)).cos_crit0 = n1 > n2 ? Math.sqrt(1.0 - n2 * n2 / (n1 *
n1)) : 0.0;

        n2 = slab.getLayers().get(i + 1).getRefractiveIndex(null);
        ((HomogeneousLayer) slab.getLayers().get(i)).cos_crit1 = n1 > n2 ? Math.sqrt(1.0 - n2 * n2 / (n1 *
n1)) : 0.0;
    }
}

private void HopDropSpinInTissue(InParams inparams, Photon photon, OutParams outparams, int sld)
{
    StepSizeInTissue(photon, inparams);
    if (HitBoundary(photon, inparams)) {
        Hop(photon);
        CrossOrNot(inparams, photon, outparams, sld);
    } else {
        Hop(photon);
        Drop(inparams, photon, outparams, sld);
        Spin(photon.getCurrLayer().getAnisotropy (null), photon);
    }
}

private void StepSizeInTissue(Photon photon, InParams inparams) {
    double mua = photon.getCurrLayer().getMua();
    double mus = photon.getCurrLayer().getMus();
    if (photon.getRemainingStep() == 0) {
        double rnd;
        do {
            rnd = Utils.RandomNum();
        } while (rnd <= 0.0);
        photon.setS(-Math.log(rnd) / (mua + mus));
    } else {
        photon.setS(photon.getRemainingStep() / (mua + mus));
        photon.setRemainingStep(0.0);
    }
}

private boolean HitBoundary(Photon photon, InParams inparams) {
    double dl_b = 0;
    double uz = photon.getUz();
    boolean hit;
```

```

if (uz > 0.0) {
    dl_b = (photon.getCurrLayer().getZ1() - photon.getLocation().getZ()) / uz;
} else if (uz < 0.0) {
    dl_b = (photon.getCurrLayer().getZ0() - photon.getLocation().getZ()) / uz;
}

if (uz != 0.0 && photon.getS() > dl_b) {
    double mut = photon.getCurrLayer().getMua() + photon.getCurrLayer().getMus();

    photon.setRemainingStep((photon.getS() - dl_b) * mut);
    photon.setS(dl_b);

    hit = true;
} else {
    hit = false;
}

return (hit);
}

private void Hop(Photon photon) {
    double s = photon.getS();
    photon.getLocation().setX(photon.getLocation().getX() + (s * photon.getUx()));
    photon.getLocation().setY(photon.getLocation().getY() + (s * photon.getUy()));
    photon.getLocation().setZ(photon.getLocation().getZ() + (s * photon.getUz()));
}

private void CrossOrNot(InParams inparams, Photon photon, OutParams outparams, int sld) {
    if (photon.getUz() < 0.0) {
        CrossUpOrNot(inparams, photon, outparams, sld);
    } else {
        CrossDnOrNot(inparams, photon, outparams, sld);
    }
}

private void CrossUpOrNot(InParams inparams, Photon photon, OutParams outparams, int sld) {
    double uz = photon.getUz();
    double uz1 = 0.0;
    double r = 0.0;
    double ni = photon.getCurrLayer().getRefractiveIndex(null);
    double nt = inparams.getBiologicalSlab().getLayers().get(photon.getCurrLayer().getIdx() -
1).getRefractiveIndex(null);
    HomogeneousLayer tempLayer = (HomogeneousLayer) photon.getCurrLayer();
    if (-uz <= tempLayer.getCos_crit0()) {
        r = 1.0;
    } else {
        ReflectanceModel RModel = new Fresnel();
        double[] res = RModel.Reflectance(ni, nt, -uz);

```

```
    r = res[0];
    uz1 = res[1];
}

if (Utils.RandomNum() > r) {
    if (photon.getCurrLayer().getIdx() == 1) {
        photon.setUz(-uz1);
        RecordR(0.0, inparams, photon, outparams, sld);
        photon.setDead(true);
    } else {

photon.setCurrLayer(inparams.getBiologicalSlab().getLayers().get(photon.getCurrLayer().getIdx() - 1));
        photon.setUx((ni / nt) * photon.getUx());
        photon.setUy((ni / nt) * photon.getUy());
        photon.setUz(-uz1);
    }
} else {
    photon.setUz(-uz);
}
}

private void RecordR(double Refl, InParams inparams, Photon photon, OutParams outparams, int sld)
{
    double x = photon.getLocation().getX();
    double y = photon.getLocation().getY();

    int longX = inparams.getNx1();
    int longY = inparams.getNy1();

    short ix, iy;
    short ir, ia;
    double ixd, iyd;
    double ird, iad;

    ird = Math.sqrt(x * x + y * y) / inparams.getDr();
    if (ird > inparams.getNr() - 1) {
        ir = (short) (inparams.getNr() - 1);
    } else {
        ir = (short) ird;
    }

    iad = Math.acos(-photon.getUz()) / inparams.getDa();
    if (iad > inparams.getNa() - 1) {
        ia = (short) (inparams.getNa() - 1);
    } else {
        ia = (short) iad;
    }
}
```

```

iyd = photon.getLocation().getY() / inparams.getDy1();
iyd = iyd < 0 ? (short) ((longY / 2) - Math.abs(iyd)) : (short) ((longY / 2) + iyd);

if (iyd > inparams.getNy1() - 1) {
    iy = (short) (inparams.getNy1() - 1);
} else if (iyd < 0) {
    iy = 0;
} else {
    iy = (short) iyd;
}

ixd = photon.getLocation().getX() / inparams.getDx1();
ixd = ixd < 0 ? (short) ((longX / 2) - Math.abs(ixd)) : (short) ((longX / 2) + ixd);
if (ixd > inparams.getNx1() - 1) {
    ix = (short) (inparams.getNx1() - 1);
} else if (ixd < 0) {
    ix = 0;
} else {
    ix = (short) ixd;
}

outparams.setRd_xyaValue(sld, ix, iy, ia, photon.getWeight() * (1.0 - Refl));
photon.setWeight(Refl * photon.getWeight());
}

private void CrossDnOrNot(InParams inparams, Photon photon, OutParams outparams, int sld) {
    double uz = photon.getUz();
    double uz1 = 0.0;
    double r = 0.0;
    double ni = photon.getCurrLayer().getRefractiveIndex(null);
    double nt = 0;
    try {
        nt = inparams.getBiologicalSlab().getLayers().get(photon.getCurrLayer().getIdx() +
1).getRefractiveIndex(null);
    } catch (IndexOutOfBoundsException e) {
        System.out.println("Error in method CrossDnOrNot");
    }

    HomogeneousLayer tempLayer = (HomogeneousLayer) photon.getCurrLayer();
    if (uz <= tempLayer.getCos_crit1()) {
        r = 1.0;
    } else {
        ReflectanceModel RModel = new Fresnel();
        double[] res = RModel.Reflectance(ni, nt, uz);
        r = res[0];
        uz1 = res[1];
    }
}

```

```
if (Utils.RandomNum() > r) {
    if (photon.getCurrLayer().getIdx() == inparams.getNum_layers() - 2) {

        photon.setUz(uz1);
        RecordT(0.0, inparams, photon, outparams, sld);
        photon.setDead(true);
    } else {

photon.setCurrLayer(inparams.getBiologicalSlab().getLayers().get(photon.getCurrLayer().getIdx() + 1));
        photon.setUx((ni / nt) * photon.getUx());
        photon.setUy((ni / nt) * photon.getUy());
        photon.setUz(uz1);

    }
} else {
    photon.setUz(-uz);
}
}

private void RecordT(double Refl, InParams inparams, Photon photon, OutParams outparams, int sld) {
    double x = photon.getLocation().getX();
    double y = photon.getLocation().getY();

    int longX = inparams.getNx1();
    int longY = inparams.getNy1();

    short ix, iy;
    short ir, ia;
    double ixd, iyd;
    double ird, iad;
    ird = Math.sqrt(x * x + y * y) / inparams.getDr();
    if (ird > inparams.getNr() - 1) {
        ir = (short) (inparams.getNr() - 1);
    } else {
        ir = (short) ird;
    }

    iad = Math.acos(photon.getUz()) / inparams.getDa();
    if (iad > inparams.getNa() - 1) {
        ia = (short) (inparams.getNa() - 1);
    } else {
        ia = (short) iad;
    }

    iyd = photon.getLocation().getY() / inparams.getDy1();
    iyd = iyd < 0 ? (short) ((longY / 2) - Math.abs(iyd)) : (short) ((longY / 2) + iyd);

    if (iyd > inparams.getNy1() - 1) {
```

```

        iy = (short) (inparams.getNy1() - 1);
    } else if (iyd < 0) {
        iy = 0;
    } else {
        iy = (short) iyd;
    }

    idx = photon.getLocation().getX() / inparams.getDx1();
    idx = idx < 0 ? (short) ((longX / 2) - Math.abs(idx)) : (short) ((longX / 2) + idx);
    if (idx > inparams.getNx1() - 1) {
        ix = (short) (inparams.getNx1() - 1);
    } else if (idx < 0) {
        ix = 0;
    } else {
        ix = (short) idx;
    }
    outparams.setTt_xyaValue(sld, ix, iy, ia, photon.getWeight() * (1.0 - Refl));
    photon.setWeight(Refl * photon.getWeight());
}

private void Drop(InParams inparams, Photon photon, OutParams outparams, int sld) {
    double dwa;
    double x = photon.getLocation().getX();
    double y = photon.getLocation().getY();
    double izd, ird, idx, iyd;
    short iz, ir, ix, iy;
    double mua, mus;
    int longX = inparams.getNx1();
    int longY = inparams.getNy1();
    izd = photon.getLocation().getZ() / inparams.getDz1();
    if (izd > inparams.getNz1() - 1) {
        iz = (short) (inparams.getNz1() - 1);
    } else {
        iz = (short) izd;
    }
    iyd = photon.getLocation().getY() / inparams.getDy1();
    iyd = iyd < 0 ? (short) ((longY / 2) - Math.abs(iyd)) : (short) ((longY / 2) + iyd);
    if (iyd > inparams.getNy1() - 1) {
        iy = (short) (inparams.getNy1() - 1);
    } else if (iyd < 0) {
        iy = 0;
    } else {
        iy = (short) iyd;
    }
    idx = photon.getLocation().getX() / inparams.getDx1();
    idx = idx < 0 ? (short) ((longX / 2) - Math.abs(idx)) : (short) ((longX / 2) + idx);
    if (idx > inparams.getNx1() - 1) {

```

```
        ix = (short) (inparams.getNx1() - 1);
    } else if (ixd < 0) {
        ix = 0;
    } else {
        ix = (short) ixd;
    }
    mua = photon.getCurrLayer().getMua();
    mus = photon.getCurrLayer().getMus();
    dwa = (photon.getWeight() * mua) / (mua + mus);
    photon.setWeight(photon.getWeight() - dwa);
    outparams.setA_xyzValue(sId, ix, iy, iz, dwa);
}

private void Spin(double g, Photon photon) {

    double cost, sint;
    double cosp, sinp;
    double ux = photon.getUx();
    double uy = photon.getUy();
    double uz = photon.getUz();
    double psi;

    cost = SpinTheta(g);
    sint = Math.sqrt(1.0 - cost * cost);
    psi = 2.0 * PI * Utils.RandomNum();
    cosp = Math.cos(psi);
    if (psi < PI) {
        sinp = Math.sqrt(1.0 - cosp * cosp);
    } else {
        sinp = -Math.sqrt(1.0 - cosp * cosp);
    }

    if (Math.abs(uz) > COSZERO) {
        photon.setUx(sint * cosp);
        photon.setUy(sint * sinp);
        photon.setUz(cost * Math.signum(uz));
    } else {
        double temp = Math.sqrt(1.0 - uz * uz);
        photon.setUx(sint * (ux * uz * cosp - uy * sinp) / temp + ux * cost);
        photon.setUy(sint * (uy * uz * cosp + ux * sinp) / temp + uy * cost);
        photon.setUz(-sint * cosp * temp + uz * cost);
    }
}

private double SpinTheta(double g) {
    double cost;
    if (g == 0.0) {
        cost = 2 * Utils.RandomNum() - 1;
    }
```

```

    } else {
        double temp = (1 - g * g) / (1 - g + 2 * g * Utils.RandomNum());
        cost = (1 + g * g - temp * temp) / (2 * g);
        if (cost < -1) {
            cost = -1;
        } else if (cost > 1) {
            cost = 1;
        }
    }
}
return (cost);
}

private void Roulette(Photon photon) {
    if (photon.getWeight() == 0.0) {
        photon.setDead(true);
    } else if (Utils.RandomNum() < CHANCE) {
        photon.setWeight(photon.getWeight() / CHANCE);
    } else {
        photon.setDead(true);
    }
}

private void HopInGlass(InParams inparams, Photon photon, OutParams outparams, int sld) {
    double dl;
    if (photon.getUz() == 0.0) {
        photon.setDead(true);
    } else {
        StepSizeInGlass(photon, inparams);
        Hop(photon);
        CrossOrNot(inparams, photon, outparams, sld);
    }
}

private void StepSizeInGlass(Photon photon, InParams inparams) {
    double dl_b;
    double uz = photon.getUz();
    if (uz > 0.0) {
        dl_b = (photon.getCurrLayer().getZ1() - photon.getLocation().getZ()) / uz;
    } else if (uz < 0.0) {
        dl_b = (photon.getCurrLayer().getZ0() - photon.getLocation().getZ()) / uz;
    } else {
        dl_b = 0.0;
    }
    photon.setS(dl_b);
}

public OutParams getOutparams() {
    return outparams;
}

```

```
    }  
}  
  
package Simulation;  
  
import java.io.File;  
import java.io.FileNotFoundException;  
import java.io.PrintStream;  
import java.util.ArrayList;  
import java.util.Iterator;  
import java.util.logging.Level;  
import java.util.logging.Logger;  
  
/**  
 * We should keep the array type of the output variables since this struct  
 * (array) is faster than List objects (however I need to investigate for a  
 * faster access struct, if any, to improve the performance)  
 *  
 * @author Bibiana  
 */  
public class OutParams {  
  
    InParams inparams;  
    double Rsp;    /** specular reflectance. [-]*/  
    double Rd;    /** total diffuse reflectance. [-]*/  
    double A;    /** total absorption probability. [-]*/  
    double Tt;    /** total transmittance. [-]*/  
  
    ArrayList[][] Cell_Rd_xya; /**Reflectance in tissue in function of the output angle a*/  
    ArrayList[][] Cell_Tt_xya; /**Transmittance in tissue in function of the output angle a*/  
    ArrayList[][] Cell_A_xyz; /** 3D grid to store absorption in tissue*/  
  
    public OutParams(InParams inparams) {  
        this.inparams = inparams;  
        this.Rsp = 0.0;  
        this.Rd = 0.0;  
        this.A = 0.0;  
        this.Tt = 0.0;  
        initArrays();  
    }  
  
    /**  
     * Instantiate the output variables with the corresponding size. As defined  
     * by the members of the InParams object.  
     */  
    private void initArrays() {  
  
        this.Cell_Rd_xya = new ArrayList[inparams.getNx1()][inparams.getNy1()][inparams.getNa()];
```

```

this.Cell_Tt_xya = new ArrayList[inparams.getNx1()][inparams.getNy1()][inparams.getNa()];
this.Cell_A_xyz = new ArrayList[inparams.getNx1()][inparams.getNy1()][inparams.getNz1()];
}

public void setA_xyzValue(int sld, int ix, int iy, int iz, double val) {
    double value = val;

    if (this.Cell_A_xyz[ix][iy][iz] == null) {
        this.Cell_A_xyz[ix][iy][iz] = new ArrayList<Cell>();
        this.Cell_A_xyz[ix][iy][iz].add(new Cell((byte) sld, value));
    } else {
        boolean found = false;
        for (int i = 0; i < this.Cell_A_xyz[ix][iy][iz].size(); i++) {
            if (((Cell) this.Cell_A_xyz[ix][iy][iz].get(i)).getSld() == sld) {
                ((Cell) this.Cell_A_xyz[ix][iy][iz].get(i)).addValue(val);
                found = true;
                break;
            }
        }
        if (!found) {
            this.Cell_A_xyz[ix][iy][iz].add(new Cell((byte) sld, value));
        }
    }
}

public void setRd_xyaValue(int sld, int ix, int iy, int ia, double val) {
    double value = val;

    if (this.Cell_Rd_xya[ix][iy][ia] == null) {
        this.Cell_Rd_xya[ix][iy][ia] = new ArrayList<Cell>();
        this.Cell_Rd_xya[ix][iy][ia].add(new Cell((byte) sld, value));
    } else {
        boolean found = false;
        for (int i = 0; i < this.Cell_Rd_xya[ix][iy][ia].size(); i++) {
            if (((Cell) this.Cell_Rd_xya[ix][iy][ia].get(i)).getSld() == sld) {
                ((Cell) this.Cell_Rd_xya[ix][iy][ia].get(i)).addValue(val);
                found = true;
                break;
            }
        }
        if (!found) {
            this.Cell_Rd_xya[ix][iy][ia].add(new Cell((byte) sld, value));
        }
    }
}

```

```
public void setTt_xyaValue(int sld, int ix, int iy, int ia, double val) {
    double value = val;

    if (this.Cell_Tt_xya[ix][iy][ia] == null) {
        this.Cell_Tt_xya[ix][iy][ia] = new ArrayList<Cell>();
        this.Cell_Tt_xya[ix][iy][ia].add(new Cell((byte) sld, value));
    } else {
        boolean found = false;
        for (int i = 0; i < this.Cell_Tt_xya[ix][iy][ia].size(); i++) {
            if (((Cell) this.Cell_Tt_xya[ix][iy][ia].get(i)).getSld() == sld) {
                ((Cell) this.Cell_Tt_xya[ix][iy][ia].get(i)).addValue(val);
                found = true;
                break;
            }
        }
        if (!found) {
            this.Cell_Tt_xya[ix][iy][ia].add(new Cell((byte) sld, value));
        }
    }
}
```

\*\*\*\*\*

```
package SensingGeometry;
```

```
import Simulation.Cell;
import Simulation.InParams;
import Utility.Point3D;
import java.util.ArrayList;
```

```
/**
```

```
*
```

```
* @author Bibiana
```

```
*/
```

```
public class Detector {
```

```
    private Point3D position;
```

```
    private double radius;
```

```
    private int id;
```

```
    private int sourceId;
```

```
    public Detector(int id) {
```

```
        this.id = id;
```

```
        position = new Point3D(0, 0, 0); //Default position
```

```
    }
```

```

public Detector(Point3D position, double radious, int id) {
    this.position = position;
    this.radious = radious;
    this.id = id;
}

/**
 * Computes the diffuse reflectance collected by this detector
 *
 * @param mat The matrix where the diffuse reflectance has been computed and
 * stored
 * @return The diffuse reflectance Rd sensed by this detector.
 */
public double getRd(ArrayList[][][] mat, InParams inparams, int sld) {
    int longX = inparams.getNx1();
    int longY = inparams.getNy1();
    short ix, iy, ia;
    double ixd, iyd, iad;

    iyd = this.getPosition().getY() / inparams.getDy1();
    iyd = iyd < 0 ? (short) ((longY / 2) - Math.abs(iyd)) : (short) ((longY / 2) + iyd);

    if (iyd > inparams.getNy1() - 1) {
        iy = (short) (inparams.getNy1() - 1);
    } else if (iyd < 0) {
        iy = 0;
    } else {
        iy = (short) iyd;
    }

    ixd = this.getPosition().getX() / inparams.getDx1();
    ixd = ixd < 0 ? (short) ((longX / 2) - Math.abs(ixd)) : (short) ((longX / 2) + ixd);
    if (ixd > inparams.getNx1() - 1) {
        ix = (short) (inparams.getNx1() - 1);
    } else if (ixd < 0) {
        ix = 0;
    } else {
        ix = (short) ixd;
    }

    double Rd = 0;
    int Nx = (int) (radious / inparams.getDx1());

    for (int i = ix - Nx; i <= ix + Nx; i++) {
        for (int j = iy - Nx; j <= iy + Nx; j++) {
            for (int a = 0; a < mat[ix][iy].length; a++) {
                if (mat[i][j][a] != null) {
                    for (int k = 0; k < mat[i][j][a].size(); k++) {

```

```
        if (((Cell) mat[i][j][a].get(k)).getSld() == sld) {
            Rd += ((Cell) mat[i][j][a].get(k)).getValue();
            break;
        }
    }
}
}
}
return Rd;
}

public double getRdAllSources(ArrayList[][][] mat, InParams inparams) {
    int longX = inparams.getNx1();
    int longY = inparams.getNy1();
    short ix, iy, ia;
    double ixd, iyd, iad;

    iyd = this.getPosition().getY() / inparams.getDy1();
    iyd = iyd < 0 ? (short) ((longY / 2) - Math.abs(iyd)) : (short) ((longY / 2) + iyd);

    if (iyd > inparams.getNy1() - 1) {
        iy = (short) (inparams.getNy1() - 1);
    } else if (iyd < 0) {
        iy = 0;
    } else {
        iy = (short) iyd;
    }

    ixd = this.getPosition().getX() / inparams.getDx1();
    ixd = ixd < 0 ? (short) ((longX / 2) - Math.abs(ixd)) : (short) ((longX / 2) + ixd);
    if (ixd > inparams.getNx1() - 1) {
        ix = (short) (inparams.getNx1() - 1);
    } else if (ixd < 0) {
        ix = 0;
    } else {
        ix = (short) ixd;
    }

    double Rd = 0;
    int Nx = (int) (radius / inparams.getDx1());

    for (int i = ix - Nx; i <= ix + Nx; i++) {
        for (int j = iy - Nx; j <= iy + Nx; j++) {
            for (int a = 0; a < mat[ix][iy].length; a++) {
                if (mat[i][j][a] != null) {
                    for (int k = 0; k < mat[i][j][a].size(); k++) {
                        Rd += ((Cell) mat[i][j][a].get(k)).getValue();
                    }
                }
            }
        }
    }
}
```

```

        break;
    }
}
}
}
return Rd;
}
}
}

```

---

```

package SensingGeometry;

import Utility.Point3D;

/**
 *
 * @author Bibiana
 */
public class LightSource {
    private Point3D position;
    private long nphotons;
    private int sld;

    public LightSource(Point3D position, long nphotons, int sld) {
        this.position = position;
        this.nphotons = nphotons;
        this.sld=sld;
        position=new Point3D(0,0,0);
    }

    public LightSource(int id) {
        this.sld=id;
        position=new Point3D(0,0,0); //Default position
    }

    public long getNphotons() {
        return nphotons;
    }

    public void setNphotons(long nphotons) {
        this.nphotons = nphotons;
    }

    public Point3D getPosition() {
        return position;
    }
}

```

```
public void setPosition(Point3D position) {  
    this.position = position;  
}  
}
```

## Glosario

Espectroscopía	Estudio de las propiedades físicas de un objeto a través del estudio de su espectro electromagnético, ya sea por la luz emitida o absorbida por él.
Espectro electromagnético	Es el conjunto de todos los tipos de ondas electromagnéticas, las cuales se diferencian por el número de veces que se repite la onda por unidad de tiempo (frecuencia). Está constituido por las ondas de radio, microondas, infrarrojos, luz visible, ultravioleta, rayos X y rayos gamma.
Radiación electromagnética	Combinación de campos eléctricos y magnéticos oscilantes, que se propagan a través del espacio transportando energía de un lugar a otro.
Biomédico	Estudio de los aspectos biológicos de la medicina. Su objetivo fundamental es investigar los mecanismos moleculares, bioquímicos, celulares y genéticos de las enfermedades humanas.
Atenuación óptica	Es la extinción de la luz por absorción y dispersión.
Cromóforo	Conjunto de átomos de una molécula responsable de su color. El cromóforo es la semimolécula que causa un cambio en la conformación del conjunto al recibir luz.
Espectro de extinción	Atenuación a través de un conjunto de longitudes de onda.
Imagen óptica difusa	Modalidad de bioimagen funcional con una alta versatilidad aplicable a diferentes tejidos de forma no invasiva, basada en la extinción de la luz al interactuar con la materia.
Reflexión	Tiene que ver con el acto de reflejarse un rayo de luz sobre una superficie. Puede ser especular o difusa dependiendo de la relación entre los ángulos de entrada y salida.
Transmitancia	Cantidad de energía que atraviesa un cuerpo.
Refracción	Se produce cuando la luz pasa de un medio de propagación a otro y sufre un cambio de velocidad y dirección de su trayectoria.
Absorción	Proceso por el cual la radiación es captada por la materia. Es precisamente este proceso de absorción y posterior remisión de la luz visible lo que da color a la materia.
Esparcimiento	Es el fenómeno por el cual hay una desviación del fotón de la dirección original del haz de luz por la colisión con partículas de materia en el medio.

Coeficiente de absorción	Indica la cantidad de luz que absorbe una superficie en relación a su tamaño.
Coeficiente de dispersión o esparcimiento	Indica la capacidad macroscópica del tejido para dispersar un fotón.
Índice de refracción	Es una medida que determina la reducción de la velocidad de la luz al propagarse por un medio.
Anisotropía de esparcimiento	Indica la distribución de probabilidad de la nueva dirección del fotón en cada evento de esparcimiento.
Ecuaciones de Maxwell	Son un conjunto de ecuaciones que describen por completo los fenómenos electromagnéticos. Introduciendo los conceptos de campo y corriente de desplazamiento, y unificando los campos eléctricos y magnéticos en un solo concepto: el campo electromagnético.
Método de Monte Carlo	Es una técnica que combina conceptos estadísticos (muestreo aleatorio) con la capacidad que tienen los ordenadores para generar números pseudo-aleatorios y automatizar cálculos.
In vivo	En ciencia, in vivo se refiere a experimentación hecha dentro o en el tejido de un organismo vivo, por oposición a uno parcial o muerto.
Ex vivo	En ciencia, ex vivo se refiere a los experimentos o medidas realizados en o sobre tejidos biológicos de un organismo en un ambiente artificial fuera del organismo con las alteraciones mínimas de las condiciones naturales.
Hemodinámica	Parte de la fisiología que estudia las leyes y mecanismos que rigen la circulación sanguínea.
Histología	Parte de la biología que estudia la estructura y características de los tejidos de los seres vivos.
Imaginología	Estudio y utilización clínica de las imágenes producidas por los rayos X, el ultrasonido, la resonancia magnética, etc.
Screening	Anglicismo utilizado para indicar una estrategia aplicada sobre una población para detectar una enfermedad en individuos sin signos o síntomas de esa enfermedad.
Fotosensor	Sensor que genera una señal eléctrica dependiente de la luz u otra radiación electromagnética que recibe.
Dispersión elástica	El resultado de un evento de dispersión es independiente de los demás.
Gold standard	Estándar de oro, patrón de oro, patrón de referencia. Es la técnica

---

	diagnóstica que define la presencia de la condición con la máxima certeza conocida.
Polarización de la radiación	Es el ángulo de oscilación de la onda electromagnética con respecto a la dirección de propagación de la energía
RMSE	Root Mean Squared Error, por sus siglas en inglés, mide el promedio de los errores al cuadrado, es decir, la diferencia entre el estimador y lo que se estima.



## Bibliografía

- [1] E. D. Olsen, «Métodos Ópticos de Análisis,» de *Métodos Ópticos de Análisis*, Barcelona, Reverté S.A., 1990.
- [2] J. Herrera-Vega y F. Orihuela-Espina, «Reporte Técnico - Image Reconstruction in Functional Optical Neuroimage: The modelling and separation of the scalp blood flow,» Instituto Nacional de Astrofísica, Óptica y Electrónica, 2015.
- [3] S. L. Jacques, «Corrigendum: Optical properties of biological tissues: a review,» *Phys. Med. Biol.*, vol. 58, nº 14, pp. 5007--5008, 2013.
- [4] R. Richards-Kortum y E. Sevick-Muraca, «Quantitative optical spectroscopy for tissue diagnosis,» *Annu. Rev. Phys. Chem.*, p. 47:555–606, 1996..
- [5] S. L. Jacques y B. W. Pogue, «Tutorial on diffuse light transport,» *Journal of Biomedical Optics*, vol. 13, nº 4, pp. 41302-41302, 2008.
- [6] J. H. Nobbs, «Kubelka—Munk theory and the prediction of reflectance,» *Review of Progress in Coloration and Related Topics*, vol. 15, nº 1, pp. 66-75, 1985.
- [7] L. Wang y S. L. Jacques, «Monte Carlo modeling of light transport in multi-layered tissues in standard C,» *The University of Texas, MD Anderson Cancer Center, Houston*, 1992.
- [8] S. R. Arridge y J. C. Hebden, «Optical imaging in medicine: II. Modelling and reconstruction,» *Physics in Medicine and Biology*, vol. 42, nº 5, p. 841, 1997.
- [9] D. R. Leff, C. E. Elwell, F. Orihuela-Espina, L. Atallah, D. T. Delpy, A. W. Darzi y G. Z. Yang, «Changes in prefrontal cortical behaviour depend upon familiarity on a bimanual co-ordination task: an fNIRS study,» *Neuroimage*, vol. 39, nº 2, pp. 805-813, 2008.
- [10] C. R. Simpson, M. Kohl, M. Essenpreis y M. Cope, «Near-infrared optical properties of ex vivo human skin and subcutaneous tissues measured using the Monte Carlo inversion technique,» *Physics in medicine and biology*, vol. 43, nº 9, p. 2465, 1998.
- [11] S. A. Prahl, M. Keijzer, S. L. Jacques y A. J. Welch, «A Monte Carlo model of light propagation in tissue,» *Dosimetry of laser radiation in medicine and biology*, vol. 5, pp. 102-111, 1989.
- [12] D. M. Hueber, M. A. Franceschini, H. Y. Ma, Q. Zhang, B. J R y V. Ntziachristos, «Non-invasive and quantitative near-infrared haemoglobin spectrometry in the piglet brain during hypoxic stress, using a frequency-domain multidistance instrument.,» *Phys Med*

- Biol.*, vol. 46, pp. 41-62, Jan 2001.
- [13] E. Claridge, S. Cotton, P. Hall y M. Moncrieff, «From colour to tissue histology: Physics-based interpretation of images of pigmented skin lesions.,» *Med Image Anal*, vol. 7, n° 4, pp. 489-502, Dec 2003.
- [14] E. Claridge y D. Hidovic-Rowe, «Model based inversion for deriving maps of histological parameters characteristic of cancer from ex-vivo multispectral images of the colon.,» *IEEE Trans Med Imaging*, vol. 33, n° 4, pp. 822-835, Apr 2014.
- [15] A. G. Yodh, *Diffuse Optics: Fundamentals & Tissue Applications*, U. o. Pennsylvania, Ed., Lake Como, Varenna: CLXXIII International School of Physics "Enrico Fermi" on, 2009, p. 115.
- [16] F. Orihuela-Espina, «Unit I. Introduction to biomedical imaging,» 2015. [En línea]. Available: [http://ccc.inaoep.mx/~foe/Docs/BiomedicalImaging/UnitI\\_IntroToBiomedicalImaging.pptx](http://ccc.inaoep.mx/~foe/Docs/BiomedicalImaging/UnitI_IntroToBiomedicalImaging.pptx). [Último acceso: Mayo 2015].
- [17] V. P. Coletta, *Physics Fundamentals 2nd*, Physics Curriculum & Instruction, Inc., 2010.
- [18] «Radio Electrónica,» [En línea]. Available: <http://www.radioelectronica.es/articulos-teoricos/52-las-ondas-1>. [Último acceso: 15 08 2014].
- [19] P. Collier, *A Most Incomprehensible Thing: Notes Towards a Very Gentle Introduction to the mathematics of relativity*, Incomprehensible Books, 2013.
- [20] wikipedia, «Luz,» [En línea]. Available: <http://es.wikipedia.org/wiki/Luz>. [Último acceso: 21 Julio 2014].
- [21] «Química Web,» [En línea]. Available: <http://www.quimicaweb.net>. [Último acceso: 18 08 2014].
- [22] A. Villringer y B. Chance, «Non-invasive optical spectroscopy and imaging of human brain function,» *Trends in neurosciences*, vol. 20, n° 10, pp. 435-442, 1997.
- [23] F. Scholkmann, S. Kleiser, A. J. Metz, R. Zimmermann, J. M. Pavia, U. Wolf y M. Wolf, «A review on continuous wave functional near-infrared spectroscopy and imaging instrumentation and methodology,» *Neuroimage*, vol. 85, pp. 6-27, 2014.
- [24] F. Orihuela-Espina, «Modelling and verification of the diffuse reflectance of the ocular fundus,» 2005.
- [25] P. Rolfe, «In vivo near-infrared spectroscopy,» *Annual review of biomedical*

- engineering*, vol. 2, n° 1, pp. 715-754, 2000.
- [26] S. Tominaga, «Surface reflectance estimation by the dichromatic model,» *Color Research & Application*, vol. 21, n° 2, pp. 104-114, 1996.
- [27] S. L. Jacques y S. A. Prah, «Henry-Greenstein scattering function,» Oregon Medical Laser Center, 10 2014. [En línea]. Available: <http://omlc.org/education/ece532/class3/hg.html>. [Último acceso: 1 10 2014].
- [28] S. L. Jacques, «Time resolved propagation of ultrashort laser pulses within turbid tissues,» *Applied optics*, vol. 28, n° 12, pp. 2223-2229, 1989.
- [29] H. Dehghani, M. E. Eames, P. K. Yalavarthy, S. C. Davis, S. Srinivasan, C. M. Carpenter, B. W. Pogue y K. D. Paulsen, «Near infrared optical tomography using NIRFAST: Algorithm for numerical model and image reconstruction,» *Communications in numerical methods in engineering*, vol. 25, n° 6, pp. 711-732, 2009.
- [30] M. Schweiger y S. Arridge, «The Toast++ software suite for forward and inverse modeling in optical tomography,» *Journal of biomedical optics*, vol. 19, n° 4, pp. 40801-40801, 2014.
- [31] A. Badal, I. Kyprianou, D. P. Banh, A. Badano y J. Sempau, «penMesh—Monte Carlo radiation transport simulation in a triangle mesh geometry,» *Medical Imaging, IEEE Transactions on*, vol. 28, n° 12, pp. 1894-1901, 2009.
- [32] Q. Fang y D. A. Boas, «Monte Carlo simulation of photon migration in 3D turbid media accelerated by graphics processing units,» *Optics express*, vol. 17, n° 22, pp. 20178-20190, 2009.
- [33] E. Alerstam, W. C. Yip Lo, T. D. Han, J. Rose, S. Andersson-Engels y L. Lilge, «Next-generation acceleration and code optimization for light transport in turbid media using GPUs,» *Biomedical optics express*, vol. 1, n° 2, pp. 658-675, 2010.
- [34] M. Keijzer, S. L. Jacques, S. A. Prah y A. J. Welch, «Light distributions in artery tissue: Monte Carlo simulations for finite-diameter laser beams,» *Lasers in surgery and medicine*, vol. 9, n° 2, pp. 148-154, 1989.
- [35] L. Wang y S. L. Jacques, «Optimized radial and angular positions in Monte Carlo modeling,» *Medical physics*, vol. 21, n° 7, pp. 1081-1083, 1994.
- [36] O. I. Laboratory, «mcm citations,» 2013. [En línea]. Available: <http://oilab.seas.wustl.edu/mcm citations.html>. [Último acceso: 30 Enero 2014].
- [37] R. S. Pressman, *Software Engineering: A Practitioner's Approach 6th Edition*, McGraw-

- Hill Science/Engineering/Math, 2007.
- [38] J. Lewis, *Java Software Solutions: Foundations of Program Design*, Pearson, 2011.
- [39] C. Horstmann, *Big Java*, John Wiley & Sons, 2007, p. 1248.
- [40] G. d. Aragón, «Plataforma e-educativa aragonesa,» Gobierno de Aragón, [En línea]. Available: [http://e-educativa.catedu.es/44700165/aula/archivos/repositorio/500/526/html/Unidad\\_04/pagina\\_1.html](http://e-educativa.catedu.es/44700165/aula/archivos/repositorio/500/526/html/Unidad_04/pagina_1.html). [Último acceso: 3 04 2015].
- [41] «Centralx Atlas,» [En línea]. Available: <http://www.centralx.es/p/imagen/celulas/>. [Último acceso: 19 05 2015].
- [42] Wikipedia, «Tejido biológico,» [En línea]. Available: [http://es.wikipedia.org/wiki/Tejido\\_%28biolog%C3%ADa%29#/media/File:Peripheral\\_nerve,\\_cross\\_section.jpg](http://es.wikipedia.org/wiki/Tejido_%28biolog%C3%ADa%29#/media/File:Peripheral_nerve,_cross_section.jpg). [Último acceso: 19 05 2015].
- [43] rae, «Definición de capa,» [En línea]. Available: <http://buscon.rae.es/drae/srv/search?id=aTfziMGXQDXX24J2W6B2>. [Último acceso: 1 Junio 2015].
- [44] «Sustancia gris,» [En línea]. Available: <http://footage.framepool.com/es/shot/253152250-sustancia-gris-sustancia-blanca-gyrus-craneo>. [Último acceso: 1 05 2015].
- [45] «ISS focus and discover,» [En línea]. Available: <http://www.iss.com/biomedical/instruments/imagent.html>. [Último acceso: 10 2014].
- [46] T. Vo-Dinh, *Biomedical Photonics Handbook: Biomedical Diagnostics*, vol. 2, CRC press, 2014, p. 1872 .
- [47] «MedLine Plus,» [En línea]. Available: <https://www.nlm.nih.gov/medlineplus/spanish/ency/article/000028.htm>. [Último acceso: 2014].
- [48] W.-F. Cheong, S. A. Prahl, A. J. Welch y others, «A review of the optical properties of biological tissues,» *IEEE journal of quantum electronics*, vol. 26, nº 12, pp. 2166-2185, 1990.
- [49] S. L. Jacques, «Time-resolved reflectance spectroscopy in turbid tissues,» *Biomedical Engineering, IEEE Transactions on*, vol. 36, nº 12, pp. 1155-1161, 1989.
- [50] M. Keijzer, R. R. Richards-Kortum, S. L. Jacques y M. S. Feld, «Fluorescence spectroscopy of turbid media: autofluorescence of the human aorta,» *Applied Optics*,

- vol. 28, nº 20, pp. 4286-4292, 1989.
- [51] G. Strangman, D. A. Boas y J. P. Sutton, «Non-invasive neuroimaging using near-infrared light,» *Biological psychiatry*, vol. 52, nº 7, pp. 679-693, 2002.
- [52] L. Wang y S. L. Jacques, «Hybrid model of Monte Carlo simulation and diffusion theory for light reflectance by turbid media,» *JOSA A*, vol. 10, nº 8, pp. 1746-1752, 1993.
- [53] E. I. Hang, S. Gracia y Y. V. Peris, «Biología. La ciencia de la vida,» Universidad Nacional del Nordeste, 10 2005. [En línea]. Available: [http://www.unne.edu.ar/articulacion/documentos/art2\\_biologia.pdf](http://www.unne.edu.ar/articulacion/documentos/art2_biologia.pdf). [Último acceso: 1 Agosto 2014].
- [54] S. Ramírez-García, P. H. Carranza-Castro, J. G.-O. L. Gutiérrez-Salinas y S. Hernández-Rodríguez, «Aplicación en medicina de la espectroscopia de infrarrojo cercano,» *Medicina Interna de México*, vol. 28, nº 4, p. 366, 2012.