



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**

**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**

**CARRERA TELECOMUNICACIONES**

**“DESARROLLO DE UN SIMULADOR DE PROPAGACIÓN 3D  
INDOOR EN FRECUENCIAS DE 10-20 GHZ SOBRE GPU CON  
NVIDIA-OPTIX BAJO EL MOTOR GRÁFICO UNITY PARA LA  
FIE”**

**Trabajo de Titulación**

Tipo: Proyecto de Investigación

Presentado para optar al grado académico de:

**INGENIERO EN ELECTRÓNICA, TELECOMUNICACIONES Y  
REDES**

**AUTOR:**

**KEVIN ISRAEL FREIRE LEMA**

Riobamba – Ecuador

2022



**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**

**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**

**CARRERA TELECOMUNICACIONES**

**“DESARROLLO DE UN SIMULADOR DE PROPAGACIÓN 3D  
INDOOR EN FRECUENCIAS DE 10-20 GHZ SOBRE GPU CON  
NVIDIA-OPTIX BAJO EL MOTOR GRÁFICO UNITY PARA LA  
FIE”**

**Trabajo de Titulación**

Tipo: Proyecto de Investigación

Presentado para optar al grado académico de:

**INGENIERO EN ELECTRÓNICA, TELECOMUNICACIONES Y  
REDES**

**AUTOR: KEVIN ISRAEL FREIRE LEMA**

**DIRECTOR: ING. JEFFERSON ALEXANDER RIBADENEIRA RAMIREZ, PhD.**

Riobamba – Ecuador

2022

**©2022, Kevin Israel Freire Lema**

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

Yo, **KEVIN ISRAEL FREIRE LEMA**, declaro que el presente trabajo de titulación es de mi autoría y los resultados de este son auténticos. Los textos en el documento que provienen de otras fuentes están debidamente citados y referenciados.

Como autor, asumo la responsabilidad legal y académica de los contenidos de este trabajo de titulación; el patrimonio intelectual pertenece a la Escuela Superior Politécnica de Chimborazo.

Riobamba, 29 de junio de 2022



**Kevin Israel Freire Lema**

**180494950-9**

**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**CARRERA TELECOMUNICACIONES**

El Tribunal del Trabajo de Titulación certifica que: El Trabajo de Titulación: Tipo: Proyecto de Investigación, “**DESARROLLO DE UN SIMULADOR DE PROPAGACIÓN 3D INDOOR EN FRECUENCIAS DE 10-20 GHZ SOBRE GPU CON NVIDIA-OPTIX BAJO EL MOTOR GRÁFICO UNITY PARA LA FIE**”, realizado por el señor: **KEVIN ISRAEL FREIRE LEMA**, ha sido minuciosamente revisado por los Miembros del Tribunal del Trabajo de Titulación, el mismo que cumple con los requisitos científicos, técnicos, legales, en tal virtud el Tribunal Autoriza su presentación.

<b>NOMBRE</b>	<b>FIRMA</b>	<b>FECHA</b>
Ing. Alberto Leopoldo Arellano Aucancela <b>PRESIDENTE DEL TRIBUNAL</b>		<u>29-06-2022</u>
Ing. Jefferson Alexander Ribadeneira Ramírez <b>DIRECTOR DEL TRABAJO DE TITULACIÓN</b>		<u>29-06-2022</u>
Ing. Pedro Severo Infante Moreira <b>MIEMBRO DEL TRIBUNAL</b>		<u>29-06-2022</u>

## **DEDICATORIA**

El presente trabajo lo dedico a mis padres, Marco Antonio Freire Porras y Adriana del Rocío Lema Hernández, quienes desde niño me han inculcado los mejores valores, enseñado la importancia del estudio y que todo esfuerzo tiene su recompensa. Además de brindarme su apoyo incondicional en cada etapa de mi vida, y, sobre todo, desde que nací se sacrificaron día tras día para que yo pueda ser la mejor versión de mí mismo.

Kevin

## **AGRADECIMIENTO**

Agradezco a mis padres, mis hermanos y demás familiares por su constante apoyo y siempre haber confiado en mí. A todos mis amigos por haber hecho de mi vida universitaria una verdadera aventura y siempre estar ahí en los momentos difíciles. A la Escuela Superior Politécnica de Chimborazo por haberme brindado la oportunidad de forjarme como profesional dentro de sus aulas y de igual manera, a todos los docentes de quien tuve el honor de haber sido estudiante, especialmente al ingeniero Jefferson Ribadeneira PhD., por haberme acompañado y asesorado durante la realización de este trabajo de investigación.

Kevin

## TABLA DE CONTENIDO

ÍNDICE DE TABLAS.....	ix
ÍNDICE DE FIGURAS.....	x
ÍNDICE DE ANEXOS .....	xii
RESUMEN.....	xiii
SUMMARY .....	xiv
INTRODUCCIÓN .....	1
CAPÍTULO I.....	6
1. MARCO TEÓRICO REFERENCIAL.....	6
1.1. Ondas electromagnéticas.....	6
1.2. Espectro Radioeléctrico.....	7
1.3. Ondas Milimétricas .....	8
1.4. Efecto Multipath .....	10
1.5. Ambiente <i>Indoor</i> .....	12
1.6. Modelo de propagación en el espacio libre.....	13
1.7. Modelo de propagación Rivera-Zambrano .....	14
1.8. Modelo <i>Ray Tracing</i> .....	14
1.9. Nvidia- OPTIX.....	16
1.10. Unity.....	23
1.10.1. <i>Ciclo de vida Unity</i> .....	26
1.11. Opal.....	28
CAPÍTULO II .....	30
2. MARCO METODOLÓGICO .....	30
2.1. Tipos de investigación .....	30
2.1.1. <i>Investigación Bibliográfica</i> .....	30
2.1.2. <i>Investigación Descriptiva</i> .....	31
2.2. Método de investigación.....	31
2.3. Desarrollo del estudio a través del ciclo de DEMING .....	31



2.3.1.	<i>Fase Planificar</i> .....	32
2.3.2.	<i>Fase Hacer</i> .....	33
2.3.3.	<i>Fase Actuar</i> .....	36
2.3.4.	<i>Fase Verificar</i> .....	38
2.4.	<b>Cálculo de Potencias de Recepción mediante Balance del Enlace</b> .....	40
2.5.	<b>Cálculo de Error Cuadrático Medio</b> .....	41
<b>CAPITULO III</b> .....		<b>43</b>
3.	<b>MARCO DE RESULTADOS, ANÁLISIS Y DISCUSIÓN DE RESULTADOS.</b>	43
3.1.	<b>Escenario 1: Laboratorio de microondas (virtual)</b> .....	43
3.2.	<b>Escenario 2: Corredor de la planta baja (virtual)</b> .....	46
3.3.	<b>Escenario 3: Pruebas reales en laboratorio de microondas</b> .....	49
3.5.	<b>Análisis global de resultados: Laboratorio de Microondas</b> .....	52
3.6.	<b>Análisis global de resultados planta baja FIE</b> .....	54
<b>CONCLUSIONES</b> .....		<b>57</b>
<b>RECOMENDACIONES</b> .....		<b>58</b>
<b>GLOSARIO</b>		
<b>BIBLIOGRAFÍA</b>		
<b>ANEXOS</b>		

## ÍNDICE DE TABLAS

<b>Tabla 1-1:</b>	Bandas de frecuencia del espectro radioeléctrico .....	7
<b>Tabla 1-2:</b>	Descripciones Equipo de cómputo .....	33
<b>Tabla 2-2:</b>	Valores de Ganancia para Tx y Rx .....	39
<b>Tabla 1-3:</b>	Valores PRx (dBm) en Modelo de Propagación Trazado de Rayos Escenario 1 ..	44
<b>Tabla 2-3:</b>	Valores PRx (dBm) en Modelo de Propagación Rivera-Zambrano Escenario 1 ..	45
<b>Tabla 3-3:</b>	Valores PRx (dBm) en Modelo de Propagación Trazado de Rayos Escenario 2 ..	47
<b>Tabla 4-3:</b>	Valores PRx (dBm) en Modelo de Propagación Rivera-Zambrano Escenario 2 ..	48
<b>Tabla 5-3:</b>	Valores PRx (dBm) reales en el Laboratorio de Microondas Escenario 3 .....	50
<b>Tabla 6-3:</b>	Valores PRx (dBm) reales en Planta Baja Escenario 4 .....	51
<b>Tabla 7-3:</b>	Error Cuadrático Medio entre valores reales y simulados (Lab. Microondas) .....	53
<b>Tabla 8-3:</b>	Error Cuadrático Medio entre valores reales y simulados (Planta Baja FIE) .....	55

## ÍNDICE DE FIGURAS

<b>Figura 1-1:</b>	Onda Electromagnética.....	6
<b>Figura 2-1:</b>	Espectro mmWave.....	8
<b>Figura 3-1:</b>	Efecto Multipath.....	10
<b>Figura 4-1:</b>	Reflexión y refracción en la frontera de dos medios distintos.....	12
<b>Figura 5-1:</b>	Esquema trazado de rayos.....	15
<b>Figura 6-1:</b>	Flujo de control de la segmentación de trazado de rayos.....	18
<b>Figura 7-1:</b>	Ejemplo de programa Ray Generation.....	19
<b>Figura 8-1:</b>	Aplicaciones construidas con Optix.....	19
<b>Figura 9-1:</b>	Ejecución de programas en Optix.....	21
<b>Figura 10-1:</b>	Ejecución del Kernel Optix.....	22
<b>Figura 11-1:</b>	Ejecución de eventos en Unity.....	27
<b>Figura 1-2:</b>	Estructura ciclo Deming.....	32
<b>Figura 2-2:</b>	Instalación exitosa de CUDA 10.0.....	33
<b>Figura 3-2:</b>	Instalación exitosa de Cmake.....	34
<b>Figura 4-2:</b>	Instalación exitosa de Optix 6.5.0.....	34
<b>Figura 5-2:</b>	Compilación exitosa de Cmake.....	35
<b>Figura 6-2:</b>	Renderización de Optix con Cmake.....	35
<b>Figura 7-2:</b>	Simulación de la infraestructura de la Facultad de Informática y Electrónica en Unity.....	36
<b>Figura 8-2:</b>	Simulación 1 propagación en línea directa Laboratorio de Microondas.....	37
<b>Figura 9-2:</b>	Simulación 2 propagación en línea directa pasillo primer piso.....	37
<b>Figura 10-2:</b>	Laboratorio de microondas.....	38
<b>Figura 11-2:</b>	Planta baja.....	38
<b>Figura 12-2:</b>	Ganancia Vs Frecuencia para antenas Tx y Rx.....	40
<b>Figura 1-3:</b>	Propagación de rayos Escenario 1.....	43
<b>Figura 2-3:</b>	Canvas Escenario 1.....	44
<b>Figura 3-3:</b>	Diagrama PRx vs Frecuencia en Modelo Trazado de Rayos Escenario 1.....	45
<b>Figura 4-3:</b>	Diagrama PRx vs Frecuencia en Modelo Rivera-Zambrano Escenario 1.....	46
<b>Figura 5-3:</b>	Propagación de Rayos Escenario 2.....	46
<b>Figura 6-3:</b>	Canvas Escenario 2.....	47
<b>Figura 7-3:</b>	Diagrama PRx vs Frecuencia en Modelo Trazado de Rayos Escenario 2.....	48
<b>Figura 8-3:</b>	Diagrama PRx vs Frecuencia en Modelo Rivera-Zambrano Escenario 2.....	49

<b>Figura 9-3:</b>	Mediciones con analizador de espectro Escenario 3 .....	49
<b>Figura 10-3:</b>	Diagrama PRx vs Frecuencia en resultados obtenidos Escenario 3 .....	50
<b>Figura 11-3:</b>	Diagrama PRx vs Frecuencia en resultados obtenidos Escenario 4.....	51
<b>Figura 12-3:</b>	Comparación PRx vs Frecuencia (Distancia más corta) Lab. Microondas.....	52
<b>Figura 13-3:</b>	Comparación PRx vs Frecuencia (Distancia larga) Lab. Microondas .....	53
<b>Figura 14-3:</b>	Comparación PRx vs Frecuencia (Distancia más corta) Planta baja .....	54
<b>Figura 15-3:</b>	Comparación PRx vs Frecuencia (Distancia más larga) Planta baja .....	55

## **ÍNDICE DE ANEXOS**

**ANEXO A:** DLL de configuración Nvidia Optix con Unity

**ANEXO B:** Parámetros configurados en Unity (Laboratorio de Microondas)

**ANEXO C:** Parámetros configurados en Unity (Corredor Planta Baja FIE)

**ANEXO D:** Código Rayo Línea Vista Directa

**ANEXO E:** Código Reflexiones en el Transmisor

**ANEXO F:** Código de cálculo para Potencias de Recepción

## RESUMEN

El objetivo principal del presente trabajo fue el desarrollo de un simulador de propagación 3D *indoor* en frecuencias de 10-20 GHz sobre unidad de procesamiento gráfico (GPU). La metodología de investigación utilizada fue el ciclo de Deming mediante sus cuatro etapas: planificar, hacer, verificar y actuar. Para realizar la simulación se hizo uso de diversas herramientas de Software con versiones específicas como Nvidia CUDA 10.0, CMake 3.19.4, Nvidia OPTIX 6.5, Microsoft Visual Studio enterprise 2017 y Unity 2018.4.21f1. Se definieron dos escenarios con línea de vista (LOS) dentro de la Facultad de Informática y Electrónica (FIE) de la Escuela Superior Politécnica de Chimborazo (ESPOCH), el primero en el Laboratorio de Microondas donde las distancias mínima y máxima entre antenas de transmisión (Tx) y recepción (Rx) es de 0.34m (campo lejano) y 7m; y el segundo escenario se encuentra en el corredor de la planta baja del edificio de la FIE con distancias mínima y máxima de 6 y 12m respectivamente. En dichos escenarios se realizaron mediciones tanto reales como simuladas, con saltos de frecuencia de 1GHz. Al comparar las medidas de potencia recibida simuladas que están basadas en los modelos de propagación Trazado de Rayos y Rivera-Zambrano, se observó que existe una variación que oscila entre los 10 dBm en el escenario 1 (Laboratorio de Microondas) y entre los 12 dBm en el escenario 2 (Planta baja FIE). Se concluye que el modelo Rivera-Zambrano se ajusta más a los valores de potencia reales obtenidos en ambos escenarios. Se recomienda tomar en consideración las limitantes que presenta Nvidia OPTIX en cuanto a la compatibilidad de *Hardware*, ya que limita el desarrollo en equipos con dispositivos de diferentes fabricantes.

**Palabras clave:** <MODELO DE PROPAGACIÓN>, <AMBIENTES DE INTERIORES (INDOOR)>, <TRAZADO DE RAYOS>, <MOTOR DE VIDEOJUEGOS>, <SIMULADOR 3D>, <COMUNICACIONES INALÁMBRICAS>

1606-DBRA-UTP-2022



## SUMMARY

The main objective of this work was the development of an indoor 3D propagation simulator at frequencies of 10-20 GHz on a graphics processing unit (GPU). The research methodology used was the Deming cycle through its four stages: plan, do, verify and act. In order to carry out the simulation, various software tools with specific versions were used, such as Nvidia CUDA 10.0, CMake 3.19.4, Nvidia OPTIX 6.5, Microsoft Visual Studio enterprise 2017 and Unity 2018.4.2 fl. Two scenarios with line of sight (LOS) were defined within the Faculty of Informatics and Electronics (FIE) of Escuela Superior Politécnica de Chimborazo (ESPOCH), the first in the Microwave Laboratory where the minimum and maximum distances between transmission (Tx) and reception (Rx) antennas is 0.34m (far field) and 7m; and the second stage is located in the corridor on the ground floor of the FIE building with minimum and maximum distances of 6 and 12m respectively. In these scenarios, both real and simulated measurements were made, with frequency jumps of 1 GHz. When comparing the simulated received power measurements that are based on the Ray Tracing and Rivera-Zambrano propagation models, it was observed that there is a variation that ranges between 10 dBm in scenario 1 (Microwave Laboratory) and between 12 dBm in scenario 2 (FIE ground floor). It is concluded that the Rivera-Zambrano model is more adjusted to the real power values obtained in both scenarios. It is recommended to take into consideration the limitations that Nvidia OPTIX presents in terms of Hardware compatibility, since it limits development on computers with devices from different manufacturers.

**Keywords:** <PROPAGATION MODEL> <INDOOR ENVIRONMENTS (INDOOR)> <RAY TRACE> <VIDEO GAME ENGINE> <3D SIMULATOR> <WIRELESS COMMUNICATIONS>

## INTRODUCCIÓN

Con el creciente uso de las ondas milimétricas y la aparición de tecnologías 5G, se vuelve cada vez más necesario su estudio a través de modelos de propagación para comprender su comportamiento y funcionamiento. El uso de herramientas de simulación para el análisis de sistemas de comunicaciones no es nuevo; se puede nombrar algunos *Software* como OMNET++ y Veneris que han sido usados en simulaciones de redes inalámbricas que necesitan una simulación de propagación de ondas en 3D consciente del entorno. Sin embargo, dichas herramientas aportan dinamismo y una mejor comprensión al usuario. De igual manera, cabe mencionar la existencia de simuladores (licenciados) de propagación *indoor* como *iBwave* que está hecho para la industria de diseño de redes inalámbricas en interiores.

Así, el presente trabajo pretende realizar el estudio de las tecnologías de virtualización NVIDIA Optix, *Unity* y Opal para llevar a cabo un simulador de propagación 3D *indoor* en la banda de 10-20 GHz en el interior de la FIE y posteriormente realizar mediciones de potencia, variando distancia y frecuencia, de acuerdo con parámetros de entrada y salida definidos. Dichas mediciones tienen lugar en dos escenarios con línea de vista (LOS). Posteriormente, se analiza y compara los valores de potencia obtenidos dentro del simulador, con valores de potencia reales.

## ANTECEDENTES

Estudios realizados en Colombia en 2012 sobre calibración basada en medidas para modelos de trazado de rayos en 3D para ambientes exteriores urbanos andinos, muestran el efecto que produce la optimización de los valores de permitividad para calles, paredes y techos de edificios, con relación a la precisión en la estimación de las pérdidas de propagación en un entorno exterior andino utilizando la técnica de trazado de rayos en 3D. Para obtener un modelo viable del escenario tridimensional, generalmente se toman los valores de las propiedades eléctricas de los materiales desde investigaciones realizadas por otros autores, en donde se caracterizan materiales típicos de otras ciudades, y a menudo la banda de frecuencias de operación no corresponde con la banda de frecuencias utilizadas en la caracterización. Para analizar la dependencia del valor de la pérdida de camino con respecto a la permitividad de los materiales, se estima las pérdidas de propagación para diferentes valores de permitividad en los materiales y se calcula las estadísticas de error con respecto a medidas realizadas en el escenario COST de Cali (Colombia), típico de la región andina. Finalmente, se optimizan los valores de la permitividad y se obtiene un modelo del ambiente tridimensional que mejora el desempeño del trazado de rayos en la estimación de las pérdidas de propagación (Navarro et al., 2012).



Por otra parte, el análisis de problemas de radiación de campos electromagnéticos en entornos complejos encuentra numerosas aplicaciones en el campo de la ingeniería de telecomunicación ya que durante los últimos años se ha incrementado enormemente la demanda de servicios de comunicaciones móviles. Determinar con precisión la propagación de ondas electromagnéticas en ambientes confinados, donde la onda transmitida encuentra obstáculos en su camino antes de llegar al receptor y que pueden producir reforzamiento o cancelación de la señal requiere de la valoración de múltiples fenómenos (Campillo, 2008). En Colombia, mediante el estudio de herramientas de trazado de rayos 3D se buscó el desarrollo de una herramienta de trazado de rayos de tres dimensiones mediante el software MATLAB. Dichos estudios están basados en GO (Óptica Geométrica) y UTD (Teoría de Difracción Uniforme).

En 2015, fue presentada una revisión de los conceptos básicos de rayos, algoritmos de trazado de rayos y propagación de radio utilizando métodos de rastreo de rayos. Se centró en los conceptos fundamentales y en el desarrollo de algoritmos de rastreo de rayos. También se discuten los avances más recientes y una perspectiva futura del trazado de rayos (Zhengqing & Magdy, 2015). En un futuro próximo, se concibe el modelo de propagación como un sistema inteligente y preciso en que el trazado de rayos juega un papel importante.

En Riobamba (2021), por parte de estudiantes de la Facultad de Informática y Electrónica de la ESPOCH, se determinó un modelo de propagación empírico para entornos *indoor* en la banda de la 10-20 GHz. Para lo cual se utilizó dos antenas tipo corneta, un analizador de redes vectorial, un generador de señales y un analizador de espectros, y se caracterizó las antenas con el uso del analizador de redes vectorial a una distancia de 0.34 metros (campo lejano) y variaciones de frecuencia de 100 MHz. Las mediciones se realizaron por medio de la utilización de las antenas tipo corneta conectadas al generador de señales y al analizador de espectros respectivamente con variaciones de distancia de 1m, estas mediciones fueron efectuadas en la Facultad de Informática y Electrónica dentro del Laboratorio de Microondas, entorno con línea de vista (LOS) siendo la distancia máxima de separación entre Transmisor y Receptor de 7m, posteriormente desde un aula A a un aula B, entorno sin línea de vista (NLOS) a una distancia máxima de 6m y considerando una pared entre ellas, para lo cual se determinaron las pérdidas de inserción en el material introduciendo un pedazo de pared dentro de la cámara anecoica, obteniendo mayores pérdidas a mayor frecuencia por tal motivo se dedujo una ecuación general que describa este comportamiento (Rivera & Zambrano, 2021).

## **FORMULACIÓN DEL PROBLEMA**

¿Cómo desarrollar un simulador de propagación 3D *indoor* en frecuencias de 10-20 GHz sobre GPU con NVIDIA-Optix bajo el motor gráfico Unity para la FIE?

¿Cómo funcionan las tecnologías de virtualización NVIDIA-Optix, Unity y OPAL?

¿Cómo desarrollar un simulador de propagación *indoor* multicamino en 3D para validar un modelo empírico basado en el modelo “Free Space”, mediante el motor gráfico Unity?

¿Qué parámetros de entrada y salida son necesarios para la configuración de la simulación, que será utilizada por el usuario?

¿Cómo se realizarán las simulaciones al interior del edificio principal de la FIE, de propagación multicamino para escenarios con LOS en el rango de frecuencias de 10-20 GHz?

¿Cómo se realizarán las mediciones de potencia de propagación reales al interior del edificio principal de la FIE para los escenarios LOS simulados en el rango de frecuencias de 10-20 GHz?

¿Cómo evaluar el desempeño del simulador en los escenarios LOS en el rango de frecuencias de 10-20 GHz?

## **JUSTIFICACIÓN**

### **Justificación Teórica**

En redes de comunicaciones inalámbricas la estimación a gran escala de las pérdidas de propagación resulta clave en el modelado de los sistemas de comunicación en función de la distancia y la frecuencia. Existen diferentes tipos de modelos como son determinísticos, estocásticos o empíricos. Para realizar análisis y simulaciones a nivel de enlace y de sistema en nuevas bandas, se debe conocer que un modelo puede ser utilizado en un nuevo escenario, con entornos, distancias y frecuencias diferentes, mostrando estabilidad, precisión y utilidad más allá de las mediciones tomadas in-situ y, a partir de las cuales se desarrolla (Durán, 2017), por lo que es necesario también identificar cual es el modelo de propagación adecuado.

Los simuladores son herramientas que nos permiten realizar una infinidad de tareas, como el modelado de objetos en tiempo real de forma interactiva, simulaciones y virtualizaciones en 3D tanto de espacios outdoor como *indoor*, representación de situaciones asemejándolas a la realidad, etc. Es el caso así que, en el mercado se encuentran tecnologías como Veneris, SUMO, OMNet++, entre otras. Estas tecnologías han sido mayormente utilizadas en el marco de simulación para la investigación de redes vehiculares y conducción automatizada cooperativa, aunque también se

pueden utilizar para la simulación de redes inalámbricas generales que necesitan una simulación de propagación en 3D consciente del entorno.

Con la aparición de las 5G y el creciente uso de las ondas milimétricas, se ha vuelto necesario realizar estudios de los modelos de propagación y de canal para comprender mejor su funcionamiento; realizar este tipo de simulaciones no es una tarea fácil puesto a las limitaciones generales que presenta, pero al tener al alcance softwares de licencia libre e interfaz amigable con el usuario como el motor de multi- plataforma Unity.

A medida que crece la demanda de datos móviles, el espectro sub-3 GHz se está volviendo cada vez más concurrido. Por otro lado, una gran cantidad de espectro en el rango de 3–300 GHz permanece subutilizada. El espectro de 3–30 GHz generalmente se conoce como la banda de frecuencia súper alta (SHF), mientras que 30–300 GHz se conoce como la banda de frecuencia extremadamente alta (EHF) o de onda milimétrica. Como las ondas de radio en las bandas SHF y EHF comparten características de propagación similares, se refiere al espectro de 3–300 GHz colectivamente como bandas de ondas milimétricas con longitudes de onda que varían de 1 a 100 mm (Rivera & Zambrano, 2021).

### **Justificación Aplicativa**

Para poder mostrar de manera visual, mediante simulación, cómo se transmite y recibe la potencia de una onda electromagnética, los cambios que sufre la misma a causa del efecto multipath, y además para validar un modelo de propagación empírico basado en el modelo “Free Space”, se desarrollará un simulador de propagación 3D *indoor* en la banda de 10-20 GHz sobre GPU con NVIDIA-Optix bajo el motor gráfico Unity para el edificio principal de la FIE. Se definirán dos escenarios con LOS (line-of-sight, línea de vista), uno dentro de una habitación del edificio y otro en un ambiente más abierto como los pasillos, para realizar mediciones de potencia de las ondas electromagnéticas tanto simuladas como reales variando la frecuencia y de esta manera, mediante los resultados obtenidos evaluar el desempeño del simulador a desarrollar.

Esto puede ser utilizada como una herramienta educativa que ayude a estudiantes a ver en el simulador cómo se da la propagación multicamino de manera gráfica e interactiva.

## OBJETIVOS

### OBJETIVO GENERAL

Desarrollar un simulador de propagación 3D *indoor* en la banda de 10-20 GHz sobre GPU con NVIDIA Optix bajo el motor gráfico para la FIE.

### OBJETIVOS ESPECÍFICOS

- Estudiar las tecnologías de virtualización NVIDIA Optix, Unity y OPAL.
- Definir los parámetros necesarios de entrada y salida para la configuración de la simulación que será utilizado por el usuario.
- Desarrollar un simulador de propagación *indoor* multicamino en 3D para validar un modelo empírico basado en el modelo “*Free Space*”, mediante el motor gráfico *Unity*.
- Realizar simulaciones al interior del Edificio principal de la FIE, de propagación multicamino para escenarios con LOS (línea de vista) en el rango de frecuencias de 10-20 GHz.
- Realizar mediciones de potencia de propagación reales al interior del Edificio principal de la FIE para los escenarios LOS simulados en el rango de frecuencias de 10-20 GHz.
- Evaluar el desempeño del simulador mediante pruebas de propagación simuladas vs reales en los escenarios LOS en el rango de frecuencias de 10-20 GHz.

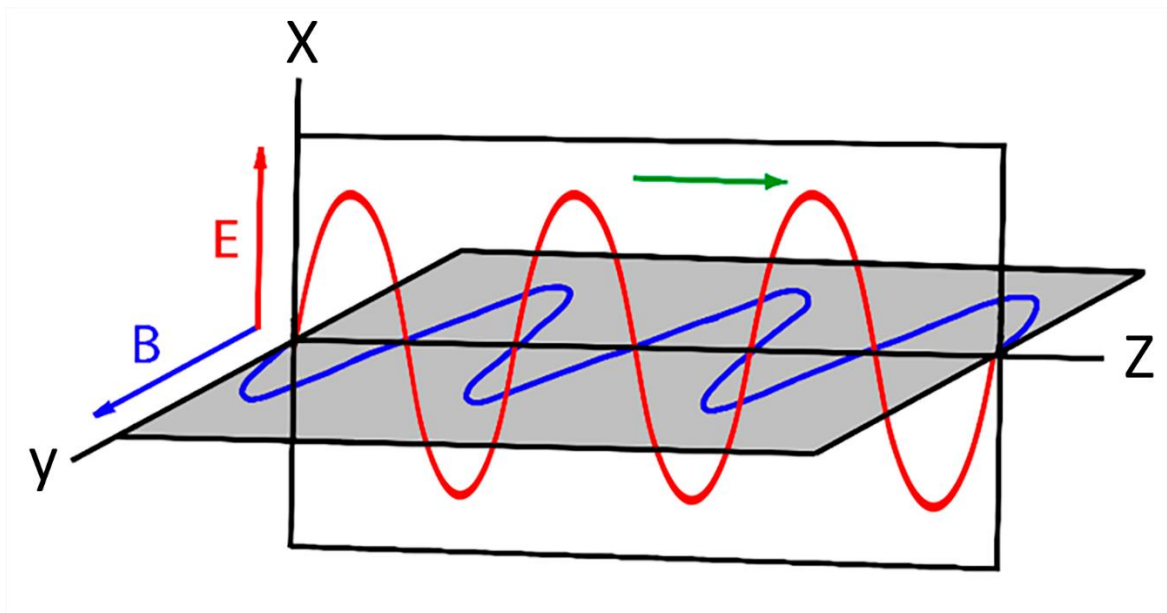
# CAPÍTULO I

## 1. MARCO TEÓRICO REFERENCIAL

En el capítulo presente se tratan diversos conceptos teóricos referentes a la realización de este trabajo de investigación tales como efecto *multipath*, ambiente *indoor*, modelos de propagación LOS y NLOS; así como del software Nvidia-OPTIX y Unity, necesarios para el desarrollo del simulador.

### 1.1. Ondas electromagnéticas

Son ondas que se generan debido a la variación en el tiempo de campos eléctricos y magnéticos que se propagan a la velocidad de la luz a través del espacio transportando energía de un lugar a otro. Estas ondas dan lugar a la denominada radiación electromagnética. Para una onda plana, el campo eléctrico y el campo magnético son transversales entre sí y estos a su vez son transversales con la dirección de propagación de la onda, véase Figura 1-1. Se basan en el hecho de que toda carga eléctrica al estar acelerada debido a la fuerza eléctrica que experimenta genera radiación electromagnética como energía (Castaño, 2008, p.7).



**Figura 1-1:** Onda Electromagnética

**Realizado por:** Freire Kevin, 2022

## 1.2. Espectro Radioeléctrico

El espectro de radio se refiere a la parte del espectro electromagnético que se utiliza para comunicar información. Las aplicaciones importantes para la sociedad, como la radiodifusión y la televisión, la aviación civil, los satélites, los servicios de defensa y de emergencia, dependen de asignaciones específicas de radiofrecuencia. El espectro radioeléctrico está conformado por diversas bandas, las cuales son utilizadas para la prestación de servicios de comunicación inalámbricas por parte de un licenciatario. Con el pasar del tiempo la demanda del espectro ha aumentado drásticamente, impulsada por cantidades crecientes de datos transmitidos a través de Internet y un número cada vez mayor de dispositivos inalámbricos, incluidos teléfonos inteligentes y tabletas, redes *Wi-Fi* y objetos cotidianos conectados a Internet (Davies, 2015, p.1).

**Tabla 1-1:** Bandas de frecuencia del espectro radioeléctrico

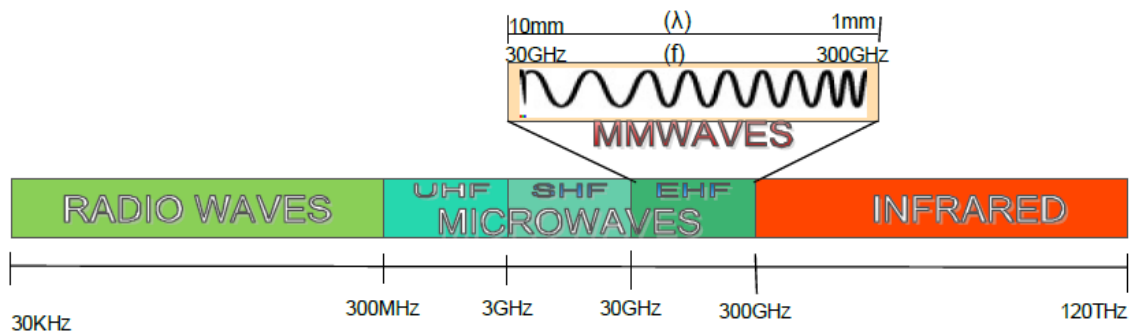
Banda	Frecuencia	Long. De onda	Nomenclatura	Uso
VLF	3-30 KHz	10-100 Km	Miriamétricas	-Radionavegación -Servicio Móvil Marítimo
LF	30-300 KHz	1-10 Km	Kilométricas	-Frecuencias Patrón
MF	300-3000 KHz	100 m – 1 Km	Hectométricas	-Radiodifusión Sonora en AM
HF	3-30 MHz	10-100 m	Decamétricas	-Telefonía Fija y Móvil -Radioaficionados -Radio en Onda Corta
VHF	30-300 MHz	1-10 m	Métricas	-Telefonía Fija y Móvil -Radioaficionados -Radiodifusión Sonora en FM -Televisión Abierta -Radionavegación
UHF	300-3000 MHz	100 mm – 1 m	Decimétricas	-Telefonía Fija y Móvil -Televisión Abierta -Radiolocalización
SHF	3-30 GHz	10-100 mm	Centimétricas	-Telefonía Fija y Móvil -Radiodifusión por Satélite -Radionavegación
EHF	30-300 GHz	1-10 mm	Milimétricas	-Telefonía Fija

Fuente: (Espinosa, 2021, p. 1)

El espectro radioeléctrico es un recurso natural finito que debe gestionarse para obtener los máximos beneficios económicos y sociales. Los países han regulado tradicionalmente el espectro de radio dentro de sus territorios. El rango del espectro radioeléctrico, atendiendo las características de propagación, ha sido dividido en bandas y sub-bandas de frecuencias, tal como se indica en la Tabla 1-1.

### 1.3. Ondas Milimétricas

Las denominadas ondas milimétricas (*MMW*) son ondas de radio con longitudes de onda del orden de los milímetros, y por ende de frecuencias altas. En la Figura 2-1 se observa que la porción del espectro que corresponde a este tipo de ondas se encuentra en la gama de longitudes de onda de 10 mm a 1 mm, son conocidas como de frecuencia extremadamente alta (*EHF*, *Extremely High Frequency*), y pertenecen a las microondas. Según la Unión Internacional de Telecomunicaciones UIT y el Sector de Radiocomunicaciones de la UIT denominado UIT-R, el rango de frecuencia de éstas, se halla desde los 30 GHz hasta los 300 GHz y es la banda más alta en cuanto a uso práctico se refiere (UIT-R, 2016, p. 2; Madroñero & Meneses, 2016, p. 4).



**Figura 2-1:** Espectro *mmWave*

**Fuente:** (Madroñero y Meneses, 2016, p. 4)

El gran interés de estudio y aplicación de las ondas milimétricas radica principalmente en las ventajas que poseen éstas, ya que se pueden obtener mayores tasas de transferencia debido al gran ancho de banda disponible. Además, debido a que el tamaño de los dispositivos transmisores es inversamente proporcional a la frecuencia de las ondas, se considera una reducción en las antenas, pero estas ventajas van de la mano con las limitaciones de las *mmW* (ondas milimétricas), ya que se necesita elevados costos de manufactura en hardware de precisión. De igual manera una limitante es la distancia de transmisión ya que la atenuación es más significativa en estos elevados

rangos de frecuencia (Vásquez & Gaspar, 2017, pp. 14- 15). Se conoce que, el espectro comprendido entre los 3 a 300 GHz aún permanece subutilizado a diferencia de las frecuencias por debajo de este rango cuyo uso ha sufrido un incremento exponencial con el pasar de los años. Dentro de la banda mencionada de 3-30 GHz se encuentran las frecuencias SHF (*Super High Frequency*) y EHF (*Extremely High Frequency*), éstas últimas conocidas como ondas milimétricas; pero debido a las características similares que comparten las ondas SHF con las EHF, en el presente trabajo se tratarán colectivamente como banda de ondas milimétricas con longitudes de onda que varían entre 1-10 mm (Rivera & Zambrano, 2021, pp. 5- 6).

### **New Radio (NR) 5G (Quinta Generación de Comunicaciones móviles)**

Es la última generación en cuanto a tecnologías inalámbricas se refiere, la cual ofrece velocidades de hasta 10 veces más rápidas que su antecesor (4G). En esta tecnología se empiezan a utilizar equipos de propósito general dejando a un lado el *hardware* específico de telefonía, lo cual a nivel de costes y de implementación representa una ventaja. Además, permite que la infraestructura sea accesible por equipos en internet e interoperable (AEPD, 2020, p. 5). Dentro de las mejoras que posee 5G en comparación con tecnologías anteriores se tiene (Ramírez & Colmenares, 2020, pp. 4- 5):

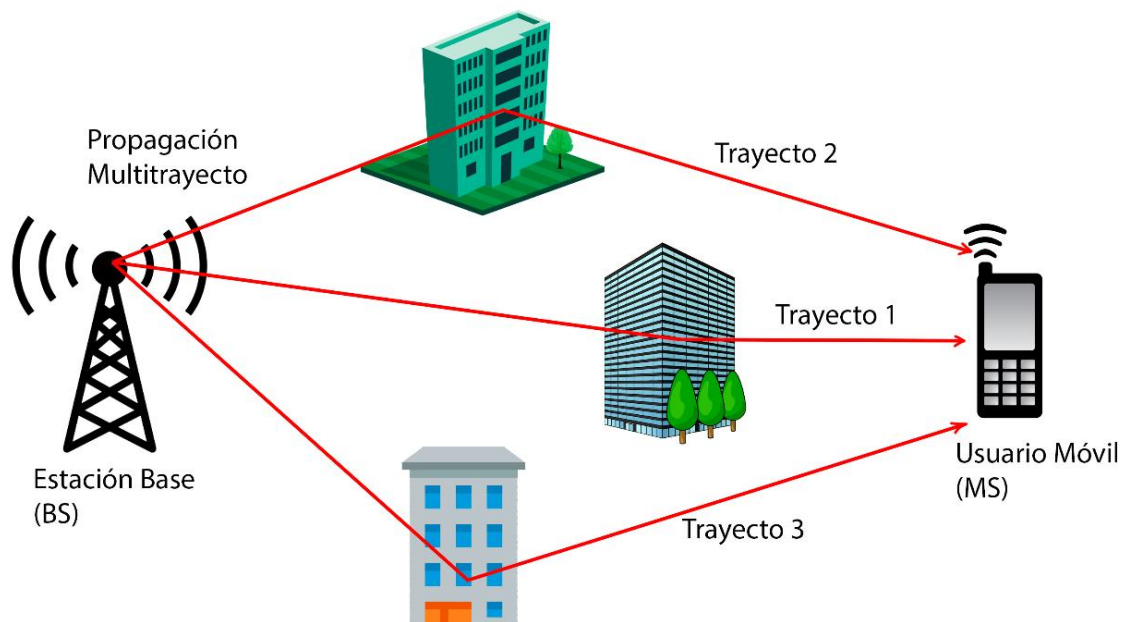
- **Mayor velocidad de la red:** Uno de los aspectos más importantes de las nuevas tecnologías inalámbricas que surgen es la velocidad de los datos. Las redes 4G alcanzan velocidades de hasta 1Gb/s ante lo cual 5G se sobrepone llegando a los 10Gb/s.
- **Mejora en el ancho de banda:** La quinta generación de tecnologías móviles posee un aumento significativo en cuanto a términos de ancho de banda, con lo cual el volumen de los datos y la capacidad de la red en sí mejoran considerablemente.
- **Baja latencia:** Las redes que utilizan tecnología 4G poseen un tiempo de latencia de alrededor de 20 milisegundos, el cual gracias a la llegada de 5G se reduce a tiempos de 1 o 2 milisegundos, es decir existe una mejora notable en el tiempo que tardan los datos en llegar desde el emisor al receptor.
- **Más dispositivos conectados:** Ya que la tecnología 5G hacen posible la administración de volúmenes de datos mucho mayores, es posible conectar alrededor de 1 millón de dispositivos por kilómetro cuadrado, en comparación con los 100 mil dispositivos que se pueden conectar por kilómetro cuadrado mediante 4G. Este incremento se debe a que las antenas 5G están basadas en tecnología MIMO.



#### 1.4. Efecto Multipath

El canal radio está caracterizado por una recepción multicamino, es decir, que, durante una transmisión, la señal que llega al receptor no posee únicamente una componente de línea de vista directa, al contrario, también está conformada de una gran cantidad de ondas de radio reflejadas. Naturalmente, esto resulta mucho peor en ambientes urbanos ya que debido a la gran cantidad de obstáculos presentes en el ambiente, no suele existir línea de visión directa y lo que llega al receptor es una combinación de las ondas reflejadas con diferentes retardos, como se puede apreciar en la Figura 3-1, y estas interfieren con la onda directa desembocando en una degradación del enlace. Si la antena receptora está en movimiento, el canal radio varía con la distancia y el tiempo, produciéndose desvanecimiento. De vez en cuando en estos casos la señal recibida es muy débil y se dan muchos errores de bits incluso si la relación señal/ruido es grande. Por otro lado, si se habla de un receptor fijo sin desvanecimiento, la tasa de error de bits (BER) se reduce considerablemente a medida que aumenta la relación S/N (Wirelesscommunication., 2001, p. 1).

Los sistemas inalámbricos deben ser diseñados de forma que se minimicen los efectos desfavorables producidos por el efecto *multipath*.



**Figura 3-1:** Efecto Multipath

Realizado por: Freire Kevin, 2022

## **Difracción**

Se refiere al hecho de propagar una onda electromagnética a través y alrededor de algún objeto que represente un obstáculo, obstrucción o un cambio de medio para la señal transmitida. El análisis de sus efectos es muy importante cuando no se tiene una línea de visión directa, éstos, dentro de las frecuencias milimétricas son despreciables a diferencia de los causados por reflexión que son los efectos que se producen principalmente durante la propagación.

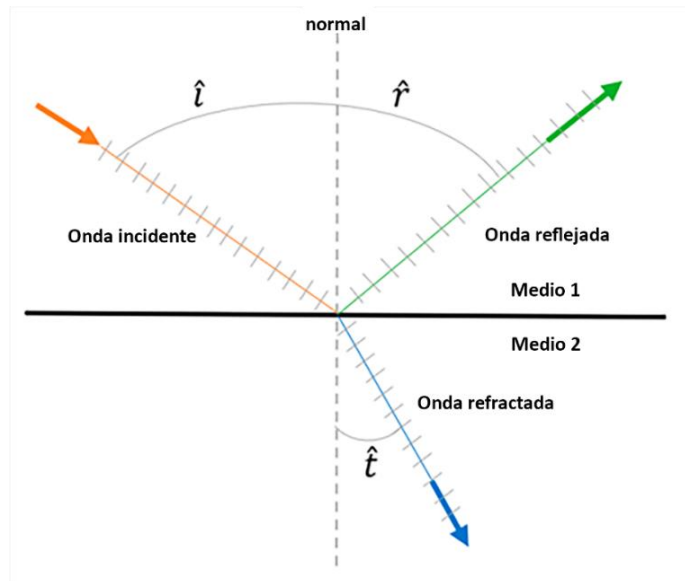
Las ondas reflejadas resultantes de la incidencia con un obstáculo están presentes a través del espacio, inclusive detrás del objeto, lo que resulta en una inflexión de ondas alrededor de éste, de igual manera como si no existiese una línea de visión directa. En altas frecuencias, esta difracción depende de la geometría del obstáculo, así como de la amplitud, fase y polarización que posea la onda incidente en el punto de difracción; y genera pérdidas en la potencia transmitida, donde la onda difractada tiene menor potencia que el frente de onda que la origina (Vaca, 2017, p. 19).

Este tipo de comportamientos puede ser comprendido de mejor manera mediante el principio de Huygens, en el cual cada punto del obstáculo genera un nuevo frente de ondas que puede inclusive hasta rodear un obstáculo. Mientras más altas sean las frecuencias transmitidas mayores serán las pérdidas por difracción, pero se difractarán menos.

## **Reflexión y Transmisión**

El fenómeno de la reflexión ocurre cuando una onda incide contra un obstáculo y parte de la potencia transmitida que golpea al objeto, es reflejada. Las ondas reflejadas permanecen en el mismo medio que las ondas incidentes, razón por lo cual sus velocidades son las mismas y el ángulo de incidencia es igual al ángulo de reflexión. Se deben tomar en cuenta tanto propiedades de la señal: longitud de onda, orientación, ángulo de incidencia, etc., como propiedades físicas del obstáculo: textura, grosor, geometría de su superficie y las propiedades dieléctricas del material del que está compuesto. Como fue comprobado, los materiales que son conductores perfectos reflejarán la onda en su totalidad, y otros materiales transmitirán una parte y reflejarán lo demás (Campos, 2004, pp.4- 5).

Es decir, si una onda que se propaga incide en la frontera de dos medios cada uno con parámetros de permitividad, permeabilidad y conductividad ( $\epsilon$ ,  $\mu$ , y  $\sigma$ ) respectivamente, como se muestra en la Figura 4-1, una parte de la potencia se refleja y la otra parte se transmite al otro medio.



**Figura 4-1:** Reflexión y refracción en la frontera de dos medios distintos

Fuente: (Rivera & Zambrano, 2021, p. 11)

## Dispersión

La dispersión se da cuando la onda electromagnética viaja por un medio que está conformado por objetos con dimensiones pequeñas en comparación de la longitud de la onda, es decir existe un gran número de obstáculos por unidad de volumen. Si es el caso de que haya varios objetos en el trayecto de la señal transmitida, la energía y el frente de onda se dividirán en varias direcciones, sumándose a las demás interferencias de la señal, destructivas y constructivas (Cerezuela, 2016, pp.11-12).

### 1.5. Ambiente *Indoor*

Se define un ambiente *indoor* a uno de dimensiones pequeñas en el cual generalmente no se toma en cuenta los efectos de atenuación por el clima o factores externos, destinado de manera usual para edificios, oficinas, hogares, etc.

En este tipo de ambientes, más importante que la distancia entre las antenas, va a ser cómo esté dispuesto el entorno donde se realizará la propagación. Se deben tomar en cuenta el número y composición de paredes y suelos, el mobiliario, el diseño del lugar, de la misma forma uno de los parámetros más importantes en la propagación será la directividad de las antenas. En ambientes *indoor* los procesos de *scattering* y difracción suelen tener mayor relevancia que en ambientes exteriores, siendo las principales causas de éstos las estructuras de metal y el mobiliario (Ballester,

2013, p. 13). La propagación en ambientes de interior varía según si en ésta existe línea de vista directa (LOS) entre la antena transmisora y el usuario terminal o no (NLOS).

### 1.6. Modelo de propagación en el espacio libre

El término de propagación en espacio libre hace referencia cuando entre ambos terminales (Tx y Rx) no hay obstáculo alguno, es decir existe línea de vista directa (LOS), siendo la distancia el principal factor que interviene en la atenuación del enlace junto con otros aspectos como la frecuencia, el diseño y tipo de las antenas utilizadas, el patrón de radiación, etc.

El modelo de propagación en espacio libre obedece a la Ecuación de Friss, misma que permite hacer una estimación de la potencia que llega a un receptor situada a cierta distancia  $d$  del transmisor (Castellanos et al., 2005, p. 132), como se muestra en la Ecuación 1-1 a continuación:

$$P_r(d) = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d^2 L}$$

#### **Ecuación 1-1:** Ecuación de Friss

Donde  $P_r(d)$  será la potencia recibida en el receptor que está en función de la distancia existente entre Tx y Rx,  $G_t$  y  $G_r$  corresponden a las ganancias de las antenas transmisora y receptora respectivamente,  $P_t$  es la potencia transmitida, y  $L$  son las pérdidas en el sistema debido a filtros, líneas, etc. En la ecuación se puede notar que, la potencia recibida es inversamente proporcional con el cuadrado de la distancia entre Tx y Rx, es decir que  $P_r$  disminuye con la distancia a razón de 20dB/década.

Las pérdidas de propagación en espacio libre para un enlace punto a punto, se calculan mediante la Ecuación 2-1 (Madroñero & Meneses, 2016, p. 19) detallada en la parte inferior:

$$l_{bf}(dB) = 32,45 + 20 \log f(MHz) + 20 \log d(Km)$$

#### **Ecuación 2-1:** Pérdidas de propagación en espacio libre

Donde:

$l_{bf}$ : pérdidas de propagación en espacio libre

$d$ : distancia

$f$ : frecuencia

### 1.7. Modelo de propagación Rivera-Zambrano

El modelo Rivera-Zambrano es un modelo empírico que predice el entorno *indoor* en la banda de 10-20 GHz con un error cuadrático medio de 4.48. Fue desarrollado a partir de antenas tipo corneta con una apertura de 5cm, caracterizadas mediante un analizador de redes vectoriales a distancia de *far field* (0,34 metros). La toma de mediciones fue efectuada dentro del edificio de la Facultad de Informática y Electrónica para lo que se definió dos escenarios, uno con línea de vista directa (LOS) ubicado dentro del Laboratorio de Microondas y distancia máxima de 7 metros entre antena transmisora (Tx) y antena receptora (Rx), y otro sin línea de vista (NLOS) establecido entre las aulas 302 y 303 del edificio con distancia máxima de 6 metros y considerado una pared entre ellas, para lo cual se determinó las pérdidas de inserción en material introduciendo un pedazo de pared de 40x40x20 cm dentro de una cámara anecoica. Las mediciones fueron realizadas con variaciones de distancia de 1 metro y de frecuencia de 100 MHz.

La expresión que muestra las pérdidas de propagación en el modelo Rivera-Zambrano está dada por la Ecuación 3-1:

$$L = 55,8904 + 20,3121 \log(d) + 4,6757 \log(f)$$

**Ecuación 3-1:** Pérdidas de propagación modelo Rivera-Zambrano

Donde:

$L$ : pérdidas de propagación

$d$ : distancia

$f$ : frecuencia

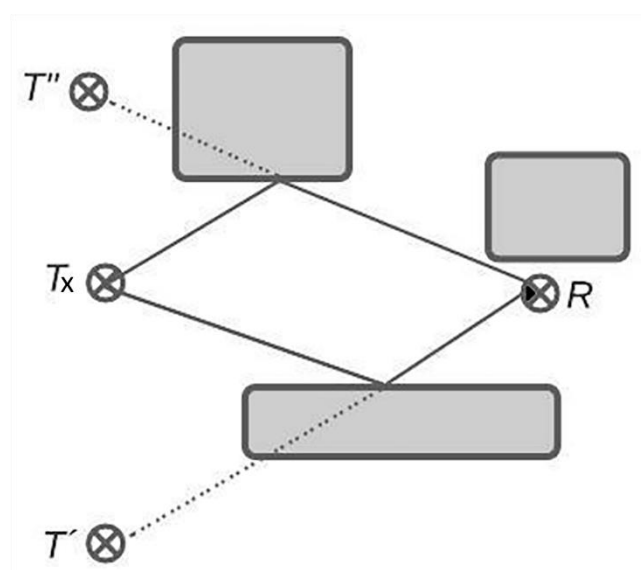
### 1.8. Modelo Ray Tracing

El modelo de trazado de rayos es un modelo basado en imágenes que asume que todos los objetos en los entornos de propagación son reflectores potenciales. En las implementaciones, el modelo de trazado de rayos utiliza las imágenes del transmisor en relación con todos los reflectores, es decir, todos los objetos en los entornos de propagación para determinar las direcciones de los rayos reflejados. A diferencia de otros modelos basados en rayos ópticos como el de lanzamiento de rayos, el modelo de trazado de rayos tiene algunas ventajas. Para el lanzamiento de rayos, se tiene que lanzar una gran cantidad de rayos desde el transmisor, pero para un receptor específico, solo unos pocos pueden llegar finalmente al receptor y la mayoría de estos rayos serían un desperdicio. Por lo tanto, para la predicción punto a punto, el lanzamiento de rayos no es eficiente. Sin embargo, al contrario, el trazado de rayos considera solo las rutas que realmente existen entre el

transmisor y el receptor. Por lo tanto, el trazado de rayos a menudo requiere mucho menos tiempo de cálculo que el lanzamiento de rayos. Pero el trazado de rayos tiene la desventaja de que su tiempo de cálculo crece exponencialmente con el orden de las reflexiones calculadas (Meiling, 2013, p. 21).

Tanto en los modelos de lanzamiento de rayos como en los de trazado de rayos, la fuerza de los rayos reflejados y los rayos refractados se calculan de acuerdo con la óptica geométrica. Los rayos difractados se calculan de acuerdo, por ejemplo, con Teoría UTD. Para el lanzamiento de rayos y el trazado de rayos, la complejidad de los entornos de propagación tiene un fuerte impacto en su carga computacional ya que más obstáculos conducen a más reflejos y difracciones, etc. Mediante este método, cada uno de los píxeles (R) que reciben potencia son computados individualmente. Se determinan imágenes del transmisor para poder determinar los rayos reflejados, ósea la imagen relativa del Tx al plano reflectante ( $T'$  o  $T''$ ) como se observa en la Figura 9-1. Ya que cada uno de los píxeles es calculado independientemente el tiempo de cómputo es elevado pero los resultados que se obtienen con este método son muy exactos por el mismo hecho de que todos los objetos relevantes en la propagación se toman en cuenta.

En escenarios de interiores el canal móvil se caracteriza por el efecto *multipath*. Los fenómenos que dominan en la propagación en el interior del edificio son el sombreado sobre paredes, difracción sobre cuñas verticales y efectos de guiado de ondas en pasillos a causa de las múltiples reflexiones. El modelo *Ray Tracing* calcula todas las posibles trayectorias desde el transmisor hasta el receptor. Al final, el nivel de la señal en una situación específica se obtiene como la suma de componentes de todas las trayectorias entre el transmisor y el receptor.



**Figura 5-1:** Esquema trazado de rayos

Realizado por: Freire Kevin, 2022

En condiciones donde se tome en cuenta los efectos producidos por reflexiones, transmisiones y difracciones, la potencia de recepción dentro del modelo de Trazado de Rayos está predicha por la Ecuación 4-1:

$$Pr = \frac{\lambda^2}{4\pi n_0} \left| \sum_{n=0}^M \frac{E_0 e^{-jks_n} \sqrt{G_{rn} G'_{tn}}}{P_n} \tilde{a}_{tn} \left( \prod_{i=0}^{a_n} R_{in} A_{rin} \right) \left( \prod_{j=0}^{b_n} T_{jn} A_{tjn} \right) \left( \prod_{m=0}^{c_n} D_{mn} A_{dmn} \right) \tilde{a}_{rn} \right|^2$$

**Ecuación 4-1:** Potencia Recibida Modelo Trazado de Rayos

Donde:

$\lambda$  = longitud de onda

$n_0$  = impedancia intrínseca del medio

$M$  = número de rayos

$E_0$  = intensidad campo eléctrico a 1 metro de Tx en dirección de la ganancia más alta =  $\sqrt{\frac{n_0}{4\pi}} P_t G_t$

$P_t$  = potencia de transmisión

$k$  = constante de propagación

$s_n$  = distancia acumulativa de cada rayo

$G_{rn}$  = ganancia de Rx en el ángulo de recepción

$G'_{tn}$  = ganancia de Tx normalizada

$P_n$  = distancia de Tx al primer punto de reflexión

$\tilde{a}_{tn}, \tilde{a}_{rn}$  = polarizaciones de antena Tx y Rx

$a_n, b_n, c_n$  = número de reflexiones, transmisiones y difracciones

$R_{in}, T_{jn}, D_{mn}$  = coeficientes de reflexión, transmisión y difracción

$A_{rin}, A_{tjn}, A_{dmn}$  = factores de *Spreading* asociados

En los modelos básicos de *Ray Tracing*, la predicción está basada en el cálculo de la transmisión en espacio libre (*free space*) y en las reflexiones de las distintas paredes. En el presente trabajo se considera un ambiente sin pérdidas de reflexión y para los cálculos en torno al modelo de trazado de rayos se utilizó la Ecuación 2-1 de pérdidas de propagación en espacio libre.

## 1.9. Nvidia- OPTIX

La multinacional Nvidia Corporation es una empresa centrada en el desarrollo de unidades de procesamiento gráfico (GPUs), tecnologías de circuitos integrados y chips que son usados especialmente en tarjetas madre de computadoras y consolas de videojuegos.

El motor NVIDIA-OptiX es un sistema programable diseñado para GPU NVIDIA y otras arquitecturas altamente paralelas. OptiX es un software intermedio para el trazado de rayos, que combina la implementación de la GPU y los algoritmos inteligentes mientras oculta muchas de las optimizaciones y decisiones de ejecución del usuario, como las optimizaciones específicas de la GPU y la gestión de la recursividad de pila. OptiX se ha utilizado en una amplia variedad de aplicaciones, como renderizado de gráficos, detección de colisiones, estimación de volumen y simulaciones científicas como propagación de sonido. Se puede pensar que OptiX tiene tanto un componente de *framework* como un componente de motor. El *framework* especifica las reglas para configurar la escena, establece la arquitectura de los programas editables por el usuario y proporciona funciones y tipos de datos. El motor impulsa los eventos de simulación, incluida la invocación de programas y la ejecución de comportamientos de simulación (Williams, 2013, p. 18).

Los algoritmos de gráficos por computadora para la representación, o síntesis de imágenes, adoptan uno de dos enfoques complementarios. Una familia de algoritmos recorre los píxeles de la imagen, calculando para cada píxel, el primer objeto visible en ese píxel; este enfoque se llama trazado de rayos porque resuelve el problema geométrico de la intersección de un rayo desde el píxel hacia los objetos. Una segunda familia de algoritmos recorre los objetos en la escena, calculando para cada objeto los píxeles cubiertos por ese objeto. Debido a que los píxeles por objeto resultantes (llamados fragmentos) están formateados para una visualización ráster, este enfoque se denomina rasterización. La estructura de datos central del trazado de rayos es un índice espacial llamado estructura de aceleración, que se utiliza para evitar probar cada rayo con todos los objetos.

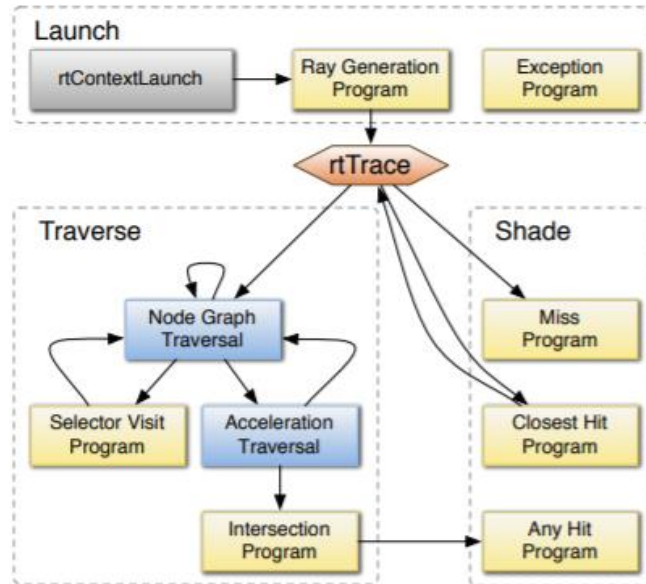
La estructura de datos central de la rasterización es el búfer de profundidad, que almacena la distancia del objeto más cercano visto en cada píxel y descarta fragmentos de objetos invisibles. Si bien ambos enfoques se han generalizado y optimizado mucho más allá de esta descripción simplista, la distinción básica permanece: el trazado de rayos itera sobre los rayos mientras que la rasterización itera sobre los objetos. El trazado de rayos y la rasterización de alto rendimiento, ambos se enfocan en renderizar los objetos más simples: triángulos (Parker et al., 2013, p. 93).

### **OPTIX como Framework**

OptiX incluye ocho tipos diferentes de programas, cada uno de los cuales opera conceptualmente en un solo rayo a la vez. Además, un programa de cuadro delimitador opera sobre la geometría para determinar los límites primitivos para la construcción de estructuras de aceleración. La combinación de programas de usuario y código de kernel OptiX codificado forma la segmentación de trazado de rayos, que se describe en la Figura 6-1. A diferencia de una segmentación de



rasterización *feed-forward*, es más natural pensar en la segmentación de trazado de rayos como un gráfico de flujo de control. La operación principal, *rtTrace*, alterna entre localizar una intersección (*Traverse*) y responder a esa intersección (*Shade*). Al leer y escribir datos en cargas útiles de rayos definidas por el usuario y en matrices de memoria de dispositivos globales llamadas búferes, estas operaciones se combinan para realizar cálculos arbitrarios durante el trazado de rayos.



**Figura 6-1:** Flujo de control de la segmentación de trazado de rayos

Fuente : (Parker et al., 2010, p. 3)

En la Figura 6-1, los cuadros amarillos representan programas especificados por el usuario y los cuadros azules son algoritmos internos de OptiX. La ejecución es iniciada por la llamada API *rtContextLaunch*. El programa de generación de rayos puede emplear una función incorporada, *rtTrace*, para proyectar rayos en la escena. Esta función también puede ser invocada de forma recursiva por el programa de impacto más cercano para sombras y rayos secundarios. El programa de excepción se ejecuta cuando la ejecución de un rayo en particular termina por un error, como un consumo excesivo de memoria.

Los programas de *Ray Generation* son la entrada a la tubería del trazado de rayos. Una sola invocación de *rtContextLaunch* creará muchas instancias de estos programas. En el ejemplo de la Figura 7-1, un programa de generación de rayos creará un rayo utilizando un modelo de cámara estenopeica para un solo píxel, iniciará una operación de rastreo y almacenará el color resultante en un búfer de salida. Con este mecanismo, también se pueden realizar otras operaciones, como crear mapas de fotones, computar iluminación horneada, procesar solicitudes de rayos pasados desde OpenGL, disparar múltiples rayos para supermuestreo o implementar diferentes modelos de cámara.

---

```

RT_PROGRAM void pinhole_camera() {
    Ray ray = PinholeCamera::makeRay( launchIndex );
    UserPayload payload;
    rtTrace( topObject, ray, payload );
    outputBuffer[launchIndex] = payload.result;
}

```

---

**Figura 7-1:** Ejemplo de programa Ray Generation

**Fuente :** (Parker et al., 2010, p. 3)

Los programas *Intersection* implementan pruebas de intersección de geometría de rayos. A medida que se atraviesan las estructuras de aceleración, el sistema invocará un programa de intersección para realizar la consulta geométrica. El programa determina si y dónde el rayo toca el objeto y puede calcular normales, coordenadas de textura u otros atributos basados en la posición del impacto. Se puede asociar un número arbitrario de atributos con cada intersección. Los programas de intersección permiten el soporte para superficies arbitrarias como esferas, cilindros, superficies de orden superior o incluso geometrías fractales como en la Figura 8-1. Sin embargo, incluso en un sistema de sólo triángulos, uno puede encontrar una amplia variedad de representaciones de malla. Una operación de intersección programable facilita el acceso directo al formato nativo, lo que puede ayudar a evitar copias al interoperar con sistemas basados en rasterización.



**Figura 8-1:** Aplicaciones construidas con Optix

**Fuente:** (Parker et al., 2013, p. 95)

Los programas *Bounding Box* calculan los límites asociados con cada primitiva para habilitar estructuras de aceleración sobre geometría arbitraria. Dado un índice primitivo, un programa simple de este tipo puede, por ejemplo, leer datos de vértices de un búfer y calcular el cuadro delimitador de un triángulo. La geometría procedimental a veces solo puede estimar los límites de una primitiva. Estas estimaciones están permitidas siempre que sean conservadoras, pero los límites flexibles pueden degradar el rendimiento.

Los programas *Closest Hit* se invocan una vez que el recorrido ha encontrado la intersección más cercana de un rayo con la geometría de la escena. Este tipo de programa se parece mucho a los sombreadores de superficie en los sistemas de renderizado clásicos. Por lo general, un programa de impacto más cercano realizará cálculos como el sombreado, proyectando potencialmente nuevos rayos en el proceso y almacenará los datos de resultados en la carga útil de rayos.

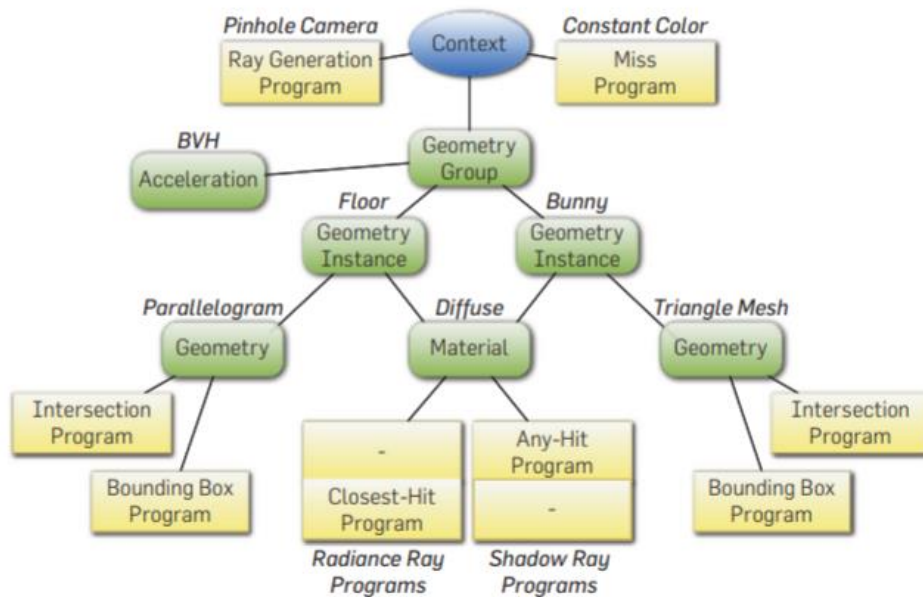
Los programas *Any Hit* se llaman durante el recorrido para cada intersección rayo-objeto que se encuentra. El programa *any hit* permite que el material participe en las decisiones de intersección de objetos mientras mantiene las operaciones de sombreado separadas de las operaciones de geometría. Opcionalmente, puede terminar el rayo usando la función incorporada *rtTerminateRay*, que detendrá todo el recorrido y desenrollará la pila de llamadas a la invocación más reciente de *rtTrace*. Este es un mecanismo de excepción ligero que se puede utilizar para implementar la terminación temprana de rayos para rayos de sombra y oclusión ambiental. Alternativamente, el programa *any hit* puede ignorar la intersección usando *rtIgnoreIntersection*, permitiendo que el recorrido continúe buscando otros objetos geométricos. Una intersección se puede ignorar, por ejemplo, en base a una búsqueda de canal de textura, implementando así una transparencia mapeada alfa eficiente sin reiniciar el recorrido.

Los programas *Miss* ejecutan cuando el rayo no se cruza con ninguna geometría en el intervalo proporcionado. Se pueden utilizar para implementar un color de fondo o una búsqueda de mapas de entorno.

Los programas *Exception* se ejecutan cuando el sistema encuentra una condición excepcional, por ejemplo, cuando la pila de recursividad excede la cantidad de memoria disponible para cada hilo, o cuando un índice de acceso al búfer está fuera de rango. OptiX también admite excepciones definidas por el usuario que se pueden lanzar desde cualquier programa. El programa de excepción puede reaccionar, por ejemplo, imprimiendo mensajes de diagnóstico o visualizando la condición escribiendo valores de color especiales en un búfer de píxeles de salida.

Los programas *Selector Visit* exponen la capacidad de programación para el recorrido de gráficos de nodos de nivel grueso. Por ejemplo, una aplicación puede optar por variar el nivel de detalle geométrico de las partes de la escena de forma individual. En este caso, el programa de visita examinaría la distancia del rayo o un diferencial de rayo almacenado con la carga útil y tomaría una decisión transversal basada en esos datos.

En la Figura 9-1 se puede observar la ejecución de los mencionados programas:



**Figura 9-1:** Ejecución de programas en Optix

Fuente: (Parker et al., 2013, p. 96)

- El programa *Ray Generation* crea rayos y los traza contra el grupo de geometría, que inicia el recorrido de BVH.
- Si el rayo se cruza con la geometría, se llamará al programa *Closest Hit* después de encontrar el punto de impacto.
- El material generará rayos de sombra y los trazará contra la geometría de la escena.
- Cuando se encuentra una intersección a lo largo del rayo de sombra, el programa *Any Hit* terminará el recorrido del rayo y volverá al programa de llamada con la información de la sombra.
- Si un rayo no se cruza con ninguna geometría de escena, se llamará al programa *Miss*.

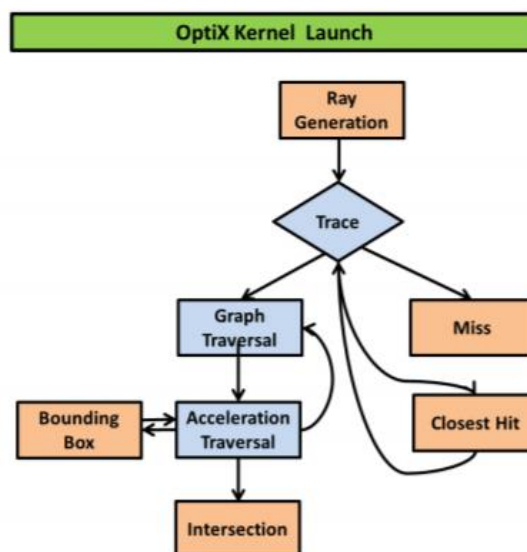
### OPTIX como Motor

*OptiXTM* no es solo una biblioteca de funciones; es un motor de software que impulsa el flujo de eventos entre los diversos componentes del modelo y sus comportamientos simulados. La función principal del motor es generar rayos en una escena y rastrear cada rayo hasta que se alcanza una condición final. El motor facilita la generación, rastreo y terminación de rayos de acuerdo con un modelo de ejecución *OptiXTM* que el usuario no tiene acceso para modificar.

Dentro del *kernel* OptiX, el programa *Ray Generation* invoca la función inaccesible OptiX *rtTrace*, que inicia el recorrido del rayo. El motor impulsa el recorrido haciendo referencia al

gráfico que se construyó previamente en el host. Una vez que finaliza *rtTrace*, vuelve al programa *Ray Generation*, donde se guardan las salidas. El programa de generación de rayos cumple una función análoga a la función principal de un programa en C. Como la función principal de C, cualquier cálculo posterior realizado por el *kernel*, desde emitir rayos hasta leer y escribir desde búferes, es generado por el programa de generación de rayos. Sin embargo, a diferencia de un programa en serie C, un programa de generación de rayos OptiX se ejecuta muchas veces en paralelo, una para cada hilo, lo que implica el número de rayos que se están simulando. Aunque el usuario lanza una serie de programas de generación de rayos paralelos, OptiX está a cargo de administrar la cantidad real de subprocesos lanzados por la GPU. Antes de la ejecución, todos los archivos de código se compilan sin conexión, lo que da como resultado una serie de archivos PTX, el lenguaje ensamblador abstracto de CUDA. En tiempo de ejecución, el motor toma el control y analiza el PTX e inserta varias optimizaciones para la GPU específica que se utiliza. El motor OptiX compila tanto el código interno como el especificado por el usuario en un *megakernel* y lanza suficientes subprocesos para llenar la máquina a la vez. Si hay más subprocesos que deben ejecutarse, se colocan en una cola. A través de este modelo de ejecución, el motor OptiX puede gestionar tanto la ejecución como la coherencia de los datos (Williams, 2013, p. 23).

En el simulador de trazado de rayos, la función *rtTrace* también se llama en *Closest Hit*, donde los rayos se generan de forma recursiva. Cada vez que se llama a *rtTrace*, se le asigna un nodo como entrada, que le indica al motor qué nodo del gráfico debe iniciar el recorrido. En la Figura 10-1 se muestra el flujo de ejecución para el *kernel* OptiX del simulador *ray tracing* donde en rojo se tiene las funcionalidades programadas por el usuario y en azul los módulos inaccesibles.



**Figura 10-1:** Ejecución del Kernel Optix

Fuente: (Williams, 2013, p. 24)

## Modelo de Ejecución de OPTIX

Se proponen diferentes modelos de ejecución para el trazado de rayos (Parker et al., 2010, p. 9). En particular, el enfoque de kernel monolítico, o *megakernel*, resulta exitoso en las GPU modernas. Este enfoque minimiza la sobrecarga de lanzamiento del *kernel*, pero reduce potencialmente la utilización del procesador a medida que los requisitos de registro crecen al máximo en los *kernels* constituyentes. Esta es una compensación difícil ya que las GPU son multiproceso y enmascaran la latencia de la memoria. OptiX implementa un *megakernel* vinculando una gran cantidad de programas de usuario individuales e iterando sobre las máquinas de estado causadas por el flujo de ejecución entre ellas en tiempo de ejecución.

Un enfoque sencillo para la ejecución de *megakernel* es una iteración simple sobre una construcción “*switch-case*” o “declaración de cambiador”. Dentro de cada caso, se ejecuta un programa de usuario y el resultado de este cálculo es el caso, o estado, para seleccionar en la siguiente iteración. Dentro de dicho mecanismo de máquina de estado, OptiX puede implementar llamadas a funciones, recursividad y excepciones.

Un mega kernel es un kernel en el que todo el software de modo de usuario habitual reside dentro del kernel. Esto puede abarcar desde la interfaz de usuario dentro del kernel hasta las utilidades, las interfaces de usuario y las aplicaciones escritas en el kernel, así como todo el sistema operativo. Esta arquitectura de kernel es muy simple y eficiente, ya que no requiere administración de memoria ni carga de software. Sin embargo, puede causar problemas que otros kernels no tienen.

### 1.10. Unity

Unity es un entorno de videojuegos multiplataforma desarrollado por la empresa Unity Technologies en 2004. Pensado sobre todo para el desarrollo de videojuegos para PC, Unity está disponible para plataformas Windows, OS X y Linux. El motor gráfico de Unity utiliza Direct3D para plataformas Windows, OpenGL para OS X y Linux, OpenGL ES para dispositivos Android e iOS, e interfaces propietarias propias de algunas consolas como Wii. Unity, además, utiliza PhysX como motor de físicas. Sus lenguajes de programación principales son UnityScript, C# y Boo (Blázquez et al., 2017).

El motor que Unity nos brinda posee un gran número de interesantes características incluso si la versión que se está utilizando sea gratuita ya que tiene particularidades de otros motores de costo elevado. Unity en cuanto a renderizado posee un pipeline moderno conocido como *deferred*

*rendering pipeline* que, sin reducir demasiado el rendimiento nos permite usar varias luces dinámicas con gran desempeño, lo cual dentro de los videojuegos es importante tener buena iluminación ya que ayuda al realismo de los mismos. Unity posee una colección de *shaders* que son programas que tienen ejecución en la tarjeta de video y nos indica la forma en que un polígono es dibujado; son utilizados con el propósito de crear efectos especiales como iluminación, fuego o niebla. Unity posee un renderizador que utiliza varios hilos para mejor eficiencia y un manejo de luces sencillo en el que para mejorar el desempeño y calidad de las imágenes se puede usar *lightmapping*, ya que en vez de usar cálculos usa texturas para iluminar las escenas (Landa, 2013, p. 14).

La creación de scripts en Unity utiliza JavaScript, C # o Boo y viene con un editor de scripts integrado llamado MonoDevelop; también se puede utilizar el editor favorito del usuario. Se recomienda utilizar C # debido a su naturaleza orientada a objetos. También hay algunas herramientas de secuencias de comandos visuales en la *asset store* para usuarios que no deseen escribir código alguno.

Para gráficos 3D se necesita un software de modelado 3D. Unity puede importar algunos formatos, sin embargo, otros formatos pueden necesitar que se instale el software de modelado para importar los modelos. Por lo general, Unity puede importar más información de los archivos que solo la malla, por ejemplo, se pueden importar mapas de altura, animaciones y texturas. Unity tiene una herramienta llamada *Mecanim* para crear y ajustar animaciones para modelos 3D; además, Unity tiene una herramienta llamada *Shuriken* para crear diversos efectos de partículas usando colores, formas, mallas, etc. Para terrenos, Unity tiene su propio editor de terrenos (Nelimarkka, 2014, p. 17).

## **Elementos Comunes**

En esta sección se describe los conceptos básicos de *Unity*.

### **MonoBehaviour**

*MonoBehaviour* es la clase básica de Unity. Todos los nuevos scripts se heredan de esta clase de forma predeterminada. *MonoBehaviour* proporciona métodos que implementan el ciclo de vida de Unity. Si la clase se hereda de *MonoBehaviour*, ésta:

- Se puede adjuntar a los objetos Unity3D como un componente
- Brinda la posibilidad de tener acceso a variables, listas, matrices de dicha clase con el panel Inspector en Unity3D Editor.

- Admite métodos y eventos estándar de Unity3D, como Start (), OnGUI (), Update (), etc.

## **GameObject**

*Game Object* es la clase principal de todos los objetos de juego existentes. Cada objeto es un contenedor. Puede contener diferentes elementos y sus combinaciones que pueden aportarles propiedades especiales. Los objetos del juego se muestran en la vista de jerarquía en el editor de Unity. Cada objeto del juego ya contiene el componente *Transform*. No es posible crear el *GameObject* sin él. *Transform* se usa con mucha frecuencia. Define la posición, el ángulo de rotación y la escala del objeto (Penzentcev, 2015, p. 26).

## **Scripts**

Cuando se crea el script y se adjunta al objeto, se muestra en *Inspector* como un componente. Todas las variables públicas (si no están marcadas de forma diferente) definidas en el script estarán disponibles en el Inspector del objeto. Si se adjunta alguna clase (secuencia de comandos) al objeto del juego, es posible obtener acceso directo a ella a través de la secuencia de comandos. En caso de que sea necesario obtener acceso al componente que se adjuntó a otro componente, se puede utilizar la función *GetComponent* <Transform> (). Pero no es una función muy rápida y se recomienda almacenar en caché la salida.

## **Prefabs**

Prefab es uno de los tipos de recursos que se utilizan para múltiples usos. Se puede insertar en cualquier cantidad de escenas y varias veces en una escena. Cuando una casa prefabricada se agrega a la escena, se crea su instancia. Todas las instancias son enlaces a la casa prefabricada original y, de hecho, a sus clones. Si se cambiará el prefabricado, todas sus instancias se cambiarán en consecuencia.

## **Recursos Unity**

Unity opera no solo con "*prefabs*" sino también con "*resources*". El recurso puede ser una textura, audio o video. Pero también se puede guardar como un objeto de unidad de archivo. La principal diferencia entre recursos y prefabs es la siguiente: el recurso siempre se carga en la escena "como está" y los cambios en el editor no influyen en el archivo de recursos inicial y se aplicarán solo a un objeto en particular en el que se crearon, pero la edición de la configuración de un prefab en el editor de Unity cambiará la configuración inicial de todos los objetos creados a partir de este



prefab. De forma predeterminada, Unity busca recursos en una carpeta llamada "Resources". Si existirán muchas carpetas con este nombre, se examinará cada una de ellas. El contenido de las carpetas "Resources" siempre se copia en la compilación del juego (Penzentcev, 2015).

## **Escenas Unity**

El juego Unity es, de hecho, un conjunto de escenas. Cada escena puede contener sus propios objetos con configuraciones especiales. Las escenas se utilizan para dividir el juego en varias partes y operarlas por separado. Por lo general, el juego es un conjunto de escenas ordenadas en secuencias especiales diseñadas por el autor. Las escenas se pueden utilizar para crear un menú principal, niveles individuales y cualquier otra cosa. La documentación sugiere pensar en cada archivo de escena único como un nivel único.

## **Paquetes Unity**

Los Packages son una colección de archivos y datos de proyectos de Unity, o elementos de proyectos, los cuales son comprimidos y almacenados en un solo archivo, similar a archivos Zip. Al igual que archivos Zip, un package (paquete) mantiene su estructura original de directorios cuando es desempacado, al igual que los meta-datos acerca de los assets (al igual que ajustes de importación y vínculos a otros assets). Los packages de Unity son una manera útil de compartir y re-utilizar proyectos al igual que colecciones de assets; Los Standard Assets de Unity e items en la Asset Store de Unity son proporcionados en packages (Unity3D, 2016).

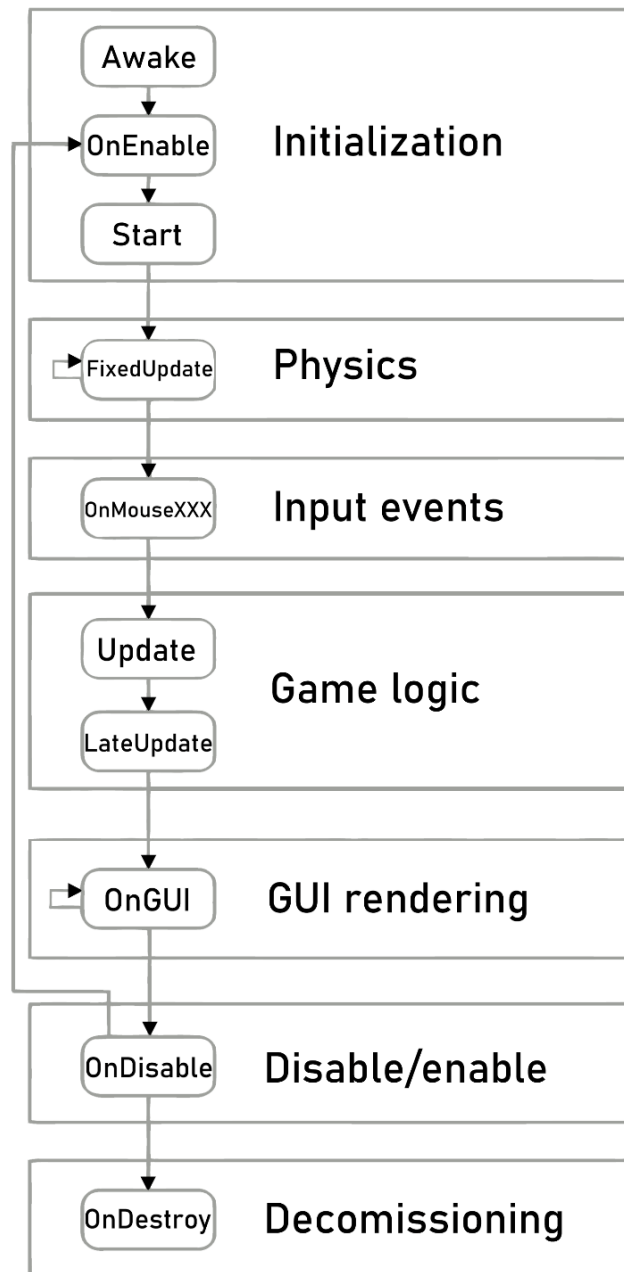
### ***1.10.1. Ciclo de vida Unity***

Unity llama a determinadas funciones de eventos en determinados momentos. Cada script heredado de MonoBehaviour puede implementar / anular esas funciones. Existe la posibilidad de deshabilitar la llamada a algunas de estas funciones. Esto se puede hacer deshabilitando la casilla de verificación en el editor. Pero solo evitará que se ejecuten Start (), Awake (), Update (), FixedUpdate () y OnGUI (). Si ninguna de estas funciones está presente, la casilla de verificación no se muestra. Las funciones pertenecen a sus etapas en el ciclo de vida. Los más importantes se enumeran a continuación (Penzentcev, 2015).

## **Carga de la primera escena**

Las siguientes funciones se llaman cuando comienza la escena (una vez por cada objeto en el cuadro):

- **Awake ():** siempre se llama primero para el objeto justo después de la inicialización de prefab (si un GameObject está inactivo durante el inicio, no se llama a Awake () hasta que se activa, o se llama a una función en cualquier script adjunto a él).
- **OnEnable ():** (llamado solo si el objeto está activo) llamado solo después de que está habilitado. Se recomienda utilizar este método para suscribirse a los eventos deseados.



**Figura 11-1.** Ejecución de eventos en Unity

Fuente: (Penzentcev, 2015, p. 30)

### **Antes de la actualización de la primera trama**

- **Start ():** se llama antes de renderizar o el primer fotograma, se llama solo una vez para un script determinado. Es posible que Start () no se llame en el mismo marco que Awake () si el script no está habilitado en el momento de la inicialización.

### **Orden de Actualización**

- **FixedUpdate ():** no depende de Update (). Se puede llamar varias veces por fotograma. Todos los cálculos físicos van inmediatamente después de FixedUpdate ().
- **Update ():** se llama una vez por fotograma. Es la función de correo para actualizaciones de marcos.
- **LateUpdate ():** se llama una vez por cuadro una vez finalizada la función de actualización. Todos los cálculos que se realicen en Update se habrán completado cuando comience LateUpdate.

### **Al Finalizar**

- **OnDisable ():** se llama cuando el comportamiento se desactiva o inactiva. Esto también se llama cuando se destruye el objeto y se puede utilizar para cualquier código de limpieza. Este método es una buena opción para cancelar la suscripción del objeto a cualquier evento al que se haya suscrito.

### **Cuando el objeto es destruido**

- **OnDestroy ():** llamado en el último marco para la existencia del objeto.

La Figura 11-1 muestra un esquema del ciclo de vida simplificado en el cual se indican solo las funciones básicas.

## **1.11. Opal**

Opal es una biblioteca para simular en GPU la propagación de canales de múltiples rutas con NVIDIA Optix para múltiples transmisores y receptores en movimiento. Opal es un simulador de propagación de radiofrecuencia determinista basado en el lanzamiento de rayos 3D. Con los métodos de lanzamiento de rayos, también llamados disparar y rebotar (SBR), las ondas electromagnéticas se simulan mediante rayos lanzados desde el transmisor con un espaciado

angular predeterminado. Estos rayos se propagan a lo largo de su trayectoria hasta que chocan con un obstáculo donde se reflejan, difractan, transmiten o dispersan. Los rayos posteriores se vuelven a trazar. Las contribuciones de los diferentes rayos que inciden en una esfera de recepción en el receptor se suman para calcular el campo eléctrico (Egea, 2021, p. 1).

Trabaja mediante el motor de trazado de rayos NVIDIA Optix para propagar los rayos en una escena dinámica, que se crea a partir de mallas cargadas desde archivos o desde otra aplicación, como un motor de juego. Por el momento, solo se calculan las reflexiones y soporta transmisiones simultáneas de múltiples transmisores a múltiples receptores. Se puede usar como una aplicación independiente o como complemento de Unity. Si se usa como complemento de Unity, un programa de Unity enviará las mallas y actualizará las transformaciones de los elementos de la escena. (Egea, 2021, p. 1).

A pesar de poseer varias ventajas para la propagación de rayos, Unity también posee bibliotecas propias del aplicativo que permite el trazado, reflexión y difracción de rayos, así como su cálculo, esto abre la opción de hacer estas simulaciones sin la necesidad de ocupar la biblioteca Opal dentro de los escenarios propuestos que se detallan en el Capítulo II.

## CAPÍTULO II

### 2. MARCO METODOLÓGICO

#### Introducción

En este capítulo se detalla el procedimiento que se utilizó para la ejecución del proyecto de investigación. Al igual que, toda la metodología de investigación, debido a que, por la complejidad del estudio planteado se usó diversas técnicas e implementación de *software* específico, mismo que, permitió el desarrollo y validación de los resultados.

#### 2.1. Tipos de investigación

En la investigación es necesario tomar en cuenta el alcance planteado en los objetivos, los cuales establecen la simulación de la propagación, y la comparación de datos simulados con los datos reales medidos dentro de la Facultad, para cumplir con los parámetros establecidos se utilizó las siguientes metodologías:

##### 2.1.1. Investigación Bibliográfica

Es el mecanismo más habitual. El investigador debe poseer un gran conocimiento del tema que va a investigar, realizar una actualización de los trabajos publicados en ese campo. Los temas objeto de investigación no suelen surgir por azar, sino directamente desde el conocimiento. Ello exige un gran esfuerzo de estudio y especialización (Crespo, 2006, p. 35).

Este tipo de investigación es de suma importancia en el desarrollo de proyectos de investigación pues, permite estructurar de forma adecuada el estudio. El uso de esta metodología permitió entender los conceptos básicos de funcionamiento del *software* utilizado. Por otra parte, facilitó conocer la teoría para el desarrollo de los cálculos, análisis y comparación.

La revisión bibliográfica (RB) consiste en la lectura y organización del material previamente escrito referente a: el problema específico que se investiga, el marco teórico y los métodos idóneos para realizar el estudio (Crespo, 2006, p. 37).

### **2.1.2. Investigación Descriptiva**

Es el estudio de un conjunto de procesos prácticos y lógicos. Identificar las características de una población, proceso social, económico, ambiental, político o cultural, así como un lugar. El objetivo principal de esta investigación es la relación entre los factores, los actores y las variables, se puede definir como un método científico que involucra describir y observar el comportamiento de un determinado fenómeno, sin embargo, de una manera adecuada (Tiposdeinvestigacion, 2021, p. 1).

Este tipo de investigación es utilizado dentro del proceso investigativo para identificar las principales características del *software* a utilizarse, al igual que, los parámetros de investigación y las diversas actividades a considerarse dentro del estudio planteado. El estudio ha permitido definir el análisis y el proceso para la elaboración y desarrollo del trabajo de titulación.

### **2.2. Método de investigación**

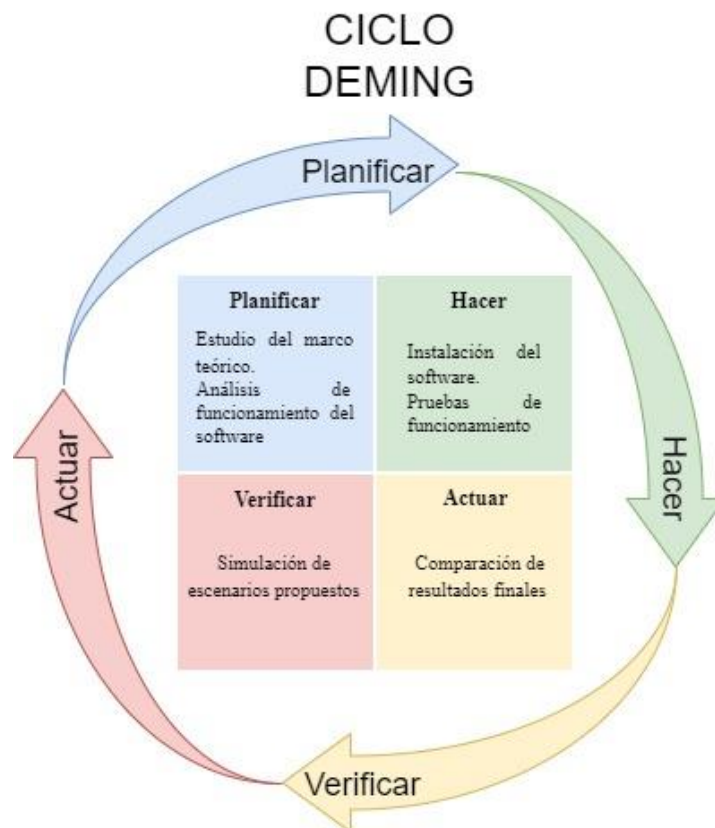
El uso del método cuantitativo permitió el desarrollo y estudio de los resultados obtenidos en la simulación. Este método de acuerdo con QuestionPro (2021) es aquella que reclama la intervención de datos cuantificables o numéricos (cantidades, magnitudes), por lo que trabajan con universos muy grandes (sobre los cuales toman muestras representativas como criterio de validación). Esta reúne datos numéricos que pueden ser jerarquizados, medidos o categorizados a través de análisis estadístico y ayuda a descubrir patrones y relaciones, así como a realizar generalizaciones.

De forma adicional, se procedió con la comparación de los resultados obtenidos en la simulación y los valores reales, los cuales se obtuvieron a través del método planteado en la investigación. Sin embargo, la metodología usada para el estudio, fue la de Deming, la cual se describe a continuación:

### **2.3. Desarrollo del estudio a través del ciclo de DEMING**

El ciclo de Deming es también denominado ciclo PDCA, por sus siglas en inglés de las etapas Plan, Do, Check y Act. De igual manera, se le conoce como espiral de mejora continua o ciclo PHVA, debido a la traducción en español de sus etapas Planificar, Hacer, Verificar y Actuar. Las aplicaciones de estas 4 etapas del ciclo de Deming permiten reevaluar los procesos una y otra vez, de forma cíclica, asegurando así, el progreso continuo de la organización (Beetrack, 2021, p. 1).

El uso del ciclo de Deming admite un desarrollo secuencial y organizado en el trabajo de titulación. Esta planificación está situada en 4 pasos, los cuales se detallan en la Figura 1-2 presentada a continuación:



**Figura 1-2:** Estructura ciclo Deming

Realizado por: Freire Kevin, 2022

### 2.3.1. Fase Planificar

La fase planificar se puede apreciar en el Capítulo I de la investigación, dentro de la cual se ha establecido el estudio de las plataformas a utilizar sobre las cuales está basada la simulación en la presente investigación; estas plataformas son las siguientes:

- Nvidia CUDA 10.0
- CMake 3.19.4
- Nvidia OPTIX 6.5
- Microsoft visual studio enterprise 2017
- Unity 2018.4.21f1

Al ser un *software* específico de Nvidia, este requiere de *hardware* que posea alta capacidad de procesamiento, es por tal razón que, se utilizó un equipo de las siguientes características:

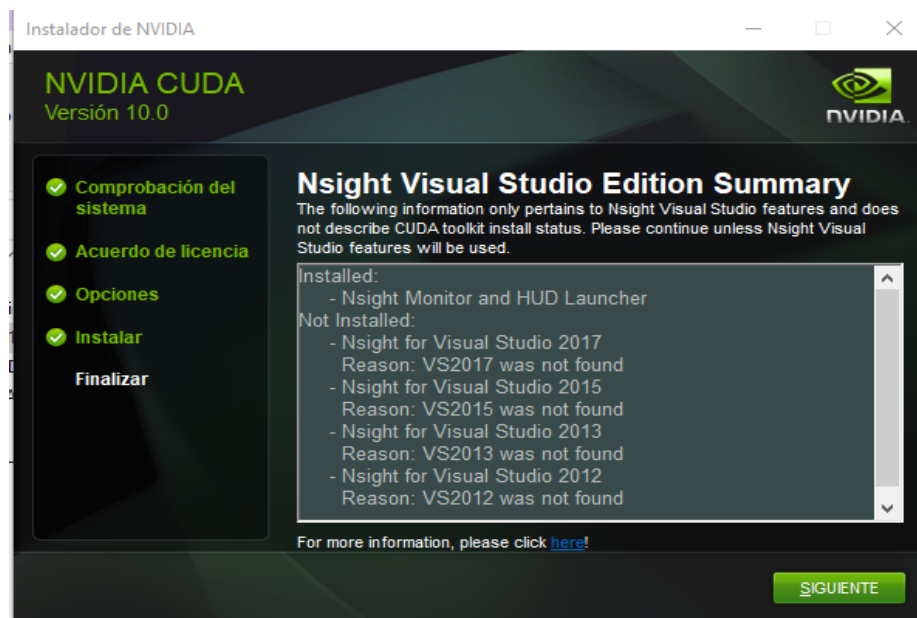
**Tabla 1-2:** Descripciones Equipo de cómputo

Equipo	Característica
Procesador	Core I7 8700K 3.7 GHz
Memoria RAM	64 GB DDR4
Tarjeta de Video	Nvidia Geforce GTX 1050 TI

Realizado por: Freire Kevin, 2022

### 2.3.2. Fase Hacer

- **Instalación del software:** Para el desarrollo del trabajo de investigación se procedió con la instalación del *software* necesario, a fin de realizar las debidas instalaciones, a diferencia de otras plataformas el uso de *Nvidia Optix* requiere un proceso de instalación con ciertos procedimientos a seguir, el cual se detalla en los pasos presentados a continuación:
- **Paso 1:** Instalar el CUDA, de acuerdo a lo estudiado en el Capítulo I, este *software* propiedad de la empresa Nvidia, permite el desarrollo de aplicaciones con motor gráfico; es compatible solo con controladores de video de la misma empresa y consiente el uso de la herramienta *Nsight Monitor and HUD Launcher*, como se evidencia en la Figura 2-2:

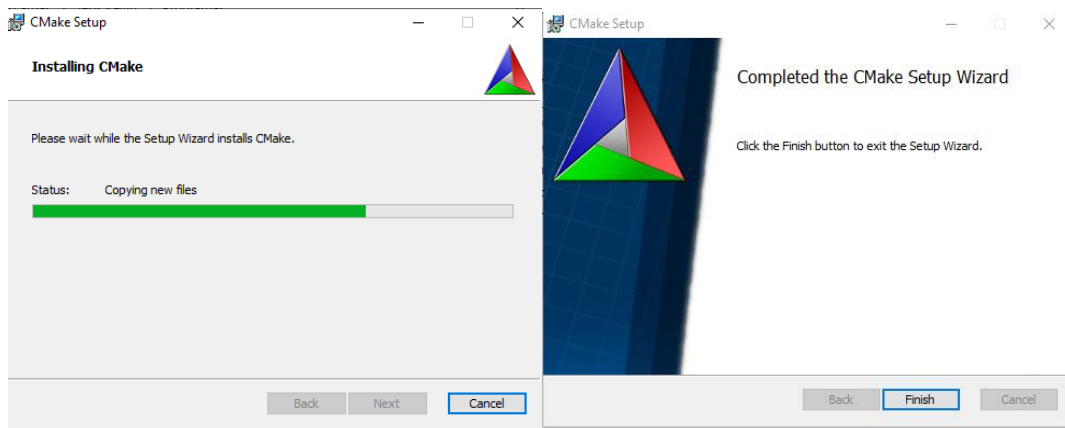


**Figura 2-2:** Instalación exitosa de CUDA 10.0

Realizado por: Freire Kevin, 2022



- **Paso 2:** Instalación de CMAKE 3.19.4, la plataforma CMAKE permite la compilación del Optix con el *software* de programación Visual Estudio 2017, la correcta instalación de dicha herramienta se visualiza en la Figura 3-2 a continuación:



**Figura 3-2:** Instalación exitosa de *Cmake*

**Realizado por:** Freire Kevin, 2022

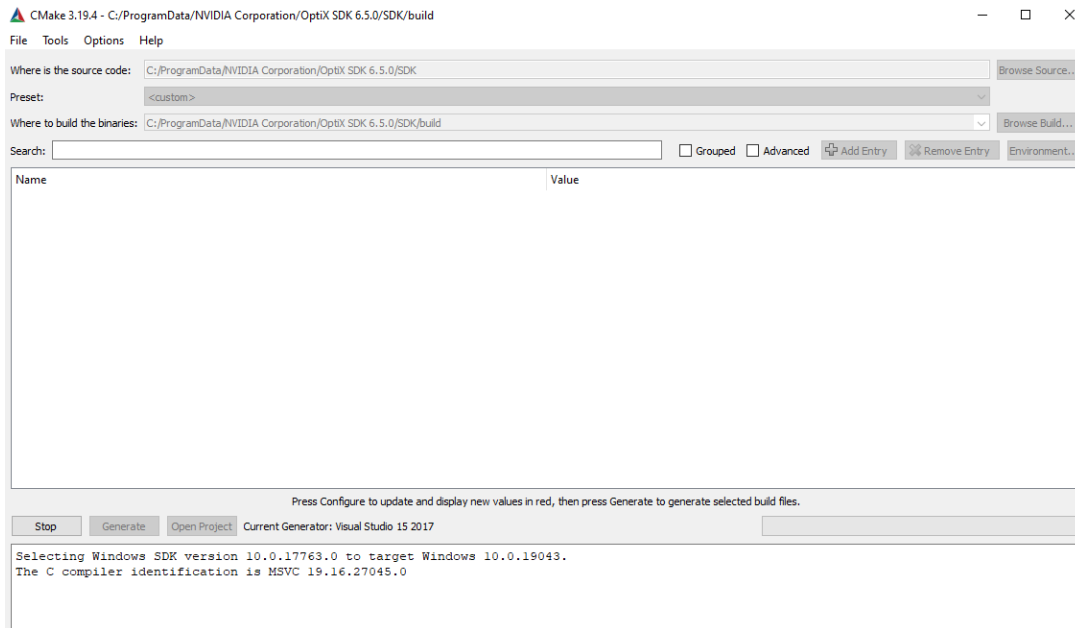
- **Paso 3:** Instalación Nvidia Optix 6.5.0, el API para el trazado de rayos de *Nvidia*, permite la integración del *software* con dimensionamiento en 3D para un mejor análisis, la instalación se evidencia en la Figura 4-2:



**Figura 4-2:** Instalación exitosa de Optix 6.5.0

**Realizado por:** Freire Kevin, 2022

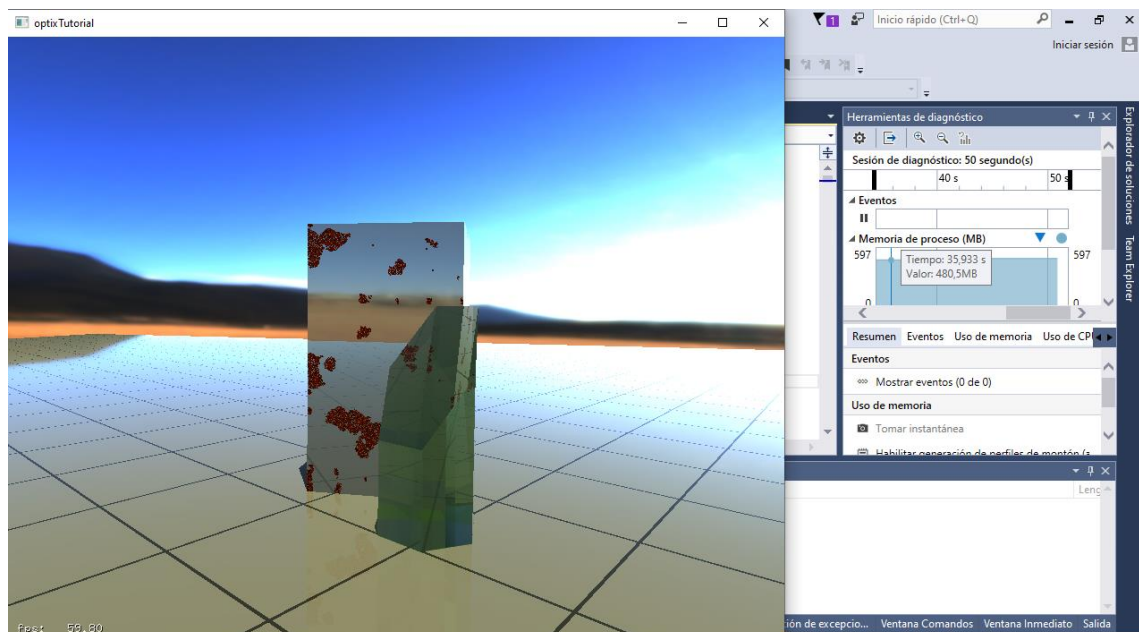
- **Paso 4:** Compilación del programa, una vez realizado la instalación de los programas necesarios para la creación del proyecto, se procede con la compilación del proyecto, para lo cual, se debe abrir *Cmake*, posteriormente buscar el origen de donde se instaló *Optix* esta ruta es: C:\ProgramData\NVIDIA Corporation\OptiX SDK 6.5.0\SDK, como se muestra en la Figura 5-2 a continuación:



**Figura 5-2:** Compilación exitosa de *Cmake*

Realizado por: Freire Kevin, 2022

- **Paso 5:** Comprobación, una vez compilada de forma correcta las plataformas instaladas, se podrá realizar renderizaciones en 3D a través de Visual Estudio, con la ayuda de CUDA como motor gráfico y *Optix* para el trazado de rayos, como se evidencia en la Figura 6-2:

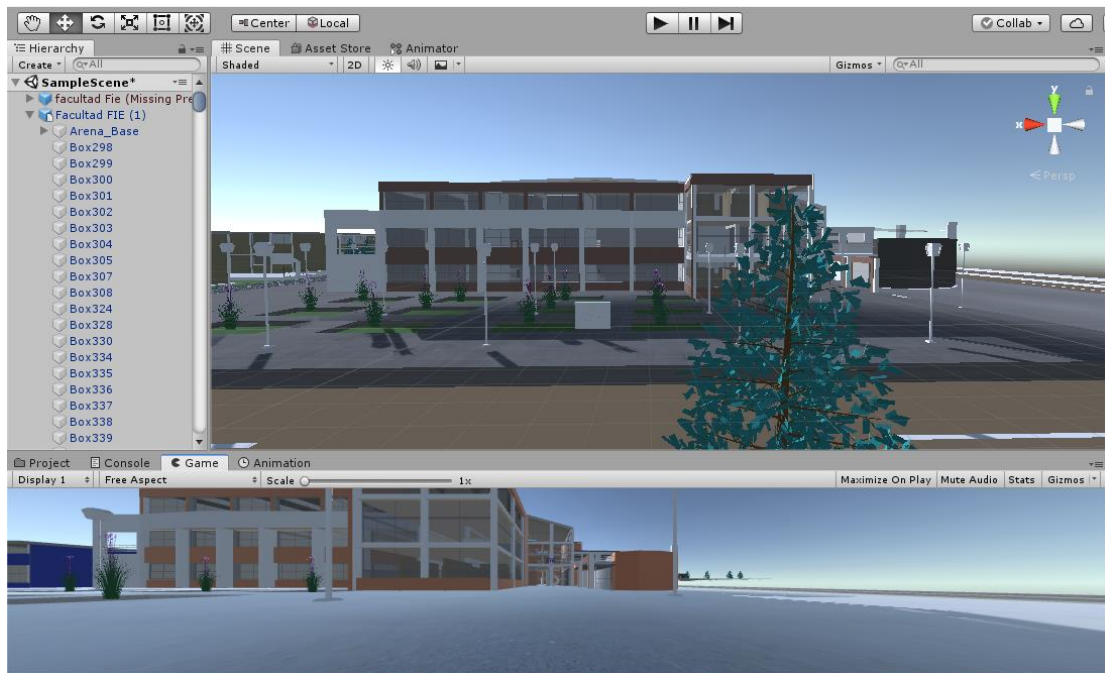


**Figura 6-2:** Renderización de *Optix* con *Cmake*

Realizado por: Freire Kevin, 2022

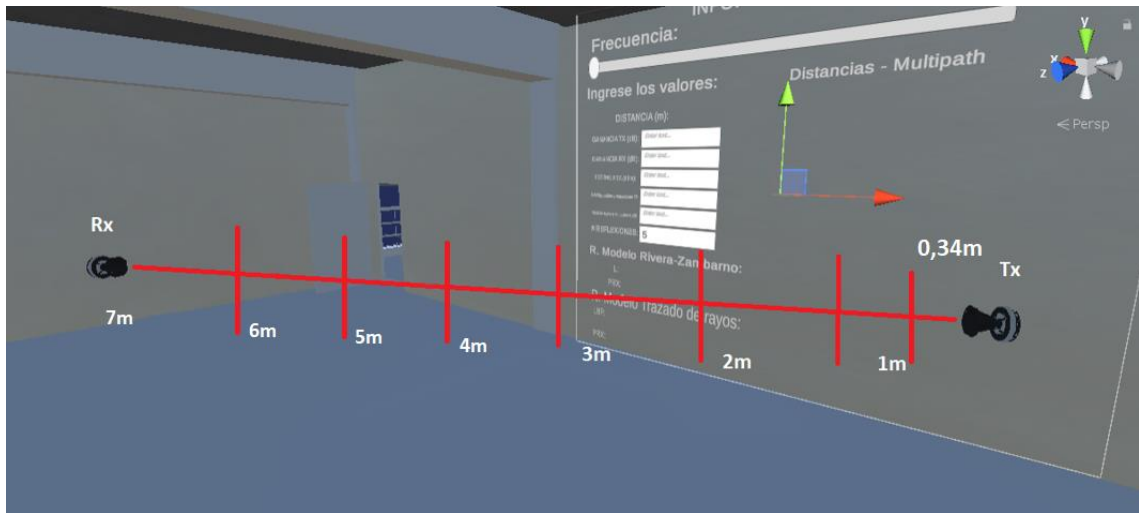
### 2.3.3. Fase Actuar

Una vez verificado que todo el sistema se haya compilado de forma correcta, se procedió con el planteamiento de los escenarios propuestos para las simulaciones en línea de vista directa, en esta simulación se usó como escenario la Facultad de Informática y Electrónica, en el *software Unity* como se evidencia en la Figura 7-2 detallada a continuación:



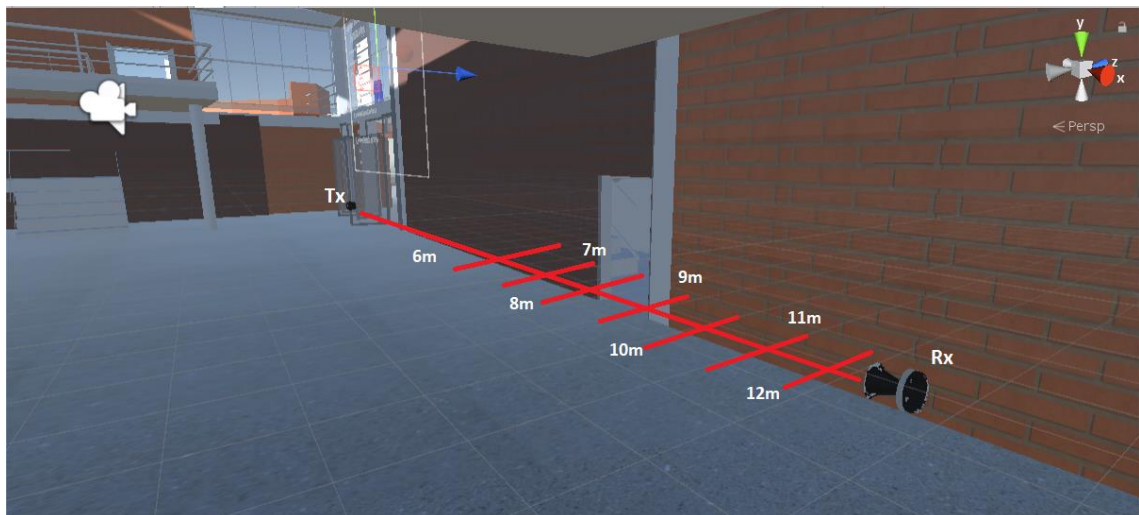
**Figura 7-2:** Simulación de la infraestructura de la Facultad de Informática y Electrónica en *Unity*  
**Realizado por:** Freire Kevin, 2022

- **Escenario 1:** En el primer escenario se propone un análisis de la propagación en vista directa, pero a corta distancia. Este escenario está planteado dentro del laboratorio de microondas ubicado en el tercer piso del edificio de la Facultad de Informática y Electrónica; la distancia máxima de transmisión es de 7 m y la mínima de 0,34 m correspondiente a la distancia de *far field*, en frecuencias de 10-20 Ghz, como se muestra en la Figura 8-2.
- **Escenario 2:** Esta propuesto el escenario de propagación de línea de vista directa, pero a mayor distancia. Es por eso que, dicho escenario se ha planteado en el corredor de la planta baja del edificio de la Facultad de Informática y Electrónica, a una distancia mínima de 6m y máxima de 12 m, en rangos de frecuencia de 10 a 20 Ghz, el escenario propuesto se evidencia en la Figura 9-2 a continuación:



**Figura 8-2:** Simulación 1 propagación en línea directa Laboratorio de Microondas

Realizado por: Freire Kevin, 2022



**Figura 9-2:** Simulación 2 propagación en línea directa pasillo primer piso

Realizado por: Freire Kevin, 2022

- **Escenario 3:** Pruebas reales en laboratorio de microondas.

En la Figura 10-2 se puede apreciar el laboratorio de microondas.

- **Escenario 4:** Pruebas reales en planta baja.

En la Figura 11-2 se muestra la planta baja del edificio de la Facultad de Informática y Electrónica donde se realizó las respectivas pruebas.



**Figura 10-2:** Laboratorio de microondas

Realizado por: Freire Kevin, 2022



**Figura 11-2:** Planta baja

Realizado por: Freire Kevin, 2022

#### **2.3.4. Fase Verificar**

Es la fase final del Ciclo de Deming donde se obtuvieron los valores de las potencias de recepción (PR<sub>x</sub>), para lo cual, en base a los modelos de propagación estudiados, Trazado de Rayos (en un entorno sin pérdidas) y Rivera-Zambrano, se definió los parámetros de entrada y salida en el simulador.

### ***Definición de Parámetros de Entrada del Simulador:***

- ***Potencia de Transmisión:*** La potencia de transmisión en los escenarios reales y simulados es de 18 dBm, es el valor con el que propagará la antena transmisora hacia la receptora.
- ***Pérdidas por cables y conectores:*** Estos valores pueden variar entre 1 a 3 dB, sin embargo, para el estudio planteado este valor será constante, de 2.
- ***Distancia:*** La distancia varía según el escenario propuesto, estos son:
  - Escenario 1: De 0.34m a 7m
  - Escenario 2: De 6m a 12m
- ***Ganancia de Receptor y Transmisor:*** Son valores que varían acorde a la frecuencia de operación, según Zambrano y Rivera (2020) estos valores se muestran en la Tabla 2-2:

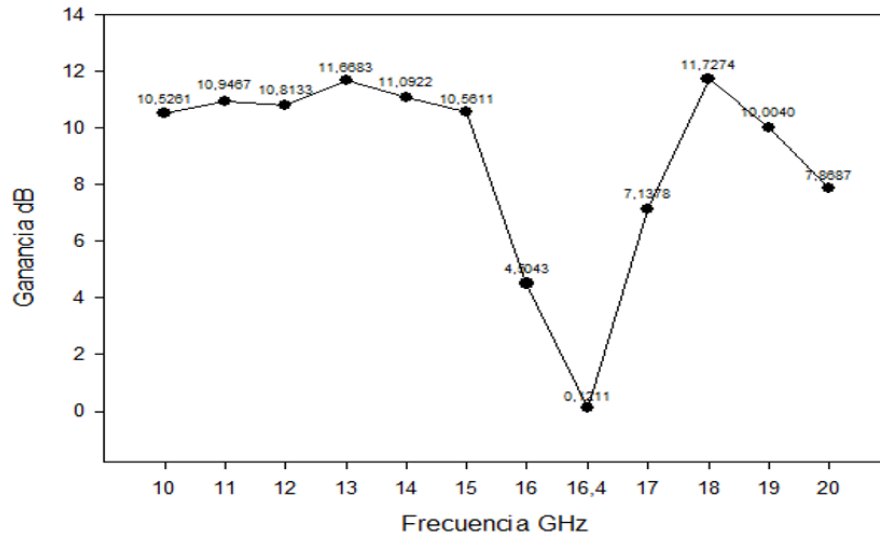
**Tabla 2-2:** Valores de Ganancia para Tx y Rx

<b>Frecuencia (GHz)</b>	<b>Ganancia (dBi)</b>
10	10,5261
11	10,9467
12	10,8133
13	11,6683
14	11,0922
15	10,5611
16	4,5043
17	7,1378
18	11,7274
19	10,004
20	7,8687

**Realizado por:** Freire Kevin, 2022

Para un mejor entendimiento se presenta en la Figura 12-2 la comparativa de las ganancias:

- ***Número de reflexiones:*** El número de reflexiones fué definido por defecto en 5, sin embargo, este podrá ser modificable de acuerdo con las necesidades del estudio.
- ***Frecuencia de operación:*** La frecuencia está definida entre 10 a 20 GHz.



**Figura 12-2:** Ganancia Vs Frecuencia para antenas Tx y Rx

Realizado por: Freire Kevin, 2022

**Definición de Parámetros de Salida del Simulador:**

- **Pérdidas de propagación:** Valor que se refiere a la cantidad de potencia que se ha perdido durante el trayecto de la señal. Para los modelos estudiados depende tanto de la distancia como de la frecuencia de operación.
- **Potencia de Recepción:** Es el valor de potencia que llega a la antena receptora.
- **Distancia-Multipath:** Valores de distancia recorridos por cada una de las componentes Multipath hasta la antena receptora.

Los resultados de las potencias en cada uno de los escenarios son presentados en el Capítulo III.

**2.4. Cálculo de Potencias de Recepción mediante Balance del Enlace**

Para el cálculo de los valores de potencia de recepción dentro del simulador con el modelo Trazado de Rayos, se basa en la ecuación de Balance del Enlace (Rábanos et al., 2013, p.80), dado por la Ecuación 1-2.

$$P_{Rx}(dBm) = P_{Tx} - L_{bf} + G_{TX} + G_{RX} - \text{Pérdidas cables y conectores (Tx)} - \text{Pérdidas cables y conectores (Rx)}$$

**Ecuación 1-2:** PRx mediante balance del enlace para modelo Trazado de Rayos

Donde:

- $P_{Rx}$ : Potencia de recepción
- $P_{Tx}$ : Potencia de Transmisión
- $L_{bf}$ : Pérdidas de propagación en espacio libre (Ecuación 2-1)
- $G_{Tx}$ : Ganancia del transmisor
- $G_{Rx}$ : Ganancia del receptor

De igual manera, se realiza el cálculo de potencia de recepción para el Modelo de propagación Rivera-Zambrano a fin de evaluar la diferencia en los valores obtenidos, a través del simulador con los resultados reales, para lo cual también se usa el Balance del Enlace como se observa en la Ecuación 2-2:

$$P_{Rx}(dBm) = P_{Tx} - L + G_{Tx} + G_{Rx} - \text{Pérdidas cables y conectores (Tx)} \\ - \text{Pérdidas cables y conectores (Rx)}$$

**Ecuación 2-2:** PRx mediante balance del enlace para modelo Rivera-Zambrano

Donde:

- $P_{Rx}$ : Potencia de recepción
- $P_{Tx}$ : Potencia de Transmisión
- $L$ : Pérdidas de propagación modelo Rivera-Zambrano (Ecuación 3-1)
- $G_{Tx}$ : Ganancia del transmisor
- $G_{Rx}$ : Ganancia del receptor

## 2.5. Cálculo de Error Cuadrático Medio

El error cuadrático medio (RMSE) examina un conjunto de dos valores y mide la cantidad de error que existe entre ellos. Es decir, compara los valores pronosticados y medidos, como se muestra en la Ecuación 3-2:



$$RMSE = \sqrt{\frac{\sum_{i=1}^m e_i^2}{m}}$$

**Ecuación 3-2:** Error cuadrático medio

Donde:

$e_i$ : es el residual o error aparente

$m$ : el número de datos analizados

Además, el residual o error aparente está definido por la Ecuación 4-2:

$$e = h_i - \hat{h}_i$$

**Ecuación 4-2:** Residual o error aparente

Donde:

$h_i$ : es el valor observado

$\hat{h}_i$ : es el valor calculado

## CAPITULO III

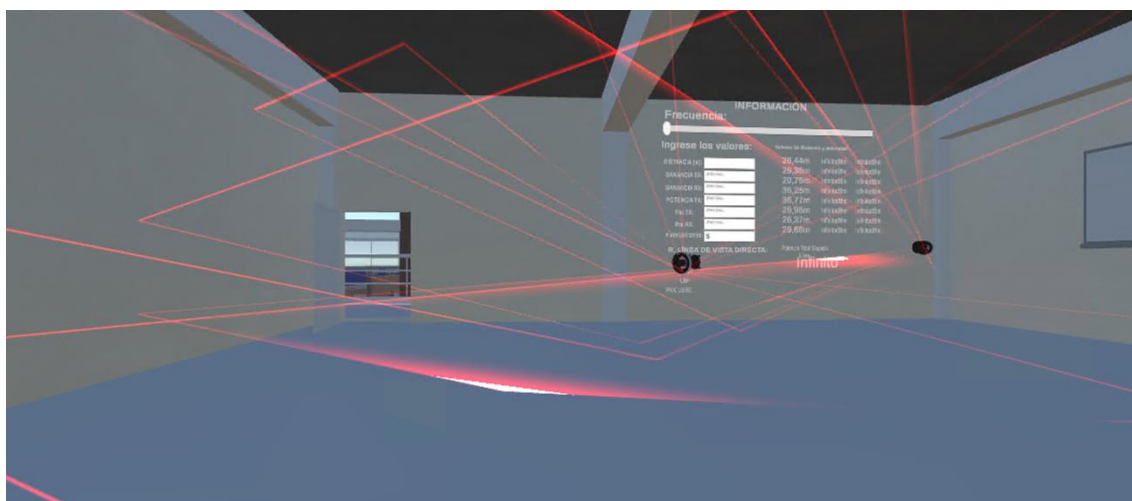
### 3. MARCO DE RESULTADOS, ANÁLISIS Y DISCUSIÓN DE RESULTADOS

El presente capítulo tiene por objetivo, realizar un análisis de los resultados obtenidos dentro de los escenarios propuestos los cuales son: dos reales y dos simulados en la plataforma *Unity*, mediante el renderizador Nvidia Optix 6.5.

Una vez realizado la programación del simulador en la plataforma *Unity*, con la ayuda de Nvidia Optix, para la renderización de los rayos de propagación, así como el cálculo de las distancias en cada una de las reflexiones, se procedió con la simulación de los escenarios propuestos en el Capítulo II, la visualización de los valores calculados se los obtuvo a través de la función *Canvas* que se presenta a continuación:

#### 3.1. Escenario 1: Laboratorio de microondas (virtual)

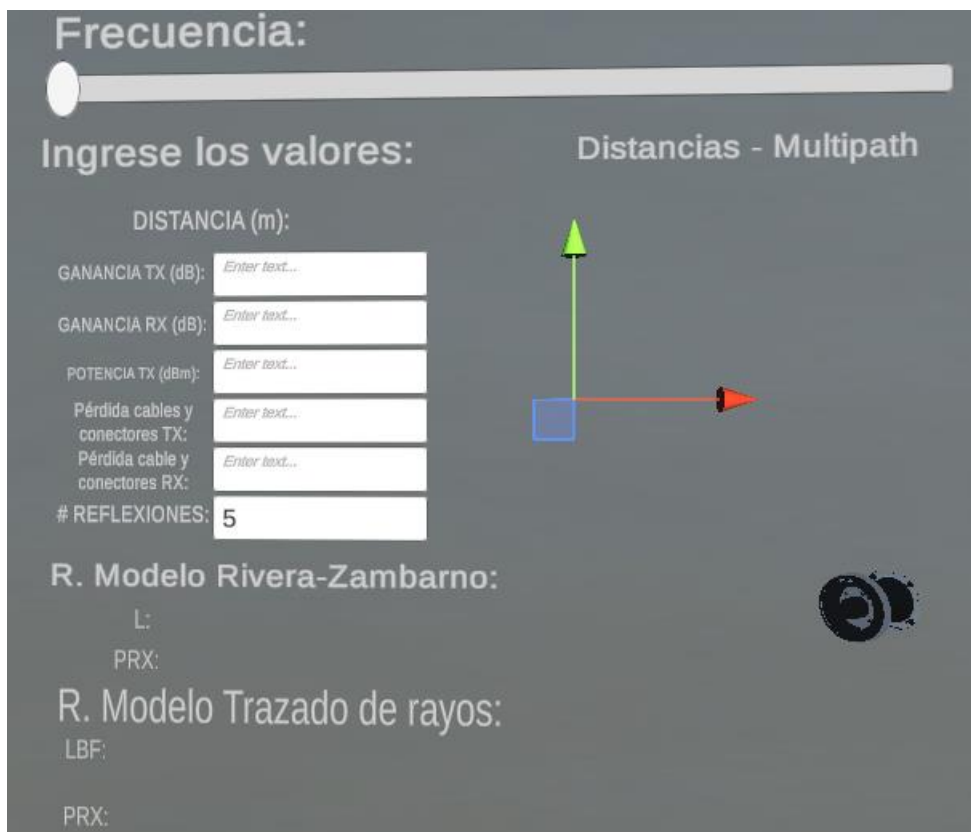
Una vez realizada las simulaciones planteadas en los dos escenarios, se procedió con la configuración de los rayos en el Laboratorio de Microondas, este posee rayos de propagación en diferentes direcciones, incluido el de vista directa, como se evidencia en la Figura 1-3:



**Figura 1-3:** Propagación de rayos Escenario 1

**Realizado por:** Freire Kevin, 2022

Cada uno de los rayos tiene parámetros de configuración específicos, mismos que, fueron ingresados mediante el elemento *Canvas*, el cual, permitió la visualización de los resultados, dicha interfaz se muestra en la Figura 2-3:



**Figura 2-3:** Canvas Escenario 1

**Realizado por:** Freire Kevin, 2022

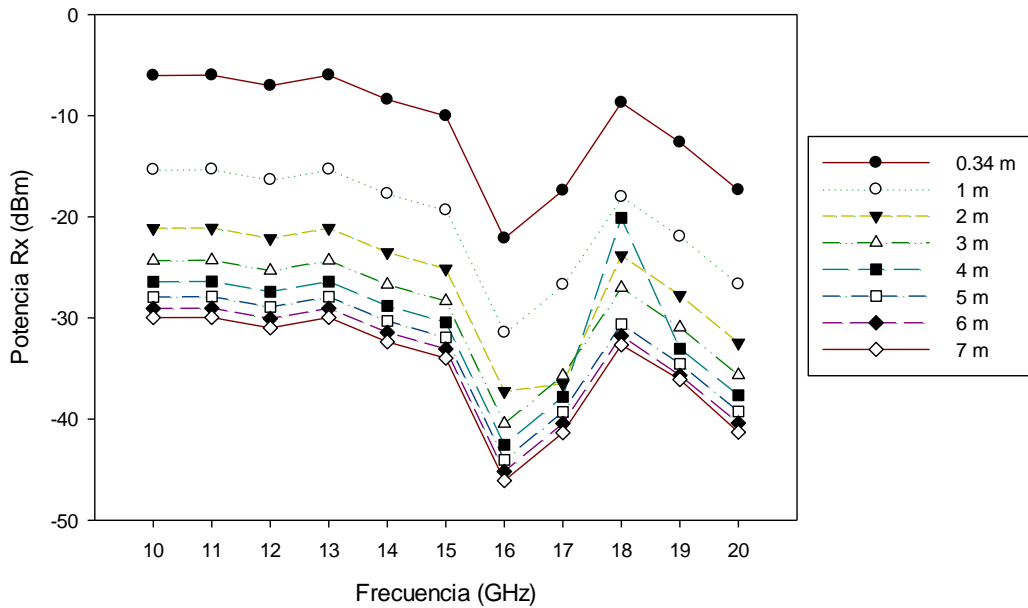
De forma adicional se procedió con la obtención de los valores de potencia de recepción para cada una de las distancias y frecuencias, y con cada uno de los modelos propuestos, estos valores (en dBm) se los aprecia en la Tabla 1-3:

**Tabla 1-3:** Valores PRx (dBm) en Modelo de Propagación Trazado de Rayos Escenario 1

F(GHz)/D(m)	0,34	1	2	3	4	5	6	7
10	-6,03	-15,34	-21,14	-24,33	-26,42	-27,93	-29,07	-29,97
11	-5,99	-15,3	-21,11	-24,3	-26,4	-27,9	-29,03	-29,95
12	-7,03	-16,33	-22,14	-25,33	-27,42	-28,93	-30,06	-30,98
13	-6,01	-15,32	-21,12	-24,32	-26,41	-27,91	-29,05	-29,96
14	-8,41	-17,72	-23,52	-26,71	-28,81	-30,31	-31,44	-32,36
15	-10,03	-19,34	-25,14	-28,33	-30,43	-31,93	-33,07	-33,98
16	-22,14	-31,45	-37,25	-40,45	-42,54	-44,04	-45,18	-46,09
17	-17,4	-26,71	-36,52	-35,71	-37,8	-39,3	-40,44	-41,35
18	-8,72	-18,03	-23,83	-27,02	-20,11	-30,62	-31,76	-32,67
19	-12,64	-21,94	-27,75	-30,94	-33,04	-34,54	-35,68	-36,09
20	-17,35	-26,66	-32,47	-35,65	-37,65	-39,25	-40,39	-41,3

**Realizado por:** Freire Kevin, 2022

Para un mejor entendimiento se presenta la Figura 3-3 donde se muestra la variación de cada uno de las potencias obtenidas:



**Figura 3-3:** Diagrama PRx vs Frecuencia en Modelo Trazado de Rayos Escenario 1

Realizado por: Freire Kevin, 2022

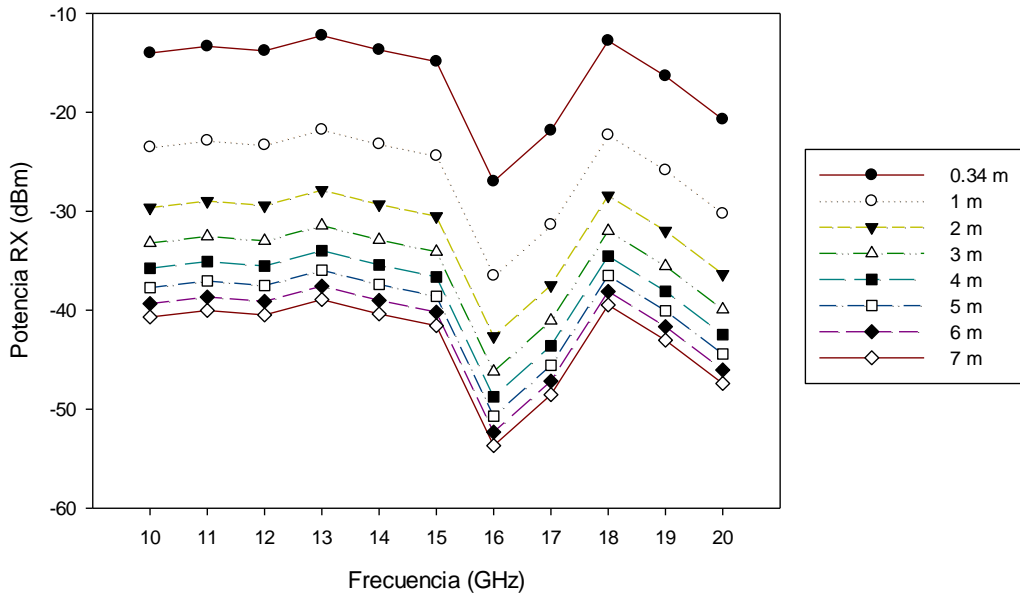
De igual forma se procedió con la obtención de los valores utilizando el modelo de propagación Rivera-Zambrano, este debió ser analizado mediante el rayo de vista directa puesto que, la fórmula utilizada ya contempla las pérdidas de todas las reflexiones, como se evidencia en la Tabla 2-3:

**Tabla 2-3:** Valores PRx (dBm) en Modelo de Propagación Rivera-Zambrano Escenario 1

F(GHz)/D(m)	0,34	1	2	3	4	5	6	7
10	-14,01	-23,52	-29,64	-33,21	-35,76	-37,73	-39,33	-40,69
11	-13,34	-22,86	-28,97	-32,55	-35,09	-37,05	-38,67	-40,02
12	-13,79	-23,31	-29,42	-33,17	-35,53	-37,51	-39,11	-40,47
13	-12,25	-21,76	-27,88	-31,45	-33,99	-35,96	-37,57	-38,93
14	-13,69	-23,21	-29,31	-32,9	-35,43	-37,4	-39,01	-40,37
15	-14,88	-24,4	-30,51	-34,09	-36,63	-38,6	-40,2	-41,56
16	-26,99	36,51	-42,63	-46,2	-48,74	-50,71	-52,32	-53,68
17	-21,85	-31,37	-37,48	-41,06	-43,6	-45,57	47,17	-48,53
18	-12,78	-22,3	-28,42	-31,99	-34,53	-36,5	-38,11	-39,47
19	-16,34	-25,86	-31,98	-35,55	-38,09	-40,06	-41,66	-43,03
20	-20,72	-30,24	-36,35	-39,93	-42,47	-44,43	-46,04	-47,4

Realizado por: Freire Kevin, 2022

A continuación, se detalla en la Figura 4-3 los resultados obtenidos:

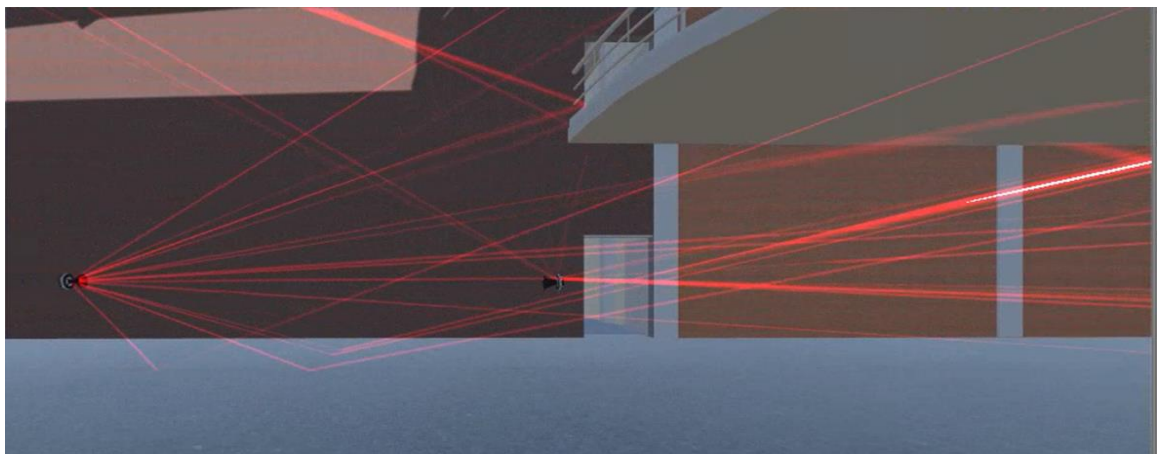


**Figura 4-3:** Diagrama PRx vs Frecuencia en Modelo Rivera-Zambrano Escenario 1

Realizado por: Freire Kevin, 2022

### 3.2. Escenario 2: Corredor de la planta baja (virtual)

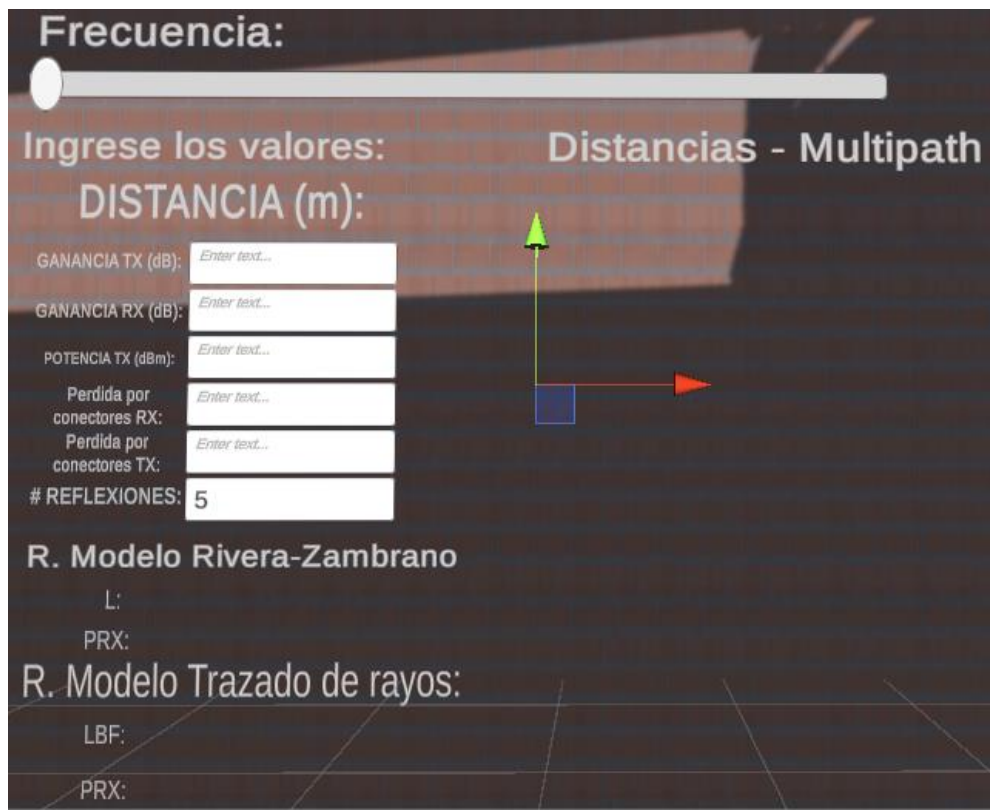
De manera adicional, las pruebas del simulador han sido realizadas en el primer piso del edificio de la Facultad de Informática y Electrónica (FIE), a diferencia de las pruebas realizadas en el Laboratorio de Microondas, los escenarios planteados están contemplados a distancias de entre 7 a 12 m en línea de vista directa, como se evidencia en la Figura 5-3 presentada a continuación:



**Figura 5-3** Propagación de Rayos Escenario 2

Realizado por: Freire Kevin, 2022

Así mismo, los parámetros de configuración y los resultados de las potencias son mostradas en el elemento *Canvas* de *Unity*, como se evidencia en la Figura 6-3:



**Figura 6-3:** Canvas Escenario 2

Realizado por: Freire Kevin, 2022

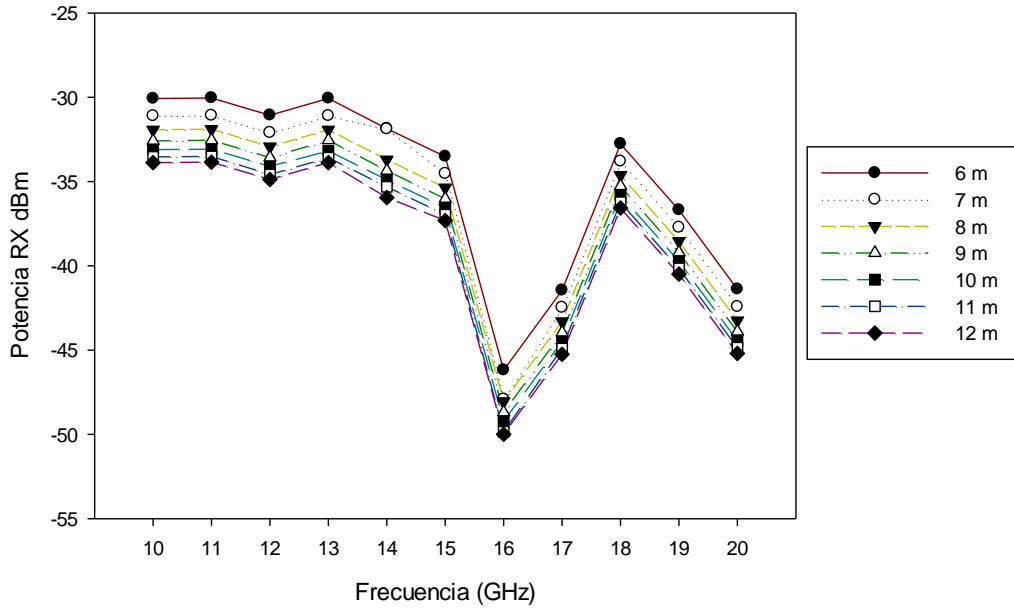
Una vez realizadas las pruebas en las distancias planteadas, se procedió con la obtención de las potencias recibidas en cada uno de los escenarios propuestos. En la Tabla 3-3 se aprecian los valores del modelo de Trazado de Rayos:

**Tabla 3-3:** Valores PRx (dBm) en Modelo de Propagación Trazado de Rayos Escenario 2

F(Ghz)/D(m)	6	7	8	9	10	11	12
10	-30,08	-31,11	-31,93	-32,59	-33,11	-33,54	-33,88
11	-30,04	-31,08	-31,9	-32,55	-33,08	-33,51	-33,85
12	-31,07	-32,1	-32,93	-33,58	-34,11	-34,59	-34,88
13	-30,07	-31,1	-31,93	-32,55	-33,18	-33,59	-33,88
14	-31,86	-31,89	-33,71	-34,36	-34,9	-35,32	-35,95
15	-33,51	-34,54	-35,37	-36,03	-36,55	-36,97	-37,32
16	-46,19	-47,92	-48,04	-48,7	-49,23	-49,85	-49,99
17	-41,45	-42,48	-43,3	-43,96	-44,49	-44,91	-45,25
18	-32,77	-33,8	-34,62	-35,27	-35,8	-36,23	-36,57
19	-36,69	-37,72	-38,54	-39,2	-39,73	-40,15	-40,49
20	-41,39	-42,43	-43,25	-43,9	-44,44	-44,86	-45,2

Realizado por: Freire Kevin, 2022

En la Figura 7-3 se aprecia resumen de los resultados obtenidos:



**Figura 7-3:** Diagrama PRx vs Frecuencia en Modelo Trazado de Rayos Escenario 2

Realizado por: Freire Kevin, 2022

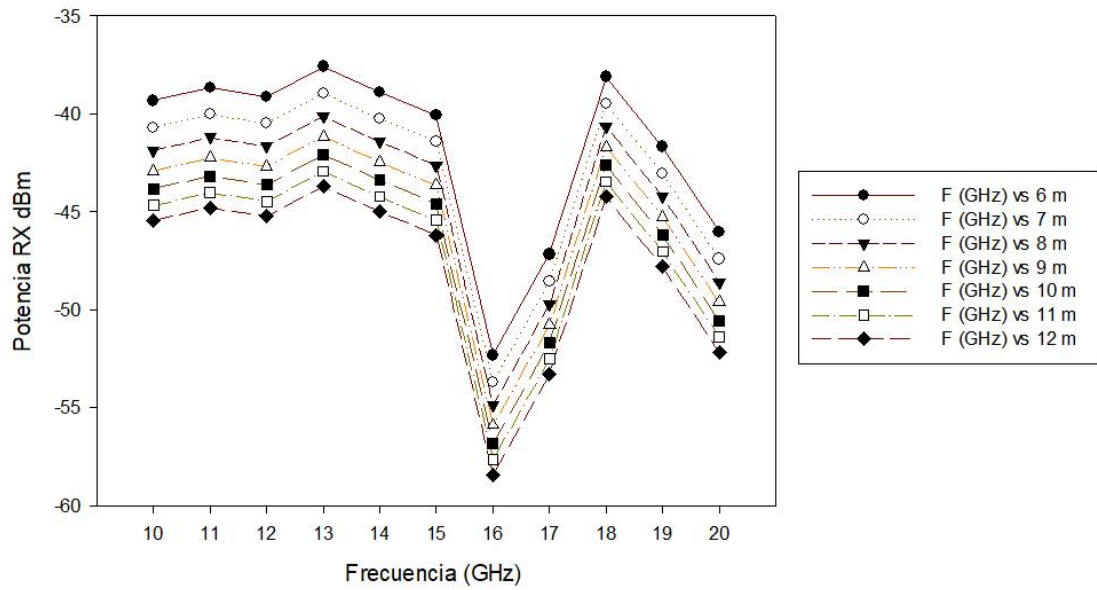
En la Tabla 4-3 se aprecian los resultados obtenidos, utilizando el Modelo de propagación Rivera-Zambrano:

**Tabla 4-3:** Valores PRx (dBm) en Modelo de Propagación Rivera-Zambrano Escenario 2

F(GHz)/D(m)	6	7	8	9	10	11	12
10	-39,33	-40,69	-41,87	-42,91	-43,83	-44,68	-45,45
11	-38,67	-40,03	-41,2	-42,24	-43,17	-44,01	-44,78
12	-39,12	-40,48	-41,65	-42,69	-43,62	-44,46	-45,23
13	-37,58	-38,94	-40,12	-41,16	-42,09	-42,93	-43,7
14	-38,88	-40,24	-41,41	-42,45	-43,38	-44,22	-44,99
15	-40,07	-41,43	-42,61	-43,65	-44,58	-45,42	-46,19
16	-52,32	-53,68	-54,86	-55,9	-56,82	-57,67	-58,43
17	-47,17	-48,53	-49,71	-50,75	-51,68	-52,52	-53,28
18	-38,11	-39,47	-40,65	-41,68	-42,61	-43,46	-44,23
19	-41,68	-43,04	-44,21	-45,25	-46,18	-47,02	-47,8
20	-46,04	-47,4	-48,58	-49,61	-50,55	-51,38	-52,16

Realizado por: Freire Kevin, 2022

En la Figura 8-3 se muestran los resultados obtenidos con el Modelo Rivera-Zambrano, del Escenario 2.



**Figura 8-3:** Diagrama PRx vs Frecuencia en Modelo Rivera-Zambrano Escenario 2

Realizado por: Freire Kevin, 2022

### 3.3. Escenario 3: Pruebas reales en laboratorio de microondas

De igual forma que los escenarios simulados, se procedió con la obtención de los resultados de la potencia recibida en las antenas receptoras con frecuencias que varían entre 10 GHz - 20 GHz, y con las mismas distancias del Escenario 1, los valores se los obtuvo a través del analizador de espectros, como se visualiza en la Figura 9-3:



**Figura 9-3:** Mediciones con analizador de espectro Escenario 3

Realizado por: Freire Kevin, 2022



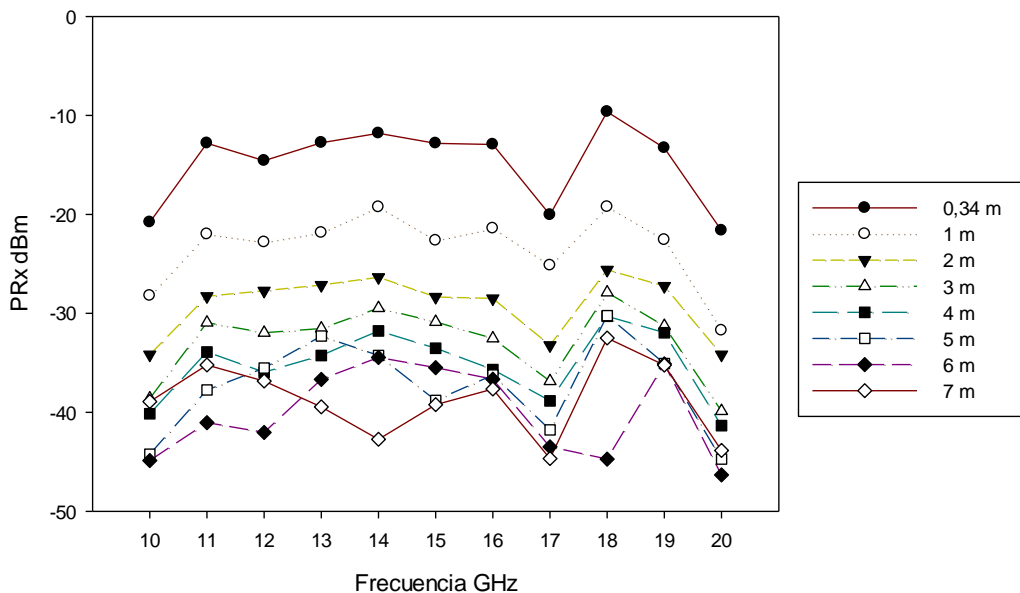
Los resultados de las potencias de recepción en el Laboratorio de Microondas se presentan en la Tabla 5-3, mostrada a continuación:

**Tabla 5-3:** Valores PRx (dBm) reales en el Laboratorio de Microondas Escenario 3

F(Ghz)/D(m)	0,34	1	2	3	4	5	6	7
10	-20,8	-28,22	-34,18	-38,61	-40,14	-44,24	-44,87	-38,92
11	-12,81	-21,99	-28,28	-30,94	-33,9	-37,74	-41,05	-35,22
12	-14,57	-22,81	-27,73	-31,97	-35,95	-35,53	-42,04	-36,84
13	-12,77	-21,85	-27,15	-31,54	-34,26	-32,29	-36,67	-39,45
14	-11,81	-19,25	-26,36	-29,48	-31,77	-34,26	-34,46	-42,73
15	-12,82	-22,64	-28,34	-30,87	-33,52	-38,8	-35,48	-39,23
16	-12,93	-21,41	-28,5	-32,54	-35,68	-36,23	-36,66	-37,64
17	-20,04	-25,16	-33,23	-36,85	-38,84	-41,78	-43,49	-44,67
18	-9,63	-19,24	-25,61	-27,89	-30,29	-30,23	-44,73	-32,49
19	-13,28	-22,57	-27,24	-31,28	-31,96	-35,08	-35,11	-35,23
20	-21,63	-31,73	-34,17	-39,88	-41,34	-44,73	-46,32	-43,86

Realizado por: Freire Kevin, 2022

En la Figura 10-3 se presenta la gráfica resultante de los valores obtenidos de las mediciones en el laboratorio de microondas:



**Figura 10-3:** Diagrama PRx vs Frecuencia en resultados obtenidos Escenario 3

Realizado por: Freire Kevin, 2022

Como se puede apreciar en la Figura 10-3 el valor de la potencia de recepción posee una constante variación con respecto a la distancia. Sin embargo, estos valores no son proporcionales a una mayor o menor frecuencia, a diferencia del valor de la distancia. Por ejemplo, a 18 Ghz se tiene

la potencia de recepción más alta, a una distancia de 0,34m; mientras que, el valor mínimo de la potencia está en 20 GHz, pero a una distancia de 6m dentro del laboratorio. Esto se debe al efecto de las múltiples contribuciones debido a reflexiones en el entorno *indoor*.

### 3.4. Escenario 4: Pruebas reales en planta baja

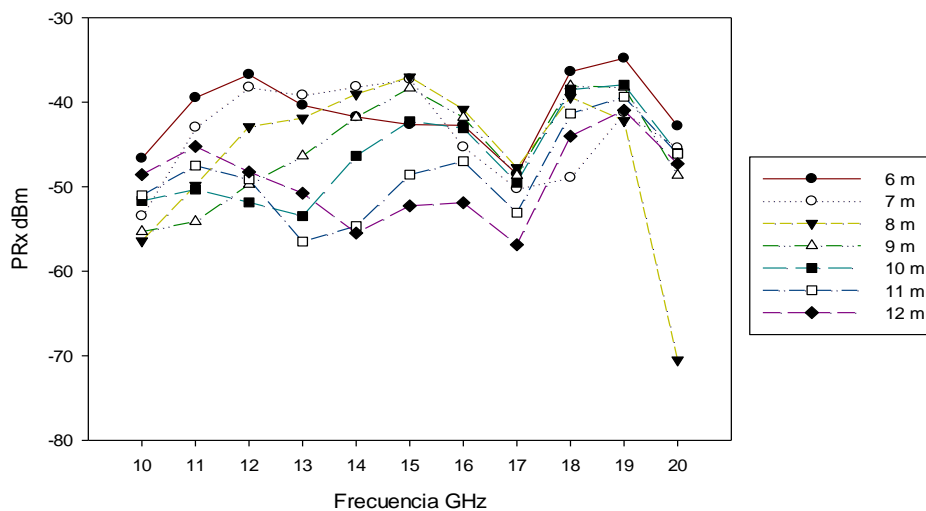
De acuerdo con los escenarios propuestos en la investigación, las pruebas del Escenario 4 fueron realizadas en el primero piso del edificio de la FIE, con distancias de 6 a 12 metros en línea de vista directa, los resultados se aprecian en la Tabla 6-3:

**Tabla 6-3:** Valores PRx (dBm) reales en Planta Baja Escenario 4

F(GHz)/D(m)	6	7	8	9	10	11	12
10	-46,64	-53,46	-56,41	-53,12	-51,68	-51,02	-48,56
11	-39,48	-42,98	-49,83	-54,13	-50,33	-47,51	-45,24
12	-36,74	-38,24	-42,91	-49,72	-51,86	-49,13	-48,25
13	-40,36	-39,16	-41,89	-46,36	-53,47	-56,49	-50,76
14	-41,73	-38,17	-39,05	-41,79	-46,35	-54,66	-55,48
15	-42,65	-37,28	-37,02	-38,33	-42,25	-48,58	-52,26
16	-42,74	-45,31	-40,84	-41,81	-43,08	-46,98	-51,89
17	-48,51	-50,23	-47,75	-48,68	-49,53	-53,08	-56,86
18	-36,38	-48,92	-39,42	-38,08	-38,51	-41,36	-44,02
19	-34,81	-41,23	-42,19	-38,24	-37,94	-39,36	-40,96
20	-42,82	-45,42	-70,49	-48,65	-46,02	-46,11	-47,28

Realizado por: Freire Kevin, 2022

En la Figura 11-3 se presenta los resultados reales medidos en el Escenario 4.



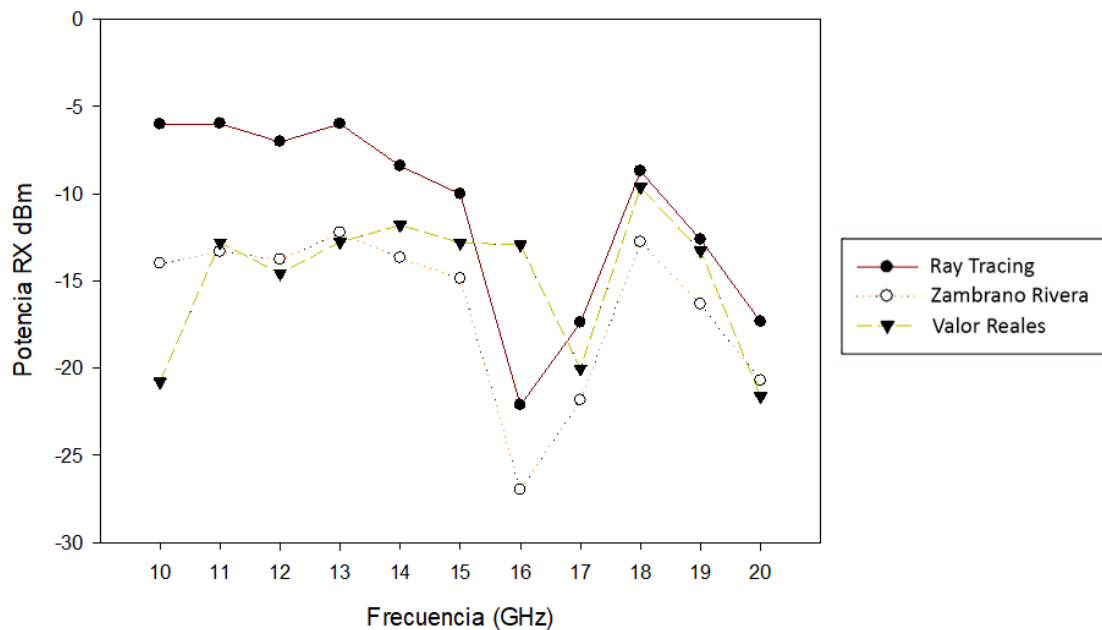
**Figura 11-3:** Diagrama PRx vs Frecuencia en resultados obtenidos Escenario 4

Realizado por: Freire Kevin, 2022

De acuerdo con el estudio planteado, en el Escenario 4 se puede apreciar que efectivamente la distancia, es un parámetro relacionado con el incremento de la potencia, puesto que, los valores obtenidos en el primer piso varían desde -34 dBm hasta los -56 dBm a excepción del valor conseguido a 8 metros de distancia y a 20 GHz de frecuencia que fue de -70,49 dBm. Estos resultados difieren de los logrados en el laboratorio de microondas, debido a que el entorno de medida *indoor* es un entorno más abierto, con mayor distancia para las posibles reflexiones.

### 3.5. Análisis global de resultados: Laboratorio de Microondas

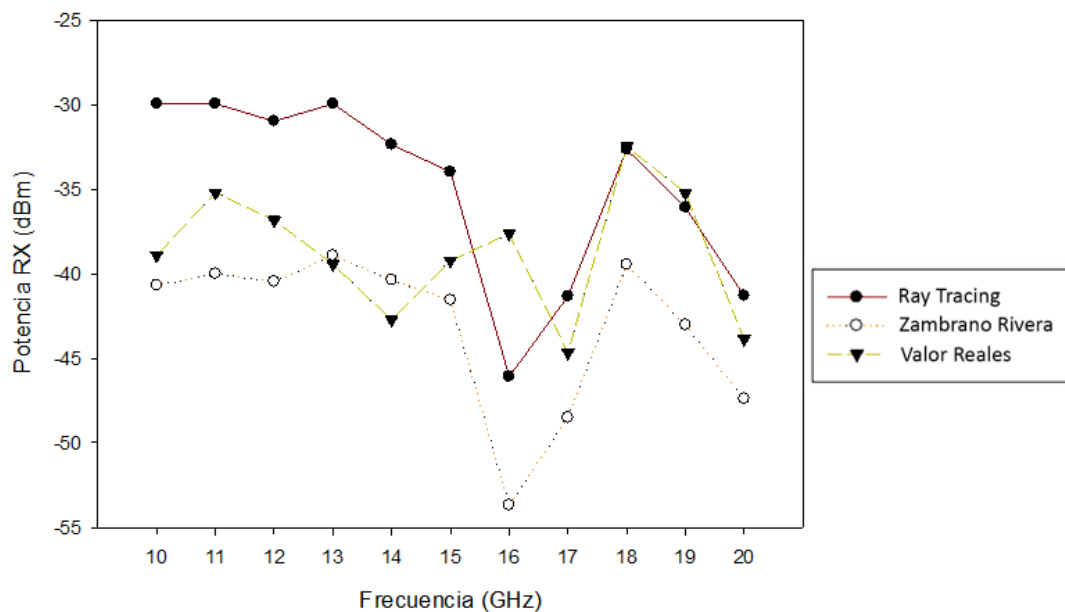
El análisis global se detalla en la Figura 12-3, donde se aprecia una variación entre los valores obtenidos de cada una de las propagaciones simuladas y reales a una distancia de 0,34 metros. Podemos observar entre Trazado de Rayos vs Valores reales, a excepción de a 10 GHz en la distancia más corta, que la variación de potencia máxima que se obtiene bordea los 8 dBm. Además, tiene una diferencia máxima de aproximadamente 8 dBm entre los Modelos de Trazado de Rayos y Rivera-Zambrano. Se puede notar además que el modelo de propagación Rivera-Zambrano se aproxima mucho más a los valores reales medidos.



**Figura 12-3:** Comparación PRx vs Frecuencia (Distancia más corta) Lab. Microondas

Realizado por: Freire Kevin, 2022

Por otra parte, también se muestra la comparativa de los resultados simulados y reales de las pruebas de propagación dentro del Laboratorio de Microondas a la mayor distancia que, es equivalente a 7 m, los resultados se aprecian en la Figura 13-3.



**Figura 13-3:** Comparación PRx vs Frecuencia (Distancia larga) Lab. Microondas

Realizado por: Freire Kevin, 2022

Como se evidencia en la Figura 13-3, la diferencia máxima de valores entre Modelo Trazado de Rayos y los cálculos reales ronda los 10 dBm, valor similar al que existe de diferencia entre los modelos de Trazado de Rayos y Rivera-Zambrano. Por otra parte, en la Tabla 7-3 se muestra el Error Cuadrático Medio RMSE de los resultados que, como se muestra en la Ecuación 3-2 depende la variable  $e$  definida como la diferencia entre los valores reales y los valores simulados:

**Tabla 7-3:** Error Cuadrático Medio entre valores reales y simulados (Lab. Microondas)

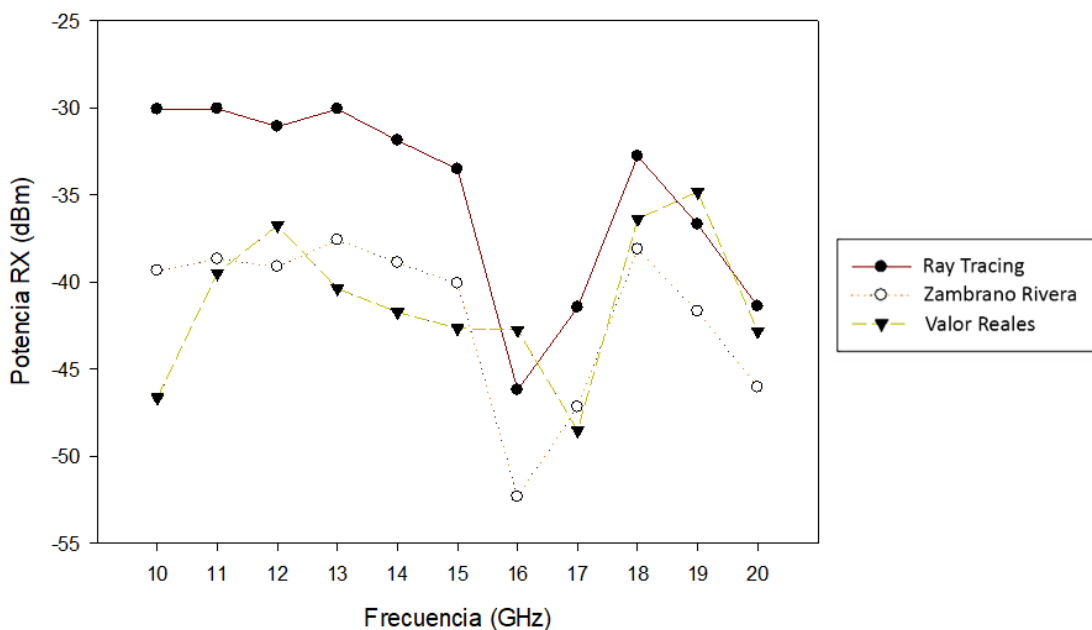
F(GHz)	Distancia 0,34 m				Distancia 7 m			
	Rivera-Zambrano		Trazado de Rayos		Rivera-Zambrano		Trazado de rayos	
	$e$	$e^2$	$e$	$e^2$	$e$	$e^2$	$e$	$e^2$
10	6,79	46,1041	14,77	218,1529	1,77	3,1329	8,95	80,1025
11	0,53	0,2809	6,82	46,5124	4,8	23,04	5,27	27,7729
12	0,78	0,6084	7,54	56,8516	3,63	13,1769	5,86	34,3396
13	0,52	0,2704	6,76	45,6976	0,52	0,2704	9,49	90,0601
14	1,88	3,5344	3,40	11,56	2,36	5,5696	10,37	107,5369
15	2,06	4,2436	2,79	7,7841	2,33	5,4289	5,25	27,5625
16	14,06	197,6836	9,21	84,8241	16,04	257,2816	8,45	71,4025
17	1,81	3,2761	2,64	6,9696	3,86	14,8996	3,32	11,0224
18	3,15	9,9225	0,91	0,8281	6,98	48,7204	0,18	0,0324
19	3,06	9,3636	0,64	0,4096	7,8	60,84	0,86	0,7396
20	0,91	0,8281	4,28	18,3184	3,54	12,5316	2,56	6,5536
	<b>RMSE=</b>	<b>5,0101</b>	<b>RMSE=</b>	<b>6,7279</b>	<b>RMSE=</b>	<b>6,3596</b>	<b>RMSE=</b>	<b>6,4465</b>

Realizado por: Freire Kevin, 2022

De acuerdo la tabla presentada se evidencia que el Modelo Rivera-Zambrano posee el RMSE más bajo tanto en la distancia más corta como en la más larga, lo cual indica que, dentro del escenario analizado, dicho modelo se aproxima más a los valores reales.

### 3.6. Análisis global de resultados planta baja FIE

Las pruebas realizadas en el primer piso del edificio de la FIE, corroboran la variación entre los valores obtenidos en el simulador, con los modelos de propagación Trazado de Rayos y Rivera-Zambrano frente a las potencias resultantes de los escenarios reales. Dicha comparativa analizada en la distancia más corta de este escenario equivalente a 6 metros, se muestra en el Figura 14-3:

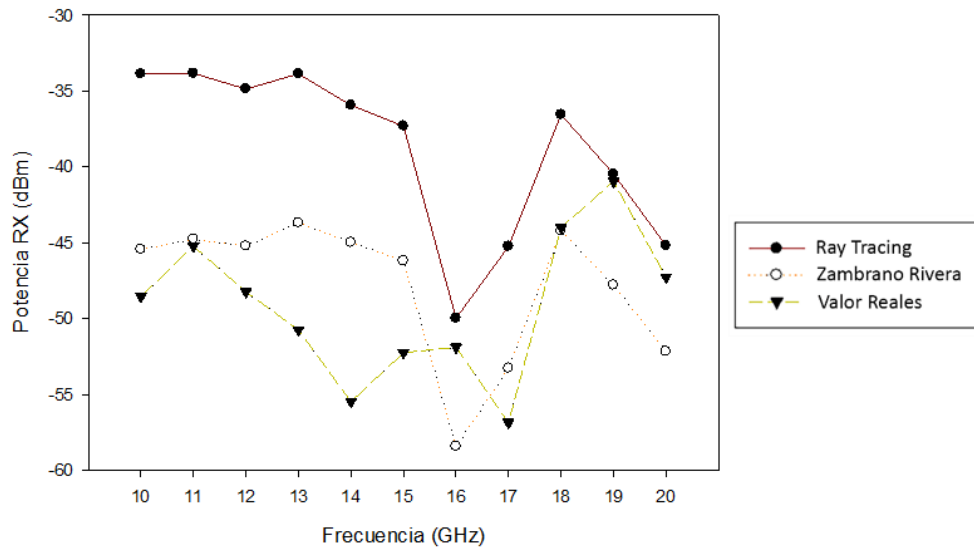


**Figura 14-3:** Comparación PRx vs Frecuencia (Distancia más corta) Planta baja

Realizado por: Freire Kevin, 2022

Como se aprecia en la figura 14-3, las variaciones en los resultados son aproximados a los obtenidos en el Laboratorio de Microondas; los valores simulados con los reales tienen alrededor de 10 dBm de diferencia a excepción del valor a 10 GHz, mismo valor de diferencia que se tiene entre los modelos simulados Trazado de Rayos y Rivera-Zambrano. Por otro lado, estos valores tienden a disminuir cuando se analiza la propagación a mayor frecuencia.

En el Figura 15-3 se muestra la variación de resultados a 12 metros de distancia entre el transmisor y receptor:



**Figura 15-3:** Comparación PRx vs Frecuencia (Distancia más larga) Planta baja

Realizado por: Freire Kevin, 2022

Al ser un escenario más abierto se obtiene mayor variación de potencias que en el escenario 1, variación que podemos ver que llega incluso hasta los 20 dBm estando a 12m y 14 GHz. La máxima diferencia de potencias entre los modelos de Trazado de Rayos y Rivera-Zambrano varía alrededor de los 10-12 dBm, estos resultados se mantienen constantes en cada una de las distancias estudiadas en los dos escenarios. Por tal razón, solo se ha considerado las distancias más cortas y las más largas, para el análisis de la variación de potencias, en la Tabla 8-3 se evidencia el RMSE de los valores obtenidos en la Planta Baja del Edificio, que de igual manera depende de la diferencia entre valores reales y simulados conocido como error aparente “e”.

**Tabla 8-3:** Error Cuadrático Medio entre valores reales y simulados (Planta Baja FIE)

F(GHz)	Distancia 6 m				Distancia 12 m			
	Rivera-Zambrano		Trazado de Rayos		Rivera-Zambrano		Trazado de rayos	
	e	e <sup>2</sup>	e	e <sup>2</sup>	e	e <sup>2</sup>	e	e <sup>2</sup>
10	7,31	53,4361	16,56	274,2336	3,11	9,6721	14,68	215,5024
11	0,81	0,6561	9,44	89,1136	0,46	0,2116	11,39	129,7321
12	2,38	5,6644	5,67	32,1489	3,02	9,1204	13,37	178,7569
13	2,78	7,7284	10,29	105,8841	7,06	49,8436	16,88	284,9344
14	2,85	8,1225	9,87	97,4169	10,49	110,0401	19,53	381,4209
15	2,58	6,6564	9,14	83,5396	6,07	36,8449	14,94	223,2036
16	9,58	91,7764	3,45	11,9025	6,54	42,7716	1,9	3,61
17	1,34	1,7956	7,06	49,8436	3,58	12,8164	11,61	134,7921
18	1,73	2,9929	3,61	13,0321	0,21	0,0441	7,45	55,5025
19	6,87	47,1969	1,88	3,5344	6,84	46,7856	0,47	0,2209
20	3,22	10,3684	1,43	2,0449	4,88	23,8144	2,08	4,3264
	<b>RMSE=</b>	<b>4,6358</b>	<b>RMSE=</b>	<b>8,3268</b>	<b>RMSE=</b>	<b>5,5756</b>	<b>RMSE=</b>	<b>12,1056</b>

Realizado por: Freire Kevin, 2022

El error cuadrático medio evidencia que de igual manera que el anterior escenario, los valores del modelo Rivera-Zambrano se acercan más a los valores reales en comparación con el modelo Trazado de Rayos.

## CONCLUSIONES

Se concluye que:

- A pesar de que la librería OPAL representa una herramienta con varias ventajas para la propagación de rayos, para nuestro aplicativo el *software* Unity posee bibliotecas propias que abrió la opción a realizar las simulaciones propuestas sin necesidad de ocupar dicha librería.
- Se definió los parámetros de entrada: potencia de transmisión, pérdidas por cables y conectores, distancia, ganancia de receptor y transmisor, número de reflexiones, frecuencia de operación, y parámetros de salida del simulador: pérdidas de propagación, potencia de recepción y distancia-multipath.
- En base al análisis y comparación de los valores de potencia obtenidos en mediciones reales como dentro del simulador mediante los modelos Trazado de Rayos y Rivera-Zambrano, se valida este último dentro de nuestro aplicativo.
- Las mediciones de potencia simuladas en los escenarios propuestos, permitió la comparación de resultados entre los modelos de Trazado de Rayos y el modelo Rivera Zambrano, se evidenció que, los resultados difieren entre los ambos modelos, sin embargo, la diferencia de los valores en la simulación bordea los 10 dBm en el escenario 1 (Laboratorio de Microondas) y llegando a bordar hasta los 12 dBm en el escenario 2 (Planta Baja Edificio FIE).
- La evaluación del desempeño del simulador frente a las mediciones reales, muestran variación en los resultados por cada uno de los modelos utilizados; gracias al resultado obtenido del RMSE, se concluye que el modelo Rivera-Zambrano se ajusta más a los valores reales de potencia, evidenciándose un margen de error de 5,0101 en el primer escenario para la distancia más corta y de 6,3596 para la distancia más larga, mientras que, en el segundo escenario los valores del RMSE son de 4,6358 para el menor trayecto y de 5,5756 para el mayor, lo cual equivale a un error aceptable.



## RECOMENDACIONES

Se recomienda:

- Tener en consideración las limitantes que presenta *Optix*, con referencia a la compatibilidad de *Hardware*, puesto que, al ser un *software* propietario de Nvidia, limita el uso en equipos con dispositivos de diferentes fabricantes o tarjetas descontinuadas.
- Realizar simulaciones en diversos escenarios que permitan analizar el efecto de los rayos transmitidos y difractados, lo cual depende de ciertos parámetros como la densidad del material o la permeabilidad.
- Definir escenarios propuestos dentro del edificio, mismos que no sean en línea de vista directa, para poder validar los resultados obtenidos en las potencias reales, frente a los que se podrían obtener en el simulador en *Unity*.
- Investigar parámetros o configuraciones que facilite las mejoras de la simulación realizada, esto permitiría obtener un simulador más didáctico, con múltiples prestaciones y configuraciones, de modo que servirá para mejorar el aprendizaje académico.

## **GLOSARIO**

**Difracción:** Fenómeno producido cuando un frente de onda se dobla sobre los extremos de un obstáculo con tamaño similar al de la longitud de onda, al impactar contra él.

**Empírico:** Basado en la práctica, experiencia y observación de hechos.

**Espectro radioeléctrico:** Medio por el cual se transmiten las frecuencias de ondas de radio electromagnéticas que permiten las telecomunicaciones

**Free Space:** Propagación de ondas en condiciones de espacio libre

**GPU:** Unidad de Procesamiento Gráfico

**LOS:** Línea de visión directa

**mmWave:** Ondas milimétricas

**Multipath:** Efecto de propagación dado cuando las señales de radio llegan a la antena o antenas receptoras por dos o más caminos y en diferentes tiempos.

**NLOS:** Línea de visión obstruida

**Pixel:** Unidad más pequeña homogénea en color que compone la imagen digital.

**Reflexión:** Rebote o cambio de dirección de una onda cuando ésta choca contra un material que, en función de su densidad, reflejará una parte de la señal y absorberá otra.

**RMSE:** Error cuadrático medio

## BIBLIOGRAFÍA

**AEPD.** *Introducción a las tecnologías 5G y sus riesgos para la privacidad.* [En línea]. 2020 [Consulta: 2021-06-08]. Disponible en: <https://www.aepd.es/sites/default/files/2020-06/nota-tecnica-privacidad-5g.pdf>

**BALLESTER, Manuel.** *Aplicación del modelo de propagación Multi-Wall para la estimación de cobertura de femtoceldas LTE en interiores* [En línea] (Tesis de pregrado). Universitat Politècnica de València, Valencia- España. 2013. pp. 1-56. [Consulta: 2021-05-12]. Disponible en:

[https://riunet.upv.es/bitstream/handle/10251/33006/Ballester\\_Lidon\\_Manuel.pdf?sequence=1](https://riunet.upv.es/bitstream/handle/10251/33006/Ballester_Lidon_Manuel.pdf?sequence=1)

**BETRACK.** *Ciclo de Deming: ejemplos, etapas, importancia, ventajas y desventajas.* [En línea]. 2021 [Consulta: 2021-10-27]. Disponible en: <https://www.beetrack.com/es/blog/ciclo-de-deming-etapas-ejemplos>

**BLÁZQUEZ, Alejandro., CASADO, Carlos. & PALACIOS, Juan.** *Producción de un Videojuego en Realidad Virtual con Seguimiento de Movimiento en Visor y Controladores Manuales utilizando Metodología Ágil* [En línea] (Tesis de pregrado). Universidad Complutense de Madrid, Madrid- España. 2017. pp. 1-125. [Consulta: 2021-06-08]. Disponible en: <https://eprints.ucm.es/id/eprint/44686/1/Memoriafinal.pdf>

**CAMPILLO, Valeriano.** *Programación de una herramienta de trazado de rayos 3D basada en GO y UTD* [En línea] (Tesis de pregrado). Universidad Politécnica de Cartagena, Cartagena-Colombia. 2008. pp. 1-147. [Consulta: 2021-05-04]. Disponible en: <https://repositorio.upct.es/bitstream/handle/10317/560/pfc2658.pdf?sequence=1&isAllowed=y>

**CAMPOS, Daniel.** *Cobertura dentro de construcciones para Comunicaciones Personales Móviles Vía Satélite* [En línea] (Tesis de pregrado). Universidad de las Américas Puebla, San Andrés Cholula, Puebla- México. 2004. pp. 1-133. [Consulta: 2021-05-12]. Disponible en: [http://catarina.udlap.mx/u\\_dl\\_a/tales/documentos/lem/campos\\_v\\_da/](http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/campos_v_da/)

**CASTAÑO, Arturo.** *Unidad VII: Ondas electromagnéticas.* [En línea]. 2008 [Consulta: 2021-05-12]. Disponible en: <http://ing.unne.edu.ar/pub/fisica3/170308/teo/teo7.pdf>

**CASTELLANOS, Everaldo., TALERO, José., RUGELES, José. & ORTEGA, Homero.** *Análisis de propagación electromagnética en espacios cerrados: herramienta software en matlab para predicción y simulación.* *Revista Colombiana de Tecnologías de Avanzada* [en línea], 2005, (Colombia) 2 (6), pp. 130-137. [Consulta: 2021-05-26]. ISSN: 1692-7257. Disponible en: [https://www.unipamplona.edu.co/unipamplona/portalIG/home\\_40/recursos/01\\_general/revista\\_6/13102011/21.pdf](https://www.unipamplona.edu.co/unipamplona/portalIG/home_40/recursos/01_general/revista_6/13102011/21.pdf)

**CEREZUELA, Francisco.** *Estudio de la difracción múltiple a 60GHz* [En línea] (Tesis de pregrado). Universidad Politécnica de Cartagena, Cartagena, España. 2016. pp. 1-50. [Consulta:

2021-05-12]. Disponible en: <https://repositorio.upct.es/xmlui/bitstream/handle/10317/5746/pfc-cer-est.pdf?sequence=1&isAllowed=y>

**CRESPO, Rodolfo.** *Etapas del proceso de la investigación.* [En línea]. 2006 [Consulta: 2021-10-27]. Disponible en: <https://www.revistasden.org/files/2-CAP%202.pdf>

**DAVIES, Ron.** *Radio spectrum A key resource for the Digital Single Market.* [En línea]. 2015 [Consulta: 2021-05-12]. Disponible en: <https://www.europarl.europa.eu/EPRS/EPRS-Briefing-554170-Radio-Spectrum-FINAL.pdf>

**DURÁN, Alejandro.** *Estudio y caracterización del canal y de la propagación en ondas milimétricas, orientada a su utilización en redes de comunicaciones móviles 5G* [En línea] (Tesis de posgrado). Universidad Politécnica de Madrid, Madrid- España. 2017. pp. 1- 153. [Consulta: 2021-05-12]. Disponible en:

[https://oa.upm.es/47318/1/TESIS\\_MASTER\\_ALEJANDRO\\_DURAN\\_BARRADO\\_2017.pdf](https://oa.upm.es/47318/1/TESIS_MASTER_ALEJANDRO_DURAN_BARRADO_2017.pdf)

**EGEA, Esteban.** *Opal.* [En línea]. 2021 [Consulta: 2021-06-08]. Disponible en: <https://gitlab.com/esteban.egea/opal>

**ESPINOSA, Oscar.** *Qué es el espectro radioeléctrico.* [En línea]. 2021 [Consulta: 2021-05-12]. Disponible en: <https://www.redeszone.net/reportajes/tecnologias/que-es-espectro-radioelectrico/>

**MADROÑERO, María. & MENESES, Sandra.** *Análisis de la propagación de una señal mmWave en entornos indoor* [En línea] (Tesis de pregrado). Universidad del Cauca, Popayán- Colombia. 2016. pp. 1- 91. [Consulta: 2021-05-12]. Disponible en: <http://repositorio.unicauca.edu.co:8080/bitstream/handle/123456789/1540/AN%C3%81LISIS%20DE%20LA%20PROPAGACI%C3%93N%20DE%20UNA%20SE%C3%91AL%20MMWAVE%20EN%20ENTORNOS%20INDOOR.pdf?sequence=1&isAllowed=y>

**MEILING, Luo.** *Indoor radio propagation modeling for system performance prediction* [En línea] (Tesis de pregrado). L'Institut National des Sciences Appliquées de Lyon, Villeurbanne- Francia. 2013. pp. 1-126. [Consulta: 2021-05-26]. Disponible en: <https://tel.archives-ouvertes.fr/tel-00937481/document>

**MENDES, Dayana.** *Caracterización indoor y ampliación de un rem (radio environment map) en el campus nord de la UPC* [En línea] (Tesis de pregrado). Universidad Politécnica de Catalunya, Barcelona- España. 2015. pp. 1-99-. [Consulta: 2021-05-26]. Disponible en: [https://upcommons.upc.edu/bitstream/handle/2099.1/25617/PFC\\_Dayana%20Mendes%20Definitiva.pdf](https://upcommons.upc.edu/bitstream/handle/2099.1/25617/PFC_Dayana%20Mendes%20Definitiva.pdf)

**NAVARRO, Andrés., GUEVARA, Dinael. & AFRICANO, María.** Calibración Basada en Medidas para Modelos de Trazado de Rayos en 3D para Ambientes Exteriores Urbanos Andinos. *Revista S&T* [en línea], 2012, (Colombia) 10 (21), pp. 43-63. [Consulta: 2021-05-04]. Disponible en:

[https://www.researchgate.net/publication/277222181\\_Calibracion\\_Basada\\_en\\_Medidas\\_para\\_Modelos\\_de\\_Trazado\\_de\\_Rayos\\_en\\_3D\\_para\\_Ambientes\\_Exteriores\\_Urbanos\\_andinos](https://www.researchgate.net/publication/277222181_Calibracion_Basada_en_Medidas_para_Modelos_de_Trazado_de_Rayos_en_3D_para_Ambientes_Exteriores_Urbanos_andinos)  
**NELIMARKKA, Paavo.** *Teaching unity3D in game programming module* [En línea] (Tesis de pregrado). Jyväskylä University of Applied Sciences, Jyväskylä- Finlandia. 2014. pp. 1-104. [Consulta: 2021-06-08]. Disponible en: [https://www.theseus.fi/bitstream/handle/10024/75534/bachelors\\_thesis\\_paavo\\_nelimarkka.pdf?sequence=1&isAllowed=y](https://www.theseus.fi/bitstream/handle/10024/75534/bachelors_thesis_paavo_nelimarkka.pdf?sequence=1&isAllowed=y)

**PARKER, Steven., FRIEDRICH, Heiko., LUEBKE, David., MORLEY, Keith., BIGLER, James., HOBEROCK, Jared., MCALLISTER, David., ROBISON, Austin., DIETRICH, Andreas., HUMPHREYS, Greg., MCGUIRE, Morgan. & STICH, Martin.** GPU Ray Tracing. *Communications of the ACM* [en línea], 2013, (Estados Unidos) 56 (05), pp. 93-101. [Consulta: 2021-05-26]. Disponible en: <https://www.classes.cs.uchicago.edu/archive/2016/winter/32001-1/papers/p93-parker.pdf>

**PARKER, Steven., BIGLER, James., DIETRICH, Andreas., FRIEDRICH, Heiko., HOBEROCK, Jared., LUEBKE, David., MCALLISTER, David., MCGUIRE, Morgan., MORLEY, Keith., ROBISON, Austin., STICH, Martin., NVIDIA. & WILLIAMS COLLEGE.** *ACM Transactions on Graphics* [en línea], 2010, (Estados Unidos) 29 (04), pp. 66:2-66:13. [Consulta: 2021-05-26]. Disponible en: [https://raytracing-docs.nvidia.com/optix6/whitepaper/nvidia\\_optix\\_TOG\\_v29\\_n4.pdf](https://raytracing-docs.nvidia.com/optix6/whitepaper/nvidia_optix_TOG_v29_n4.pdf)

**PENZENTCEV, Aleksei.** *Architecture and implementation of the system for serious games in Unity 3D* [En línea] (Tesis de pregrado). Masarykova Univerzita, Brno, República Chec. 2015. pp. 1- 68. [Consulta: 2021-06-08]. Disponible en: <https://is.muni.cz/th/whosc/Thesis.pdf>

**QUESTIONPRO.** *Métodos de investigación: Qué son y cómo elegirlos.* [En línea]. 2021 [Consulta: 2021-10-27]. Disponible en: <https://www.questionpro.com/blog/es/metodos-de-investigacion/>

**RÁBANOS, J.** “*Transmisión por Radio*”. [En línea], Madrid: Centro De Estudios Ramón Areces S.A, 2013, pp. 75-85. [Consulta: 25 de enero 2022]. Disponible en: <https://books.google.com.co/books?id=eHCnDAAAQBAJ&printsec=copyright#v=onepage&q&f=false>

**RAMÍREZ, Dany. & COLMENARES, Diego.** *Qué es la tecnología 5G, implementación de la red en Colombia y como cambiara nuestras vidas* [En línea] (Tesis de pregrado). Universidad Cooperativa de Colombia, Bogotá- Colombia. 2020. pp. 1- 9. [Consulta: 2021-05-12]. Disponible en: [https://repository.ucc.edu.co/bitstream/20.500.12494/33439/2/2021\\_Tecnologia\\_Implementacion\\_5G.pdf](https://repository.ucc.edu.co/bitstream/20.500.12494/33439/2/2021_Tecnologia_Implementacion_5G.pdf)

**RIVERA, Cristhian. & ZAMBRANO, J.** *Determinación de un modelo de propagación empírico para entornos indoor en la banda de frecuencias de 10 A 20 GHz* [En línea] (Tesis de pregrado). Escuela Superior Politécnica de Chimborazo, Riobamba- Ecuador. 2021. pp. 1- 48. [Consulta: 2021-05-04]. Disponible en:

**VACA, Jorge.** *Análisis y simulación de la propagación de ondas milimétricas en un ambiente indoor* [En línea] (Tesis de pregrado). Escuela Politécnica Nacional., Quito- Ecuador. 2017. pp. 1-110. [Consulta: 2021-05-12]. Disponible en: <https://bibdigital.epn.edu.ec/handle/15000/17422>

**VÁSQUEZ, Denis. & GASPAR, Rosmell.** *Diseño y simulación de un modelo de canal de ondas milimétricas basado en ray-tracing para la tecnología de quinta generación 5G en el centro histórico de la ciudad de Trujillo* [En línea] (Tesis de pregrado). Universidad privada Antenor Orrego, Trujillo- Perú. 2017. pp. 1-105. [Consulta: 2021-05-12]. Disponible en: [https://repositorio.upao.edu.pe/bitstream/20.500.12759/4206/1/RE\\_ING.TELEC.Y.REDES\\_DANIS.V%C3%81SQUEZ\\_ROSMELL.GASPAR\\_SIMULACI%C3%93N.DE.UN.MODELO\\_DATOS.pdf](https://repositorio.upao.edu.pe/bitstream/20.500.12759/4206/1/RE_ING.TELEC.Y.REDES_DANIS.V%C3%81SQUEZ_ROSMELL.GASPAR_SIMULACI%C3%93N.DE.UN.MODELO_DATOS.pdf)

**TIPOSDEINVESTIGACION.** *Investigación descriptiva: qué es, características, ejemplos e imágenes animadas.* [En línea]. 2021 [Consulta: 2021-10-27]. Disponible en:

<https://tiposdeinvestigacion.org/descriptiva/>

**TORRES, Juan., PINTO, Ángel., MACEO, Mario., PÉREZ, Nelson. & RUJANO, Leidy.** *Modelo para la estimación de las pérdidas de propagación en redes wlan operando en 2,4 GHz Y 5,8 GHz, para ambientes interiores de edificios comerciales.* *Universidad, ciencia y tecnología* [en línea], 2016, (Venezuela) 20 (78), pp. 42-53. [Consulta: 2021-05-26]. ISSN: 1692-7257. Disponible en: <http://ve.scielo.org/pdf/uct/v20n78/art04.pdf>

**UIT-R.** *Recomendación UIT-R V.431-8. Nomenclatura de las bandas de frecuencias y de las longitudes de onda empleadas en telecomunicaciones.*

**WILLIAMS, Kathryn.** *Ray Tracing Simulation Tool for Portal-Based Millimeter-Wave Security Systems Using the NVIDIA® OptiX™ Ray Tracing Engine* [En línea] (Tesis de pregrado). Northeastern University Boston, Massachusetts. Boston- Estados Unidos. 2013. pp. 1-76. [Consulta: 2021-05-26]. Disponible en: <https://repository.library.northeastern.edu/files/neu:1250/fulltext.pdf>

**WIRELESSCOMMUNICATION.** *Recepción multitrayecto.* [En línea]. 2021 [Consulta: 2021-05-12]. Disponible en: <http://www.wirelesscommunication.nl/reference/chaptr03/fading/fading.htm>

**YÁNEZ, Paulina.** *Estudio de los canales con desvanecimiento sobre redes fijas y móviles en sistemas de radio comunicación* [En línea] (Tesis de pregrado). Escuela Politécnica Nacional., Quito- Ecuador. 2009. pp. 1-187. [Consulta: 2021-05-12]. Disponible en: <https://bibdigital.epn.edu.ec/bitstream/15000/1455/1/CD-2729.pdf>

**ZHENGQING, Yun. & MAGDY, Iskander.** *Trazado de rayos para el modelado de propagación de radio: principios y aplicaciones* [En línea] (Tesis de pregrado). University of Hawaii at Manoa, Manoa- Hawaii. 2015. pp. 1089- 1100. [Consulta: 2021-05-04]. Disponible en: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7152831>

**UNITY3D.** *Asset Packages (Paquetes de assets)*. [En línea]. 2016 [Consulta: 2021-06-08]. Disponible en: <https://docs.unity3d.com/es/530/Manual/AssetPackages.html>

**LANDA, Nicolás.** *Unity Diseño y Programación de Videojuegos* 1ª ed. Buenos Aires: Fox Andina, 2013, pp. 1- 322.

## ANEXOS

### ANEXO A: DLL de configuración Nvidia Optix con Unity

```
#include <optix/optixpp_namespace.h>
#include <optixu/optixu_math_stream_namespace.h>

#include <sutil.h>
#include "common.h"
#include "random.h"
#include <Arctball.h>

#include <cstdlib>
#include <cstring>
#include <iostream>
#include <sstream>
#include <math.h>
#include <stdint.h>

using namespace optix;

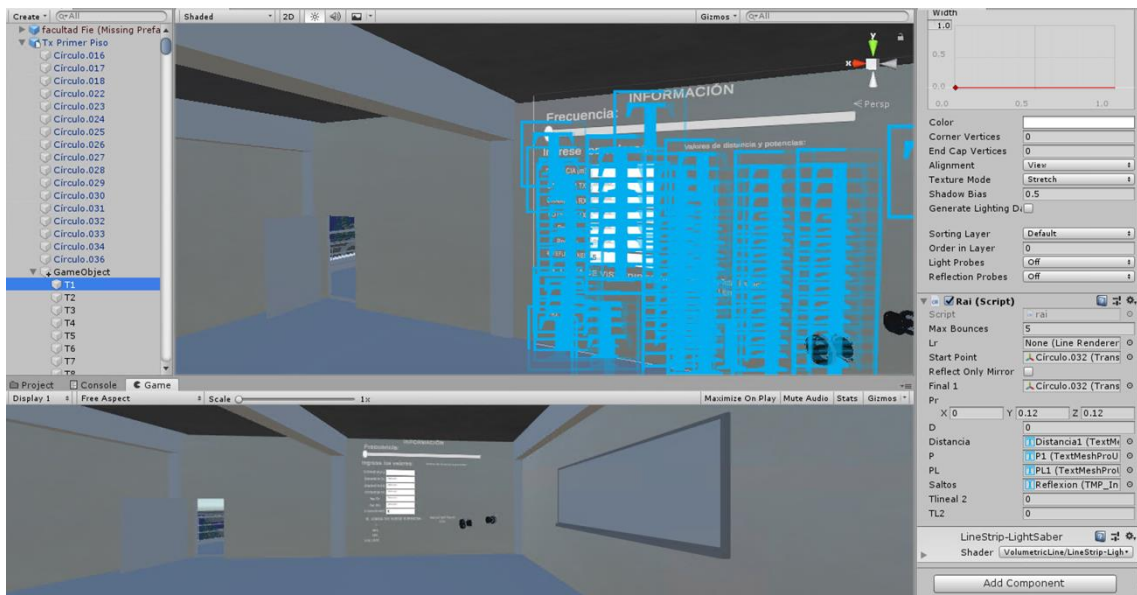
const char* const SAMPLE_NAME = "Optix Rellecciones DLL";

static float rand_range(float min, float max)
{
    static unsigned int seed = 0u;
    return min + (max - min) * rnd(seed);
}

// ...

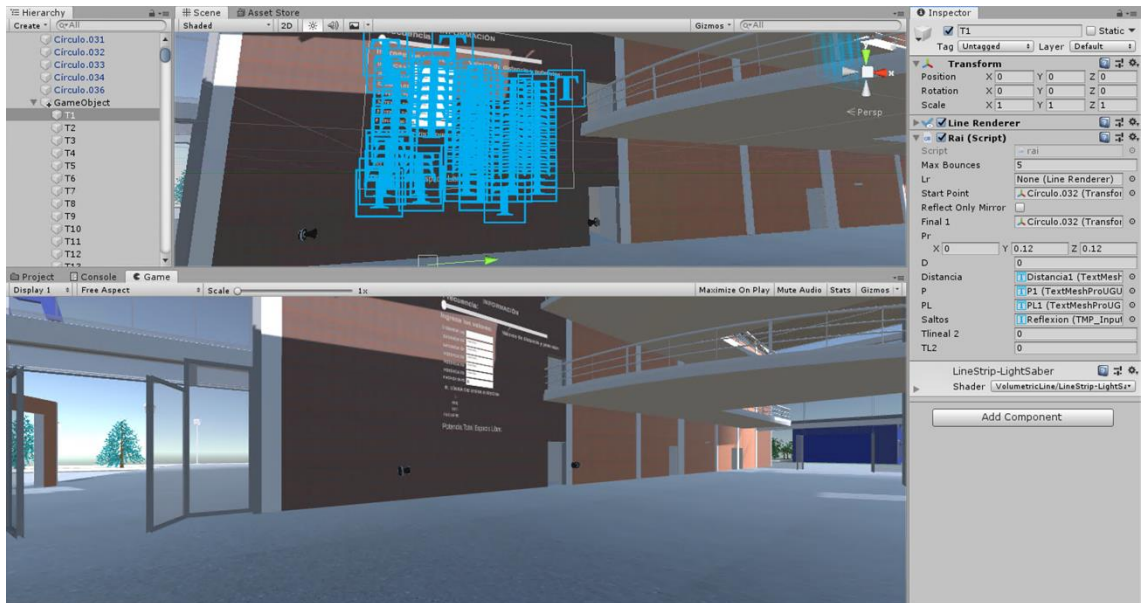
Context      context;
uint32_t     width  = 1080u;
uint32_t     height = 720;
```

### ANEXO B: Parámetros configurados en Unity (Laboratorio de Microondas)





## ANEXO C: Parámetros configurados en Unity (Corredor Planta Baja FIE)



## ANEXO D: Código Rayo Línea Vista Directa

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 0 referencias
6 public class Rai : MonoBehaviour
7 {
8     public Transform Destino;
9     LineRenderer renderer;
10    public List<Vector3> Vectores;
11    0 referencias
12    private void Start()
13    {
14        renderer = GetComponent<LineRenderer>();
15    }
16    0 referencias
17    private void Update()
18    {
19        Vectores.Clear();
20        Vectores.Add(transform.position);
21        Vectores.Add(Destino.position);
22        renderer.SetPositions(Vectores.ToArray());
23    }
24 }
```

## ANEXO E: Código Reflexiones en el Transmisor

```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using UnityEngine.UI;
5 using System;
6 using TMPPro;
7
8
9
10 public class rai : MonoBehaviour
11 {
12
13     public int maxBounces;
14     public LineRenderer lr;
15     public Transform startPoint;
16     public bool reflectOnlyMirror;
17     public Transform final1;
18     public Vector3 pr;
19     public float d;
20     public TextMeshProUGUI Distancia,P, PL;
21     public TMP_InputField Saltos;
22     public double Tlinea12, TL2;
23
24
25     // Start is called before the first frame update
26     void Start()
27     {
28         lr = GetComponent<LineRenderer>();
29         lr.SetPosition(0, startPoint.position);
30         if (Saltos==null)
31     }
```

## ANEXO F: Código de cálculo para Potencias de Recepción

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using System;
using UnityEngine.UI;
using TMPPro;

12 referencias
public class Formula : MonoBehaviour
{
    public static double L, Prx1, Grx = 0, Gtx = 0, Lbf, Prx2, P, Tlinea11, Tlinea12, Psumadas, PTFinal, PTFinal1, TL1, TL2, PsumadasL;
    public static double PerCTX, PerCRX;
    public double distancia = 0;
    public Slider frecuencia;
    public rai[] Potencias;

    [Header("Parametros de Salida")]

    public TextMeshProUGUI Lt;
    public TextMeshProUGUI PrxT, LbfT, Ft, PrxL, PTotal, Ptotal ;
    public Transform TX;
    public Vector3 TXPosicion;

    public static double f, f1;

    [Header("Parametros de entrada")]
    public TMP_InputField GananciaT;
    public TMP_InputField GananciaR;
    public TMP_InputField Distancia;
    public TMP_InputField Potencia;
```