

EVALUACIÓN DEL DESEMPEÑO DE LAS CONDICIONES DE FRONTERA
ABSORBENTES EN LA SIMULACIÓN DE LA PROPAGACIÓN DE ONDAS
ELECTROMAGNÉTICAS 2-D REALIZADA CON EL MÉTODO FDTD.

JESUS ALBERTO LAITÓN SÁNCHEZ
JOHAN SEBASTIAN DÍAZ RODRIGUEZ

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y DE
TELECOMUNICACIONES
BUCARAMANGA

2017

EVALUACIÓN DEL DESEMPEÑO DE LAS CONDICIONES DE FRONTERA
ABSORBENTES EN LA SIMULACIÓN DE LA PROPAGACIÓN DE ONDAS
ELECTROMAGNÉTICAS 2-D REALIZADA CON EL MÉTODO FDTD.

JESUS ALBERTO LAITON SANCHEZ
JOHAN SABASTIAN DIAZ RODRIGUEZ

TRABAJO DE GRADO PARA OPTAR AL TÍTULO DE INGENIEROS
ELECTRÓNICOS

DIRECTOR
ERNESTO AGUILERA BERMÚDEZ
DOCTOR EN INGENIERÍA ELECTRÓNICA

CODIRECTOR
IVÁN JAVIER SÁNCHEZ GALVIS
ESTUDIANTE DE MAESTRÍA EN INGENIERÍA ELECTRÓNICA

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO MECÁNICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y DE
TELECOMUNICACIONES
BUCARAMANGA

2017

DEDICATORIA

Gracias a Dios que me permitirme vivir esta experiencia por guiarme y protegerme en cada paso dado.

A mis padres, especialmente a mi madre por todo su esfuerzo dedicación y sacrificio, ya que sin ella no hubiera podido realizar este logro.

A mis queridos hermanos, por su apoyo, amor y consejos dados, gracias este logro también es de ustedes.

A mi querida novia, gracias por su amor, su cariño, su apoyo incondicional en los momentos felices y en los difíciles.

A mi compañero de proyecto, por su colaboración, esfuerzo y amistad

A mis amigos, especialmente a los del “garaje”, que fueron un gran apoyo en este logro, gracias por sus enseñanzas y compañía.

Jesús Alberto Laitón Sánchez

DEDICATORIA

A Dios por su misericordia...

A mis Padres y hermana por ser el motor de mi vida...

A mis amigos por robarme tantas sonrisas...

Johan Sebastián Díaz Rodríguez

CONTENIDO

INTRODUCCIÓN	15
1. DESCRIPCIÓN Y FORMULACIÓN DE LOS DISTINTOS MÉTODOS PARA COMPROBAR EL DESEMPEÑO DE LAS CONDICIONES DE FRONTERA	19
1.1 DESCRIPCIÓN DEL DOMINIO COMPUTACIONAL DE REFERENCIA.....	19
1.2 DESCRIPCIÓN Y FORMULACIÓN DE LOS MÉTODOS.....	19
1.2.1 Coeficiente de reflexión.....	20
1.2.1 Medición de la energía	23
2. IMPLEMENTACIÓN DE LAS CONDICIONES DE FRONTERAS SELECCIONADAS EN LA PLATAFORMA DE SIMULACIÓN	26
2.1 LAS ECUACIONES DE MAXWELL EN EL VACÍO	26
2.2 DESCRIPCIÓN DEL MALLADO DE YEE	27
2.3 CONDICIÓN DE FRONTERA ABSORBENTE: MUR.....	29
2.3.1 Formulación abc de mur.....	29
2.4 CAPAS PERFECTAMENTE ACOPLADAS – PML	31
2.4.1 Definición del medio PML.....	31
2.4.2 Propagación de ondas planas en un medio vacío – PML	33
2.4.3 Propagación de ondas planas en un medio PML – PML.....	35
2.4.4 Implementación del método PML.....	38
2.5 LA CONVOLUCIÓN EN EL MÉTODO DE LAS CAPAS PERFECTAMENTE ACOPLADAS (CPML)	43
2.5.1 Formulación CPML.....	44
2.5.2 Ecuaciones en el dominio del tiempo	45
2.5.3 Implementación de las condiciones CPML	46
3. RESULTADOS OBTENIDOS DE LA COMPARACIÓN DE LAS CONDICIONES DE FRONTERAS	50
3.1 MEDIDAS DEL COEFICIENTE DE REFLEXIÓN.....	50

3.1.1 Contexto de medición.....	50
3.1.2 Resultados	51
3.2 MEDICION DE ENERGÍA	62
3.2.1 Formulación sobre la medicion de la energia en el mallado.....	62
3.2.1 Contexto de medición.....	63
3.1.2 Resultados	64
4. CONCLUSIONES.....	66
REFERENCIAS BIBLIOGRÁFICAS	69
BIBLIOGRAFIA	71
ANEXOS	72

LISTA DE FIGURAS

Figura 1 Error en función del tiempo en el tiempo.....	20
Figura 2 Error en el dominio de la frecuencia.....	21
Figura 3 Comparación de la energía en las fronteras CPML Y PML, respecto al tiempo	24
Figura 4 Posicionamiento de los componentes del campo en el mallado de Yee	28
Figura 5 Problema transversal Magnético TMz	33
Figura 6 Transmisión de onda plana entre dos medios PML	36
Figura 7 Distribución de pérdidas en las regiones de la PML	38
Figura 8 Perfil de pérdidas en las capas PML.....	40
Figura 9 Perfil de pérdidas en las capas PML.....	40
Figura 10 Componentes de los campos en el modo de propagación TMz.....	41
Figura 11 Grafica del pulso gaussiano respecto al tiempo.....	52
Figura 12 Representación del espectro de la señal de entrada.	52
Figura 13 Error de las condiciones de frontera respecto al tiempo.	53
Figura 14 Error en dB, de las fronteras absorbentes PML 6 capas, PML 8 capas y PML 10 capas, en función del tiempo.	55
Figura 15 Error en dB, de las fronteras absorbentes PML 6 capas, PML 8 capas y PML 10 capas, en función de la frecuencia.....	56
Figura 16 Error de la condición de frontera CPML, variando sus capas, en función del tiempo.....	57
Figura 17 Error en función de la frecuencia, para PML y CPML de tercer orden y Mur de segundo orden.	59
Figura 18 Error en función del tiempo, de la comparación de Mur, PML 4 capas y CPML 4 capas de 4 orden.....	60
Figura 19 Error en función de la frecuencia comparando Mur, PML Y CPML de 4 capas de orden 4.....	61
Figura 20 Cantidad de energía en el medio computacional, representado en dB para cada frontera	65

LISTA DE TABLAS

Tabla 1 Abreviatura de los parámetros usados en la simulación	50
Tabla 2 Parámetros de simulación utilizados en las pruebas.....	51
Tabla 3 Pruebas realizadas en la plataforma para la medición del error.....	51
Tabla 4 Resultados de la primera prueba	54
Tabla 5 Resultados de la primera prueba	55
Tabla 6 Resultado de la prueba 2, error máximo	56
Tabla 7 Resultado de la prueba 2, error medio.	56
Tabla 8 Resultados de la prueba 3, error máximo.....	58
Tabla 9 Resultados de la prueba 3, error medio.	58
Tabla 10 Resultados de la prueba 4, error máximo para orden 3	60
Tabla 11 Resultados de la prueba 4, error medio para orden 3	60
Tabla 12 Resultados de la prueba 4, error máximo para orden 4	61
Tabla 13 Resultados de la prueba 4, error medio para orden 4	61
Tabla 14 Parámetros de simulación para el análisis de la energía.	64
Tabla 15 Descripción de la pruebas para el análisis de la energía.	64

LISTA DE ANEXOS

ANEXOS A. CODIGOS UTILIZADOS	72
ANEXO A.1 CONDICIÓN DE FRONTERA ABSORBENTE: MUR	72
ANEXO A.2 CONDICIÓN DE FRONTERA ABSORBENTE PML.....	79
ANEXO A.3 CONDICIÓN DE FRONTERA ABSORBENTE CPML	85
ANEXO A.4 PLATAFORMA DE SIMULACION	93
ANEXO A.5 DOMINIO COMPUTACIONAL DE REFERENCIA.....	101

RESUMEN

TÍTULO:

Evaluación del desempeño de las condiciones de frontera absorbentes en la simulación de la propagación de ondas electromagnéticas 2-D realizada con el método FDTD.*

AUTORES:

JESUS ALBERTO LAITON SANCHEZ
JOHAN SEBASTIAN DIAZ RODRIGUEZ**

PALABRAS CLAVE:

CFA (Condiciones de Frontera Absorbentes), PML (Perfectly Matched Layer), FDTD(Finite-Difference Time-Domain), ABC (Absorbing Boundary Conditions), CPML (Convolutional Perfectly Matched Layer)

DESCRIPCIÓN:

Con el desarrollo de este trabajo de investigación se pretende evaluar el desempeño de tres métodos de condición de fronteras absorbentes (CFA) en una plataforma de simulación para la propagación de electromagnéticas en 2-D El objetivo principal es el de iniciar el estudio de los detalles en las formulaciones hechas sobre la simulación FDTD (Finite- Difference Time-Domain), analizando ondas electromagnéticas, Para ello, con esta propuesta de investigación se ha comenzado con la recapitulación de las técnicas de evaluación de desempeño de las condiciones de fronteras, elaboradas en el contexto de las ondas electromagnéticas. A partir de una revisión bibliográfica se seleccionarán tres clases de fronteras absorbentes a ser analizadas. Posteriormente se implementarán cada una de las condiciones de frontera absorbentes seleccionadas en una plataforma de simulación de propagación de ondas electromagnéticas. Para evaluar el desempeño de cada una de las condiciones de frontera absorbentes se usarán medidas existentes en la literatura o alguna otra que pueda ser propuesta y fácilmente implementada. También serán tenidos en cuenta algunos parámetros que alteran su desempeño: ángulo de incidencia, cambios de medio. Finalmente se realizará un estudio comparativo entre cada una de las condiciones de fronteras absorbentes implementadas según los criterios de evaluación establecidos. Este trabajo se pensará como una introducción y motivación al estudio de las condiciones de frontera absorbentes y en general al estudio de los métodos numéricos en la simulación de la propagación de ondas.

* Trabajo de investigación.

** Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones Director: Ernesto Aguilera Bermudez. Doctor en ingeniería electrónica. Co-director: Ivan Javier Sanchez estudiante de maestría en ingeniería electrónica.

ABSTRACT

TITLE:

Evaluation the performance of the absorbing boundary conditions in the simulation of the propagation of electromagnetic waves 2-d performed with the fdtd method.

AUTHORS:

JESUS ALBERTO LAITON SANCHEZ
JOHAN SABASTIAN DIAZ RODRIGUEZ

KEY WORDS:

PML (Perfectly Matched Layer), FDTD (Finite- Difference Time-Domain), ABC (Absorbing Boundary Conditions), CPML (Convolutional Perfectly Matched Layer)

DESCRIPTION:

The research project aims to evaluate the performance of three absorbing boundary conditions in a simulation platform for the electromagnetic propagation waves in 2D. The main objective is to start the study about the details on the formulation made to the FDTD (Finite Difference Time Domain) analyzing electromagnetic waves. According to above, this research has started with the recapitulation of evaluation techniques to the performance of conditions boundaries made in the electromagnetic wave context. First, a bibliographic revision was done in order to select the three types of absorbing boundaries. Then, every selected Absorbing Boundary Conditions will be implemented in a simulation platform of electromagnetic waves. In order to evaluate the performance of every Absorbing Boundary Conditions, existent measures will be used or something that can be easily proposed and implemented. Some parameters that can affect the Absorbing Boundary Conditions will be taking into account too, as the angle of incidence and medium changes. Finally, a comparative study will be done between every implemented Absorbing Boundary Conditions according to the evaluation parameters. This work is seen as an introduction and motivation to the study of the absorbing boundary conditions and in general the study of numerical methods in the simulation of wave propagation.

* Research Work

** Faculty of Physical-Mechanic Engineering. School of Electrical, Electronic and Telecommunications Engineering.
Advisor: Ernesto Aguilera Bermudez. Doctor of Electronic Engineering. Co-advisor: Ivan Sanchez.
Master's degree student in electronic engineering.

INTRODUCCIÓN

Debido al carácter finito de la simulación computacional, es necesario diseñar métodos que permitan resolver problemas que involucren el uso de dominios infinitos, tales como los problemas de radiación de una onda electromagnética desde una antena o la propagación a través de superficies de orden continental de ondas elásticas causadas por las perturbaciones que generan los pequeños movimientos de las placas tectónicas.

MOTIVACIÓN Y JUSTIFICACIÓN DEL TRABAJO

Diferentes formulaciones de condiciones de fronteras absorbentes se han usado para resolver este problema desde la aproximación FDTD. Una referencia importante, es el trabajo de Gerrit Mur¹, documentado en la obra *Absorbing Boundary Conditions For The Finite Difference Approximation Of The Time Domain Electromagnetic- Field Equations* 1981. Es un método potencialmente superior al sugerido por Taflove², y consiste en introducir un medio con pérdidas alrededor del mallado problema, que logren absorber las ondas viajantes, además de disminuir los efectos refractivos que aparecen cuando las ondas no se propagan en una dirección específica, inherente desventaja de una primera aproximación del método

Una de las publicaciones más reconocidas en el campo de las ondas electromagnéticas, es el elaborado por Berenger³. Conocidas como "PML" (Perfectly Matched Layer), la técnica de Berenger se basa en el uso de capas absorbentes especialmente diseñadas para eliminar reflexiones de las ondas

¹ Mur, G. Absorbing Boundary Conditions for the Finite-Difference Approximation of the Time-Domain Electromagnetic-Field Equations. *IEEE Transactions on Electromagnetic Compatibility*, 377 - 382. (1981). P. 377 – 382.

² Taflove, A. Application of the Finite-Difference Time-Domain Method to Sinusoidal Steady-State Electromagnetic-Penetration Problems. *IEEE Transactions on Electromagnetic Compatibility*, 191 - 202. (1980). P. 191 – 202.

³ Berenger, J.-P. A perfectly matched layer for the absorption of electromagnetic waves. *Journal of computational physics*. (1994). P. 185-200.

electromagnéticas. La PML consiste en un medio de espesor finito que rodea el espacio computacional, constituido por parámetros ficticios para crear una condición de emparejamiento de impedancia entre la interfaz vacío-PML y la interfaz PML-PML

Otra propuesta revisada es el método Convolutional PML (CPML)⁴. Se presenta como una variante de las PML, que logra resolver los problemas en la absorción de las ondas evanescentes que presenta el método clásico PML. En consecuencia, se dice, que esta propuesta permite ubicar las fronteras más cerca de las fuentes u objetos dentro del espacio problema. Además, desde la formulación CPML no es necesario dividir los campos para lograr los parámetros constitutivos del medio que permiten evaluar las condiciones de acople de impedancia.

Uno de los principales objetivos es la implementación de los tres métodos escritos anteriormente, mediante el uso de MATLAB, y establecer criterios que permitan medir y comparar cada condición de frontera.

Al final del trabajo de grado, se expondrá en tablas los resultados obtenidos de la plataforma, mediante el análisis de las mismas, se describirá cual método tiene mejor desempeño en cada una de las pruebas realizadas, además se diseñará la plataforma de simulación, en la cual se podrán escoger los métodos a comparar.

PLANTEAMIENTO DEL PROBLEMA DE INVESTIGACIÓN

Es importante formalizar las ventajas o desventajas de un método sobre otro con el fin de facilitar la elección del más indicado en diversos problemas. El objetivo es iniciar en la selección técnicas más intuitivas y posteriormente precisarlas para que puedan ser utilizados en futuros proyectos de modelado computacional. Para ello este trabajo de investigación propone valorar el desempeño de las soluciones reseñadas en el problema de propagación de ondas electromagnéticas,

⁴ J. Alan Roden, S. D. Convolution PML (CPML): An efficient FDTD implementation of the CFS-PML for arbitrary media. Microwave and Optical Technology Letters, 27, (2000). P. 334-339.

estudiando los criterios, procedimientos y cantidades físicas más apropiadas que permitan compararlas sobre una plataforma de simulación 2D.

OBJETIVOS DEL TRABAJO DE INVESTIGACIÓN

El objetivo general del trabajo de investigación consiste en evaluar el desempeño de tres métodos de condición de fronteras absorbentes en una plataforma de simulación para la propagación de ondas electromagnéticas en 2-D, Para ello se proponen los siguientes objetivos específicos:

- Establecer los criterios, procedimientos y las cantidades físicas adecuadas que permitan medir el grado de reflexión en las paredes absorbentes que permitan su revisión.
- Implementar tres condiciones de fronteras absorbentes seleccionadas, sobre una plataforma de simulación de la propagación de ondas electromagnéticas en 2-D
- Comparar las condiciones de fronteras absorbentes implementadas según los criterios, procedimientos y las cantidades físicas establecidas.

ESTRUCTURA DEL DOCUMENTO

Este documento se ha dividido en 5 capítulos, descritos a continuación:

- **Capítulo 1:** En este capítulo se describe el planteamiento del problema y los objetivos.
- **Capítulo 2:** En este capítulo se realiza una revisión bibliográfica sobre los diferentes métodos que permitan medir el grado de reflexión en las paredes absorbentes para el análisis del desempeño de las fronteras seleccionadas.
- **Capítulo 3:** En este capítulo se describen las condiciones de frontera escogidas, con el fundamento teórico y la formulación respectiva, además, se menciona las características más importantes de cada método. Teniendo en cuenta esto, se describe la implementación sobre la plataforma.
- **Capítulo 4:** En este capítulo se muestran los resultados obtenidos en el análisis de las condiciones de frontera, generados en la plataforma de simulación, mostrando en tablas los resultados obtenidos.

- **Capítulo 5:** En este capítulo se describen las conclusiones de acuerdo al análisis de los datos obtenidos en el capítulo 4.
- **Anexos:** En esta sección se pondrá en detalle los códigos utilizados en este proyecto.

1. DESCRIPCIÓN Y FORMULACIÓN DE LOS DISTINTOS MÉTODOS PARA COMPROBAR EL DESEMPEÑO DE LAS CONDICIONES DE FRONTERA

En este capítulo se muestran y describen los distintos métodos y criterios utilizados para realizar el análisis del desempeño de las condiciones de frontera.

1.1 DESCRIPCIÓN DEL DOMINIO COMPUTACIONAL DE REFERENCIA

Para tomar el dominio computacional de referencia, en este caso sin fronteras absorbentes, el mallado tendrá un mayor tamaño respecto a los dominios con condiciones de fronteras absorbentes a analizar, donde las fronteras de este espacio, son fronteras conductoras o fronteras PEC (por sus siglas en inglés), la fuente se ubica en el centro del mallado, y el punto de muestreo será ubicado a la misma distancia de la fuente tanto en el espacio computacional de referencia como con las fronteras absorbentes. El tamaño del mallado en el dominio computacional de referencia se establece con el fin de emular la propagación de una onda sobre un dominio infinito. Además, el mallado debe ser lo suficientemente grande para que las reflexiones desde las fronteras PEC no alcancen el campo muestra.

1.2 DESCRIPCIÓN Y FORMULACIÓN DE LOS MÉTODOS.

Mediante una búsqueda bibliográfica, se seleccionaron algunos trabajos que presentan diferentes métodos para medir el desempeño de las condiciones de frontera, dentro de los más relevantes se encontraron los siguientes:

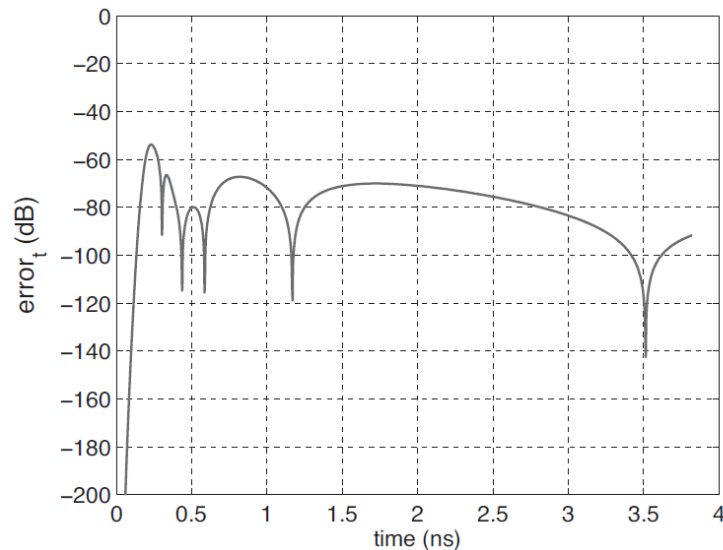
1.2.1 Coeficiente de reflexión. En el libro de Atef⁵, evalúan la PML en dos dimensiones para el caso Tez. El espacio problema tiene dimensiones de 36x36 celdas con tamaño de celda de 1 mm, con 8 celdas para las fronteras PML, con un coeficiente teórico de reflexión de 10^{-8} . La onda de excitación es una onda gaussiana y se ubica en el centro de la rejilla. El punto de muestreo se ubica en el punto (8,8) y la simulación se ejecuta durante 1800 pasos de tiempo, representando el campo magnético muestreado Hz en función del tiempo.

La expresión denotada en la ecuación (1) es el error en función del tiempo:

$$error_t = 20 \log_{10} \left(\frac{|H_z^{pml} - H_z^{ref}|}{\max(|H_z^{ref}|)} \right) \quad (1)$$

Donde H_z^{ref} es el caso referencia tomado de la simulación con fronteras conductoras, donde los resultados obtenidos con esta fórmula se representan en la figura (1)

Figura 1. Error en función del tiempo en el tiempo



Fuente: Atef Z. Elsherbeni, V. D. *The Finite-Difference Time-Domain Method for Electromagnetics with MATLAB Simulations*. SciTech Publishing, Inc. (2009). P. 221

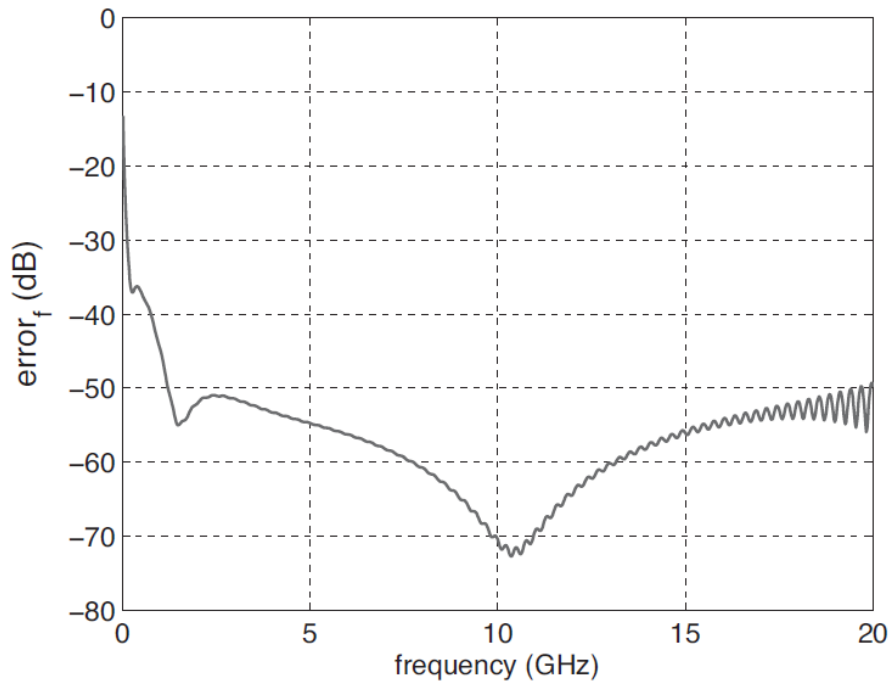
⁵ Atef Z. Elsherbeni, V. D. *The Finite-Difference Time-Domain Method for Electromagnetics with MATLAB Simulations*. SciTech Publishing, Inc. (2009). P.187-229

Como se observa en la figura (1), el pico máximo se encuentra ligeramente por encima de los $-60dB$. Para el error en el dominio de la frecuencia, se obtiene de la diferencia entre las transformadas de Fourier de los campos magnéticos muestreados, a partir de la ecuación (2):

$$error_t = 20 \log_{10} \left(\frac{|F(H_z^{pml}) - F(H_z^{ref})|}{F(H_z^{ref})} \right). \quad (2)$$

Donde $\mathcal{F}()$ denota la transformada de Fourier. Estos resultados son mostrados en la figura (2) donde se obtiene la gráfica en *decibeles* respecto a la frecuencia.

Figura 2. Error en el dominio de la frecuencia



Fuente: Atef Z. Elsherbeni, V. D. *The Finite-Difference Time-Domain Method for Electromagnetics with MATLAB Simulations*. SciTech Publishing, Inc. (2009). P. 221

Otro trabajo analizado es el realizado por Yuan⁶, donde se estudia la propagación de una onda plana 2-D en un medio computacional con frontera PML y ABC de MUR de segundo orden, donde lleva los resultados de Berenger a la aplicación de problemas acústicos.

En trabajo realizado por Yuan, realiza un experimento numérico con las siguientes características: Utilizan un mallado de 129x129 celdas de dominio computacional con fronteras PML de 8 capas, y las condiciones de Mur de segundo orden, para realizar la respectiva comparación, y ubicando el pulso de excitación en el punto (64,64), ejecutando la simulación en 350 pasos de tiempo, y como referencia un dominio computacional de 256x256 con fronteras PML.

$$error(x, y) = 10 \log_{10} \left[\frac{\int_0^T |p_{true}(x, y, t) - p(x, y, t)|^2 dt}{\int_0^T |p_{true}(x, y, t)|^2 dt} \right] \quad (3)$$

Con la función error descrita en la ecuación (3), Yuan obtuvo los siguientes resultados:

- El error de reflexión promedio para la simulación PML es de -80 dB.
- El error para segundo orden de Mur promedio Es -40 dB.

donde sus resultados muestran que la frontera PML, es mejor que la de Mur⁷ por una diferencia de 40 dB,

En el trabajo publicado por Daniel S. katz⁸, reproduce las condiciones de frontera propuestas por Berenger, y además de esto implementa las PML a un caso de 3-

⁶ Xiaojun Yuan, D. B. Formulation and validation of Berenger's PML absorbing boundary for the FDTD simulation of acoustic scattering. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 44. (1997). P. 816 – 822.

⁷ Mur, G. Absorbing Boundary Conditions for the Finite-Difference Approximation of the Time-Domain Electromagnetic-Field Equations. *IEEE Transactions on Electromagnetic Compatibility*, 377 - 382. (1981). P. 377 – 382.

⁸ D.S. Katz, E. T. Validation and extension to three dimensions of the Berenger PML absorbing boundary condition for FD-TD meshes. *IEEE Microwave and Guided Wave Letters*, 4. (1994). P. 268 – 270.

D, donde funcionan tan bien como en 2-D. las características del experimento numérico realizado por Daniel S. Katz, se describen a continuación:

Una cuadrícula cubica FDTD, 100x100x50 celdas del mallado computacional, con el pulso de excitación ubicado en el centro del mallado, donde el error de los campos calculados en el espacio computacional se obtuvo restando el campo en cualquier punto dentro de esta cuadrícula (y en cualquier paso de tiempo) con los campos obtenidos en el espacio computacional de referencia. Donde sus resultados verifican las condiciones PML, y además las lleva a problemas de 3-D. en los resultados obtenidos, por Katz, encuentra que las PML son 40 dB más precisas que las de Mur de segundo orden.

1.2.1 Medición de la energía. Otro método para la evaluación de las condiciones de fronteras, es comparar la energía de cada uno de los medios computacionales con fronteras absorbentes. A continuación, se mostrarán algunos resultados de otros trabajos.

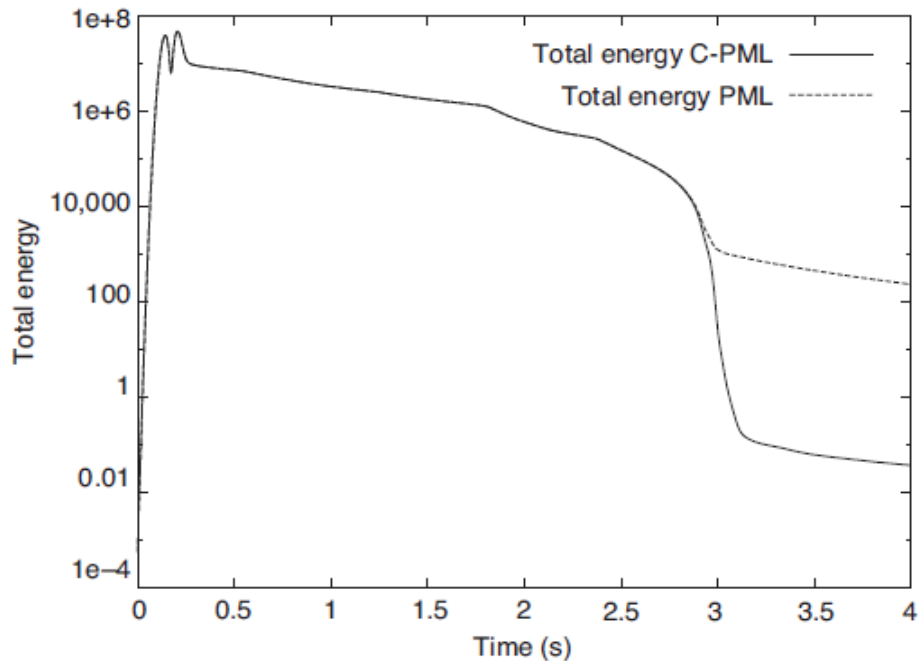
Los resultados obtenidos en el trabajo de Dimitri Komatitsch and Roland Martin⁹, realizan la comparación de la energía total en el caso de la condición de frontera CPML Y PML, para ondas elásticas en un problema de 3 dimensiones, con un dominio computacional de referencia 1000x6400x6400, y un medio con fronteras absorbentes de 101x642x641

A 4s del inicio de la simulación la energía total en todo el dominio computacional con fronteras PML, es de 235,12 J , mientras que en la CPML la energía total es de $3.82 \times 10^{-2} \text{J}$

⁹ Martin, D. K. An unsplit convolutional perfectly matched layer improved. *Geophysics*, 72. (2007). P. 155-167

A continuación, se muestran los resultados obtenidos por este autor, en la comparación de los dos métodos de frontera en la siguiente figura:

Figura 3. Comparación de la energía en las fronteras CPML Y PML, respecto al tiempo



Fuente: Martin, D. K. An unsplit convolutional perfectly matched layer improved. *Geophysics*, 72. (2007). P. 164.

Como se observa en la figura 3, el valor de energía respecto al tiempo de la PML, es mayor que la CPML, para todo los instantes de tiempo de simulación.

En el trabajo de Roger Matsumoto Moreira¹⁰, realizan pruebas de efectividad para condiciones de contorno optimizadas en un esquema de dominio de tiempo en diferencia finita (FDTD) aplicado al modelado de ondas acústicas. En la ecuación

¹⁰ Roger Matsumoto Moreira, R. B. Optimization of absorbing boundary methods for acoustic wave modelling. *Brazilian Congress of Thermal Sciences and Engineering*. (2010). P. 1-10.

(4) y (5) representan una media de la efectividad en la absorción de la energía en un medio computacional.

$$E_{ef} = \sum(U)^2 \quad (4)$$

Donde u representa la amplitud de la onda y w es un factor de atenuación que varía de 1 a 10,

$$U_2 = (1-w)U_1 + wU_1 \quad (5)$$

Uno de los resultados que se obtiene de este trabajo es que para la condición PML se logra un porcentaje de absorción de la energía de 99.56%.

2. IMPLEMENTACIÓN DE LAS CONDICIONES DE FRONTERAS SELECCIONADAS EN LA PLATAFORMA DE SIMULACIÓN

En este capítulo se describirán los métodos seleccionados y se dará una breve descripción de la definición, formulación e implementación de cada condición de frontera.

Al intentar resolver las ecuaciones de Maxwell en el dominio del tiempo mediante el método en diferencias finitas FDTD¹¹, surgían dificultades en el momento de solucionar los muy conocidos problemas de Scattering, ya que estos implican calcular los campos sobre dominios sin límites. Claramente el problema estaba en el hecho de que ninguna computadora podría lograr almacenar la cantidad ilimitada de datos de los campos que se evaluarán en dicho dominio infinito. Este hecho justificaba la creación de un método que permitiera restringir el mallado espacio-tiempo de la formulación FDTD, y además estableciera una condición de frontera alrededor de éste que asegurara la precisión del modelado. Desde entonces a estos métodos se les conoció como condiciones de fronteras absorbentes.

2.1 LAS ECUACIONES DE MAXWELL EN EL VACÍO

Para describir matemáticamente el fenómeno de propagación de ondas electromagnéticas, es necesario introducir las ecuaciones de Maxwell. En esta oportunidad para el caso estudiado se asumirá un medio homogéneo e isotrópico de extensión infinita en 2D para el modo transversal magnético (TM), Además se consideran que las relaciones,

$$\mathbf{D} = \varepsilon\mathbf{E}, \quad \mathbf{B} = \mu\mathbf{H}, \quad \mathbf{J} = \sigma\mathbf{E}, \quad (6),$$

¹¹ Schneider, J. B. *Understanding the Finite-Difference Time-Domain*. (2016).P. 1-403

Donde:

D: vector de densidad de campo eléctrico,

E: vector de campo eléctrico,

J: vector de densidad de corriente,

H: vector de campo magnético,

B: vector de densidad de flujo magnético.

son lineales. Cuando se consideran ondas electromagnéticas en el espacio libre se observa que carece de fuente ($\mathbf{J} = \mathbf{0}$). En estas condiciones, la ecuación de Maxwell se escribe como:

$$\mu_0 \frac{\partial H_x}{\partial t} = \frac{\partial E_z}{\partial x} \quad (7a)$$

$$\mu_0 \frac{\partial H_y}{\partial t} = -\frac{\partial E_z}{\partial y} \quad (7b)$$

$$\varepsilon_0 \frac{\partial E_z}{\partial t} = \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} \quad (7c)$$

2.2 DESCRIPCION DEL MALLADO DE YEE

Para la aproximación de las diferencias finitas de los campos se utiliza el mallado de Yee (ver figura 4), y mediante éste se expresan H y E en el vacío, para el problema de propagación de onda en 2-D en el modo TMz:

$$E_z^{n+1}(i, j) = E_z^n(i, j) + C_{ezhy} * (H_y^{n+\frac{1}{2}}(i, j) - H_y^{n+\frac{1}{2}}(i-1, j)) + C_{ezhx} * (H_x^{n+\frac{1}{2}}(i, j) - H_x^{n+\frac{1}{2}}(i, j-1)) \quad (8a)$$

Donde:

$$C_{ezhy} = \frac{\Delta t}{(\varepsilon)\Delta s} \quad (8a.1)$$

$$C_{ezhx} = \frac{-\Delta t}{(\varepsilon)\Delta s} \quad (8a.2)$$

$$H_x^{n+1/2}(i,j) = H_x^{n-1/2}(i,j) + C_{hx\epsilon z} * (E_z^n(i,j+1) - E_z^n(i,j)) \quad (8b)$$

Donde:

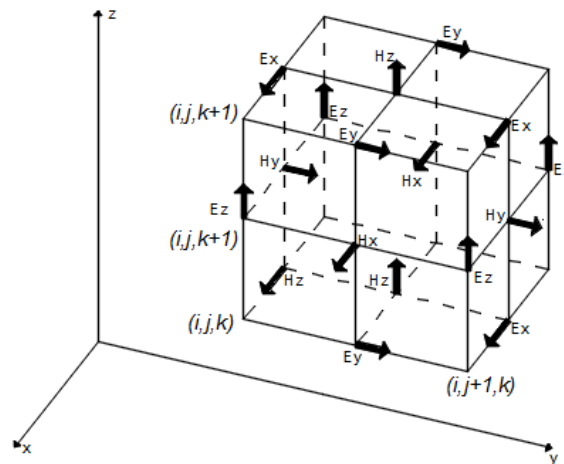
$$C_{hx\epsilon z} = \frac{-2\Delta t}{(2\mu)} \quad (8b.1)$$

$$H_y^{n+1/2}(i,j) = H_y^{n-1/2}(i,j) + C_{hy\epsilon z} * (E_z^n(i+1,j) - E_z^n(i,j)) \quad (8c)$$

Donde:

$$C_{hy\epsilon z} = \frac{\Delta t}{(\mu)\Delta s} \quad (8c.1)$$

Figura 4. Posicionamiento de los componentes del campo en el mallado de Yee



Fuente: Yee, K. Numerical solution of initial boundary value problems involving maxwell's equations in isotropic media. IEEE Transactions on Antennas and Propagation, 14. (1996). P. 303.

La condición de estabilidad para el problema de dos dimensiones descrito en la ecuación (9), es llamado parámetro de Courant:

$$\delta t \leq (C_0 \sqrt{2}) \quad (9)$$

Donde C_0 denota la velocidad de luz en el vacío.

2.3 CONDICIÓN DE FRONTERA ABSORBENTE: MUR

En el trabajo realizado por Mur¹², presenta condiciones de límite altamente absorbentes para las ecuaciones de campo electromagnético, tanto para problemas bidimensionales como para tridimensionales, la principal ventaja de este método, es que se puede aplicar fácilmente a obstáculos conductores, eléctricos o magnéticos, homogéneos o no homogéneos.

2.3.1 Formulación abc de mur. Mur presenta las condiciones de contorno que se describe en detalle por Engquist y Majda¹³ donde un componente espacio-temporal de una onda plana que se desplaza en la dirección decreciente de x , con componentes de velocidad inversa s_x , s_y y s_z , donde $s_x^2 + s_y^2 + s_z^2 = c_0^{-2}$ pueden describirse como:

¹² Mur, G. Absorbing Boundary Conditions for the Finite-Difference Approximation of the Time-Domain Electromagnetic-Field Equations. *IEEE Transactions on Electromagnetic Compatibility*, 377 - 382. (1981). P. 377 - 382.

¹³ Majda, B. E. Absorbing Boundary Conditions for the Numerical Simulation of Waves. *Mathematics of Computation*, 31(139), (1977). P.629-651.

$$W = Re(\psi(t + (c_0^{-2} - s_y^2 - s_z^2)^{\frac{1}{2}}x + s_y y + s_z z) \quad (10)$$

$$\partial_x - c_0^{-1}(1 - (1 - (c_0 s_y)^2 - (c_0 s_z)^2)^{\frac{1}{2}} \partial_t)W|_{x=0} = 0 \quad (11)$$

Con $Re(c_0^{-2} - s_y^2 - s_z^2)^{1/2} \geq 0$. Para esta onda saliente la condición de frontera de primer orden, para valores fijo de S_y y S_z , determinaría un W en la superficie externa, que es consistente con una onda saliente, como no se conoce el ángulo con que incide la onda que se acerca a $x=0$ se realiza la siguiente aproximación de la ecuación (11) y se obtiene la siguiente:

$$\left(1 - (c_0 s_y)^2 - (c_0 s_z)^2\right)^{1/2} = 1 + 0((c_0 s_y)^2 + (c_0 s_z)^2) \quad (12)$$

$$(\partial_x - c_0^{-1} \partial_t)W|_{x=0} = 0 \quad (13)$$

Usando la aproximación de la raíz cuadrada

$$\left(1 - (c_0 s_y)^2 - (c_0 s_z)^2\right)^{\frac{1}{2}} = 1 - \frac{1}{2}((c_0 s_y)^2 - (c_0 s_z)^2) + 0\left(\left((c_0 s_y)^2 - (c_0 s_z)^2\right)^2\right) \quad (14)$$

Obteniendo la segunda aproximación:

$$(c_0^{-2} \partial_{xt}^2 - c_0^{-2} \partial_t^2 + 1/2(\partial_y^2 + \partial_z^2))W|_{x=0} = 0 \quad (15)$$

De las ecuaciones de (13) y (15) se pueden hallar las aproximaciones de W en el límite exterior. Engquist y Majda¹⁴, llegaron a las mismas condiciones de contorno usando un método diferente.

Para problemas de 2 dimensiones se llega a la siguiente aproximación:

$$(\partial_x E_z - c_0^{-1} \partial_t E_z - (c_0 \mu_0 / 2) \partial_y H_x)_{x=0} = 0 \quad (16)$$

La aproximación de diferencias finitas de (16) se derivó mediante el método de diferencias centradas, utilizando la segunda aproximación en 2 dimensiones y descritas de manera que sean fácil describir computacionalmente.

$$E_z^{n+1}(0, j) = E_z^n(1, j) + \frac{c_0 \delta_t - \delta}{c_0 \delta_t + \delta} (E_z^{n+1}(1, j) - E_z^n(0, j)) - \frac{\mu_0 c_0}{2(c_0 \delta_t + \delta)} (H_x^{n+\frac{1}{2}}(0, j + \frac{1}{2}) - H_x^{n+\frac{1}{2}}(0, j - \frac{1}{2})) + H_x^{n+\frac{1}{2}}(1, j + \frac{1}{2}) - H_x^{n+\frac{1}{2}}(1, j - \frac{1}{2}) \quad (17)$$

¹⁴ Majda, B. E. absorbing Boundary Conditions for the Numerical Simulation of Waves. *Mathematics of Computation*, 31(139), (1977). P.629-651

Donde esta aproximación tiene un error de truncamiento de segundo orden en todos los incrementos, pero tiene la ventaja respecto a la de Engquist y Majda¹⁵ la cual puede utilizarse más cerca de los vértices de la malla.

2.4 CAPAS PERFECTAMENTE ACOPLADAS – PML

Una de las publicaciones más reconocidas en el campo de las ondas electromagnéticas, es el elaborado por Berenger¹⁶. Conocidas como “PML”(Perfectly Matched Layer), la técnica de Berenger se basa en el uso de capas absorbentes especialmente diseñadas para eliminar reflexiones de las ondas electromagnéticas. La PML consiste en un medio de espesor finito que rodea el espacio computacional, constituido por parámetros ficticios para crear una condición de adaptación de impedancia entre la interfaz vacío-PML y la interfaz PML-PML.

2.4.1 Definición del medio PML. En esta sección se retomará la definición de región PML propuesta por Berenger, para el problema de propagación de onda en 2D, considerando el modo de propagación transversal magnético (TMz).

Las ecuaciones de rotacional para el *medio acoplado* (Matched Medium) citado por Berenger (Berenger, 1994), se reducen bajo las consideraciones anteriores, al sistema presentado a continuación:

$$\mu_0 \frac{\partial H_x}{\partial t} + \sigma^m H_x = \frac{\partial E_z}{\partial x} \quad (18a)$$

$$\mu_0 \frac{\partial H_y}{\partial t} + \sigma^m H_y = -\frac{\partial E_z}{\partial y} \quad (18b)$$

¹⁵ Majda, B. E. Absorbing Boundary Conditions for the Numerical Simulation of Waves. *Mathematics of Computation*, 31(139), (1977). P.629-651

¹⁶ Berenger, J.-P. A perfectly matched layer for the absorption of electromagnetic waves. *Journal of computational physics*. (1994). P. 185-200

$$\varepsilon_0 \frac{\partial E_z}{\partial t} + \sigma^e E_z = \frac{\partial H_y}{\partial x} - \frac{\partial H_x}{\partial y} \quad (18c)$$

Las ecuaciones (18) se resuelven para un medio artificial que permite la absorción de ondas electromagnéticas, mientras viajan con incidencia normal hacia la interfaz vacío-medio.

Es importante observar que el símbolo σ^m , conductividad magnética, no representa un parámetro físico, sino un artificio que permite construir un medio que puede absorber los campos magnéticos al igual que los campos eléctricos, a partir de una relación simple conocida como condición de adaptación (19). Esta relación trabaja igualando las impedancias de las ondas que se propagan desde el vacío hacia el medio definido en (18).

$$\frac{\sigma^e}{\varepsilon_0} = \frac{\sigma^m}{\mu_0} \quad (19)$$

Donde:

σ^e , es la conductividad eléctrica; σ^m , es la conductividad magnética; ε_0 , es la permitividad eléctrica del vacío y μ_0 , es la permeabilidad magnética del vacío.

Berenger desarrolló un medio más general que el descrito anteriormente, basándose en la descomposición del campo E_z , para el modo TMz, en dos subcomponentes E_{zx} y E_{zy} . A este nuevo medio le llamó *medio PML* y se definió como aquel que, contando con cuatro campos, H_x , H_y , E_{zx} y E_{zy} , se rige a través de las siguientes cuatro ecuaciones:

$$\mu_0 \frac{\partial H_x}{\partial t} + \sigma_{pmy} H_x = - \frac{\partial (E_{zx} + E_{zy})}{\partial y} \quad (20a),$$

$$\mu_0 \frac{\partial H_y}{\partial t} + \sigma_{pmx} H_y = - \frac{\partial (E_{zx} + E_{zy})}{\partial x} \quad (20b),$$

$$\varepsilon_0 \frac{\partial E_{zx}}{\partial t} + \sigma_{pex} E_{zx} = - \frac{\partial H_y}{\partial x} \quad (20c),$$

$$\varepsilon_0 \frac{\partial E_{zy}}{\partial t} + \sigma_{pey} E_{zy} = -\frac{\partial H_x}{\partial y} \quad (20d).$$

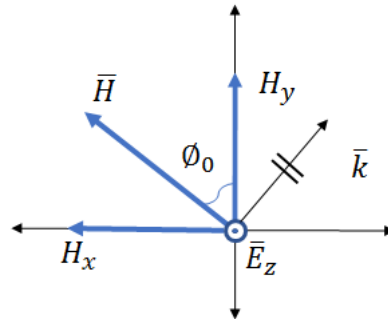
Donde:

$$E_z = E_{zx} + E_{zy} \quad (21).$$

2.4.2 Propagación de ondas planas en un medio vacío – PML. Parte de la estrategia PML busca que una onda que viaja en un medio problema, en nuestro caso particular el vacío, hacia un medio PML, logre propagarse sin reflexión. Una manera de lograrlo es encontrar una relación entre las impedancias de los medios vacío y PML que permita reducir el grado de reflexión.

Partiendo de lo anterior se considera la propagación sobre el medio (20) de una onda electromagnética en el modo ilustrado en la figura 5

Figura 5. Problema transversal Magnético TMz



Fuente: Atef Z. Elsherbeni, V. D. The Finite-Difference Time-Domain Method for Electromagnetics with MATLAB Simulations. SciTech Publishing, Inc. (2009). P.188

Se presenta además las soluciones de onda plana de las ecuaciones en (20):

$$H_x = H_{x0} e^{-jw(t-ax-\beta y)} \quad (22a)$$

$$H_y = H_{y0} e^{-jw(t-ax-\beta y)} \quad (22b)$$

$$\mathbf{E}_{zx} = \mathbf{E}_{zx0} e^{-jw(t-\alpha x-\beta y)} \quad (22c)$$

$$\mathbf{E}_{zy} = \mathbf{E}_{zy0} e^{-jw(t-\alpha x-\beta y)} \quad (22d)$$

Donde w es la frecuencia angular, t es el tiempo y α como β son cantidades complejas relacionadas con el vector de onda \bar{k} . Mediante un tratamiento de las ecuaciones en (20) desde las soluciones presentadas en (22), se logra una relación muy interesante de la impedancia del medio PML con la impedancia del vacío:

$$Z = \sqrt{\frac{u_0}{\epsilon_0}} * \frac{1}{G} \quad (23)$$

Donde:

$$G = \sqrt{w_x \cos^2(\varphi_0) + w_y \sin^2(\varphi_0)} \quad (24a)$$

$$w_x = \frac{1-j\sigma^{ex}/\omega\epsilon_0}{1-j\sigma^{mx}/\omega\mu_0} \quad (24b)$$

$$w_y = \frac{1-j\sigma^{ey}/\omega\epsilon_0}{1-j\sigma^{my}/\omega\mu_0} \quad (24c)$$

Reemplazando la condición de adaptación (19) en las ecuaciones (24b) y (24c), se logra:

$$Z = \sqrt{\frac{u_0}{\epsilon_0}} = Z_0$$

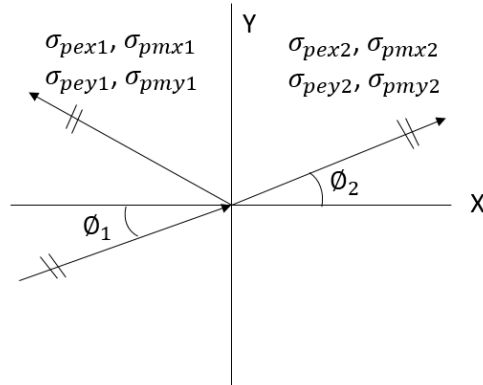
(25)

En (25) se demuestra la importancia de la relación expresada en (19), ya que, a partir de ésta, se logra hacer coincidir la impedancia del medio PML, Z , con la impedancia del vacío, Z_0 . Lo anterior concluye gran parte de los objetivos de la formulación de la propagación de onda en un medio vacío-PML sin reflexión para todas las frecuencias y todos los ángulos de incidencia. Esto último se comprueba desde la ecuación (24a) y la dependencia del parámetro G con la frecuencia ω y el ángulo φ_0 .

2.4.3 Propagación de ondas planas en un medio PML – PML. Al igual que el problema discutido en la sección anterior se buscará lograr la transmisión de una onda desde un medio PML hacia otro medio PML sin reflexión mediante la formulación discutida a continuación.

Una onda polarizada de la forma TM_z , que se propaga con un ángulo de incidencia arbitrario y se desplaza desde una capa PML a otra capa PML, se observa en la figura 6 . Allí particularmente se considera una interfaz normal al eje x .

Figura 6. Transmisión de onda plana entre dos medios PML



Fuente: Atef Z. Elsherbeni, V. D. The Finite-Difference Time-Domain Method for Electromagnetics with MATLAB Simulations. SciTech Publishing, Inc. (2009). P.190

Desde lo anterior, se considera la siguiente expresión del coeficiente de reflexión de una onda con incidencia oblicua entre dos medios con pérdidas.

$$r_p = \frac{Z_2 \cos(\phi_2) - Z_1 \cos(\phi_1)}{Z_2 \cos(\phi_2) + Z_1 \cos(\phi_1)} \quad (26)$$

Aquí Z_1 y Z_2 , representan las impedancias intrínsecas de los medios respectivos. Recordando (21), se logra representar (24) como una ecuación que dependa de la frecuencia y el ángulo de incidencia.

$$r_p = \frac{G_1 \cos(\phi_2) - G_2 \cos(\phi_1)}{G_1 \cos(\phi_2) + G_2 \cos(\phi_1)} \quad (27)$$

Al cumplir la relación (19), conseguimos que $G_1 = G_2 = 1$, sin embargo, el coeficiente de reflexión expresado en (27), sigue dependiendo del ángulo de incidencia de la onda sobre la interfaz. Para hacer más claro el objetivo de

establecer a $r_p=0$, se tiene en cuenta, ahora, la ley de Snell para una interfaz normal al eje x en dos medios con pérdidas.

$$\left(1 - i \frac{\sigma_{y1}}{\varepsilon_0 \omega}\right) \left(\frac{\sin(\theta_1)}{G_1}\right) = \left(1 - i \frac{\sigma_{y2}}{\varepsilon_0 \omega}\right) \left(\frac{\sin(\theta_2)}{G_2}\right) \quad (28)$$

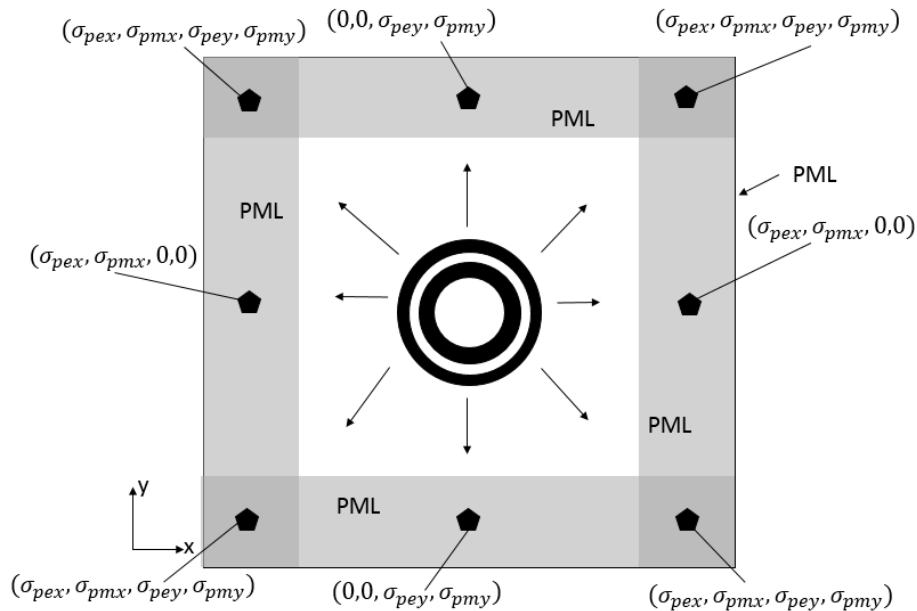
Esta ecuación muestra que cuando dos medios tienen las mismas conductividades $\sigma_{pey1} = \sigma_{pey2}$, $\sigma_{pmy1} = \sigma_{pmy2}$ y además, cuando $\sigma_{pex1}, \sigma_{pmx1}$, $\sigma_{pex2} = \sigma_{pmy2}$ y $\sigma_{pey} = \sigma_{pmy}$ satisfacen la condición (19), conseguimos igualar el ángulo de incidencia de la onda con el ángulo de transmisión y finalmente se logra establecer el valor del coeficiente de reflexión a cero.

Análogamente podemos deducir las condiciones para reducir la reflexión a cero, para el caso donde la onda se propaga hacia una interfaz normal al eje y entre dos medios con pérdidas, partiendo de la ley de Snell.

$$\left(1 - i \frac{\sigma_{x1}}{\varepsilon_0 \omega}\right) \left(\frac{\sin(\theta_1)}{G_1}\right) = \left(1 - i \frac{\sigma_{x2}}{\varepsilon_0 \omega}\right) \left(\frac{\sin(\theta_2)}{G_2}\right) \quad (29)$$

Una situación similar, discutida arriba se observa en (29). Al igualar las componentes tangenciales a la interfaz de las conductividades de los dos medios y al mantener la condición de adaptación, logramos transmitir una onda sin reflexión en la interfaz de dos medios con pérdidas. Este resultado es aplicable al problema discutido en la sección anterior, ya que podemos deducir la propagación de una onda que viaja del vacío a un medio PML, estableciendo las conductividades $\sigma_{pex1} = \sigma_{pmx1} = 0$ y $\sigma_{pey1} = \sigma_{pmy1} = 0$. En la figura 7, se ilustra la disposición de las conductividades y sus valores en las distintas regiones PML.

Figura 7. Distribución de pérdidas en las regiones de la PML



Fuente: Atef Z. Elsherbeni, V. D. The Finite-Difference Time-Domain Method for Electromagnetics with MATLAB Simulations. SciTech Publishing, Inc. (2009). P. 192.

2.4.4 Implementación del método PML. Se comenzará a describir la implementación del método *PML*, discutiendo acerca de la función de pérdidas y posteriormente se mostrarán las ecuaciones en diferencias usadas para calcular los campos.

- **PERFIL DE PÉRDIDAS**

Como se discutió en el apartado anterior la formulación *PML* requiere que se establezca una igualdad en las componentes conductivas tangenciales a la interfaz, además de conseguir cumplir la condición de adaptación, para garantizar la transmisión de la onda sin reflexión. Ya dentro de las regiones *PML* se deben atenuar las ondas mientras estas viajan en el nuevo medio. Para lo anterior se define una función que evalúa la conductividad dentro del medio *PML*, a esta función se le conoce como función de pérdidas o perfil de pérdidas.

Si se considera una interfaz normal al eje x de los medios vacío y *PML*, la función de pérdidas distribuye los valores de la conductividad dentro de la región de acuerdo a las expresiones descritas en las ecuaciones (30) Y (31):

$$\sigma_{ex} = \sigma_{max} \left(\frac{\rho}{\delta} \right)^m \quad (30)$$

$$\sigma_{max} = \frac{-(m+1)\epsilon_0 c_0 \ln(R)}{2\Delta s n_{pml}} \quad (31)$$

Donde:

m: es el orden de las *PML*,

npml: número de celdas *PML*,

R: coeficiente de reflexión teórica,

ρ : es la distancia entre la interfaz y la posición del componente del campo,

δ : espesor de la región *PML*.

La figura 8 y la figura 9, ilustran la manera de implementar el perfil de pérdidas sobre la conductividad eléctrica asumiendo una interfaz normal al eje x.

Figura 8. Perfil de pérdidas en las capas PML.

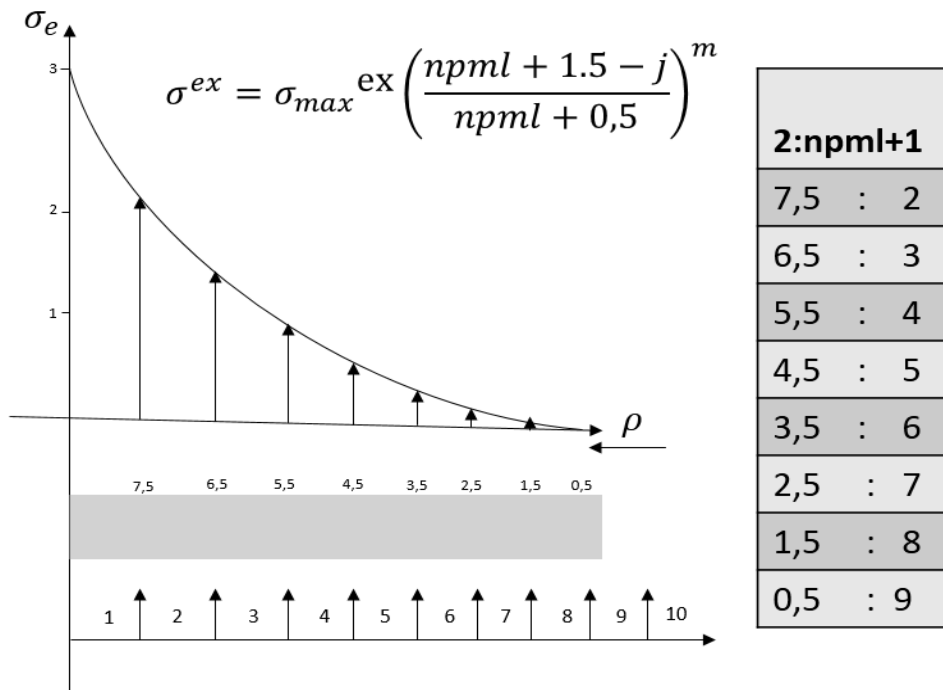
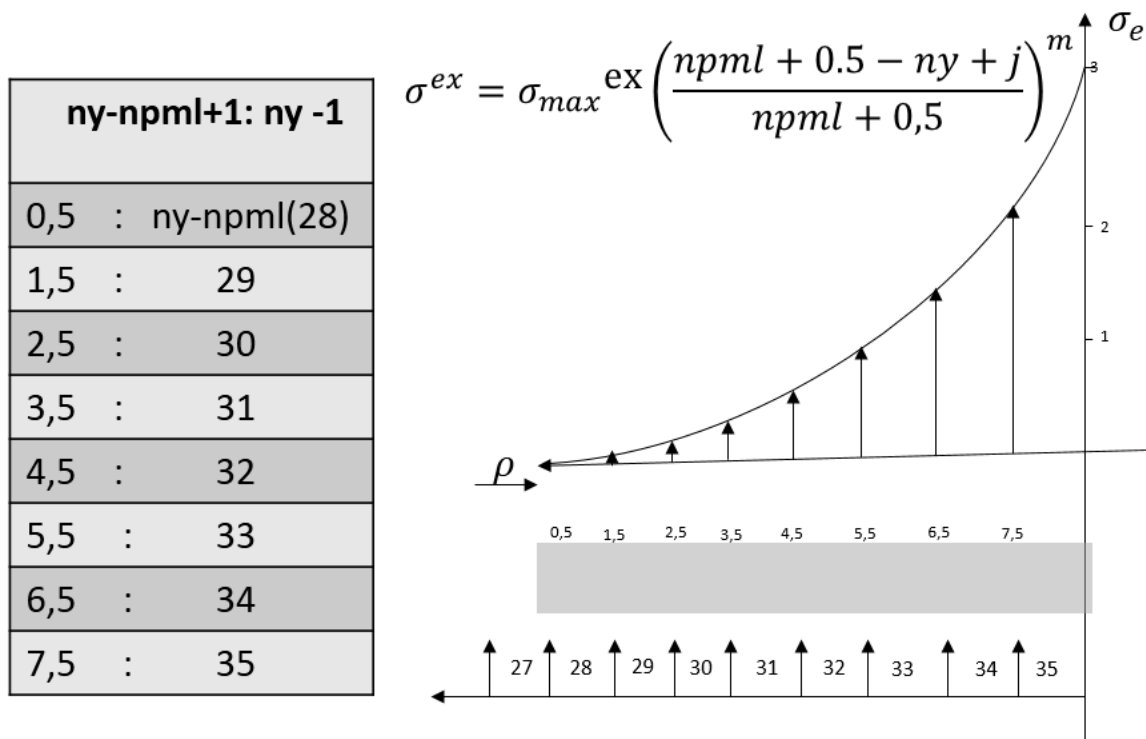


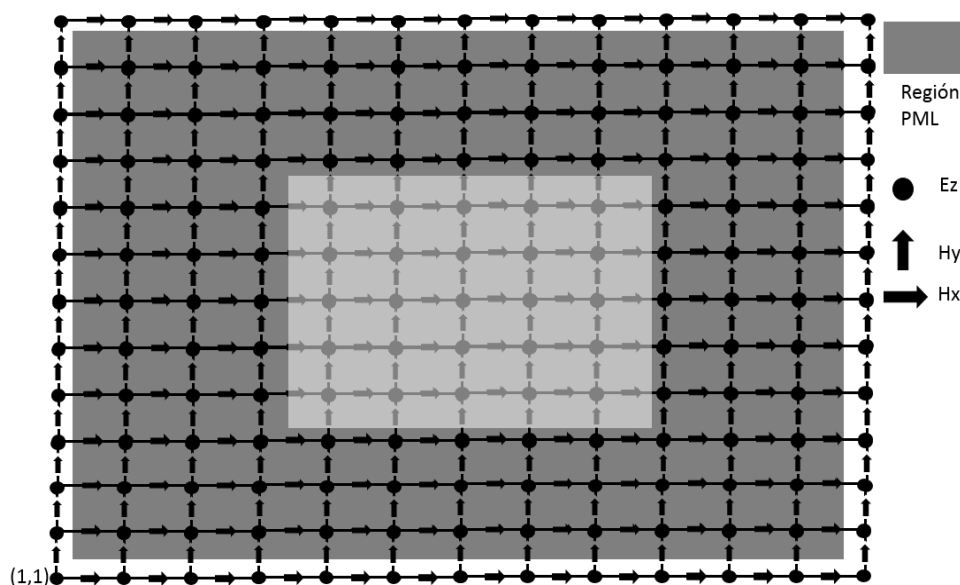
Figura 9. Perfil de pérdidas en las capas PML.



La primera columna de las tablas, representadas en las figuras 8 y 9, corresponden a los valores de ρ , distancia de penetración de la onda desde la interfaz. La segunda, los índices respectivos en las iteraciones computacionales.

En la evaluación del perfil de pérdidas para las conductividades magnéticas, es necesario resaltar que se deben evaluar sobre las posiciones de los campos magnéticos correspondientes.

Figura 10. Componentes de los campos en el modo de propagación TMz.



Fuente: Atef Z. Elsherbeni, V. D. The Finite-Difference Time-Domain Method for Electromagnetics with MATLAB Simulations. SciTech Publishing, Inc. (2009). P. 207

En la figura 10 se muestra el mallado de Yee en 2D¹⁷¹⁸, donde se ilustra el muestreo del campo eléctrico y del campo magnético para el modo de TMz. De esta manera los campos eléctricos son referenciados en los vértices de cada celda del mallado, mientras que los campos magnéticos lo son a mitad de cara.

¹⁷ Yee, K. Numerical solution of initial boundary value problems involving maxwell's equations in isotropic media. *IEEE Transactions on Antennas and Propagation*, 14. (1996). P. 302 – 307.

¹⁸ Schneider, J. B. *Understanding the Finite-Difference Time-Domain*. (2016).P. 1-403

- **Ecuaciones en diferencias finitas**

Las ecuaciones de actualización de los campos en diferencias finitas se pueden obtener desde las ecuaciones de Maxwell, mediante las aproximaciones en diferencias centradas de las derivadas en tiempo y en el espacio. De esta manera se logra las siguientes expresiones:

$$E_z^{n+1}(i,j) = C_{eze} * E_z^n(i,j) + C_{ezhy} * (H_y^{n+\frac{1}{2}}(i,j) - H_y^{n+\frac{1}{2}}(i-1,j)) + C_{ezhx} * (H_x^{n+\frac{1}{2}}(i,j) - H_x^{n+\frac{1}{2}}(i,j-1)) \quad (32a)$$

Donde:

$$C_{eze} = \frac{2\varepsilon - \Delta t \sigma^e}{2\varepsilon + \Delta t \sigma^e} \quad (32a.1)$$

$$C_{ezhy} = \frac{2\Delta t}{(2\varepsilon + \Delta t \sigma^e)\Delta s} \quad (32a.2)$$

$$C_{ezhx} = \frac{-2\Delta t}{(2\varepsilon + \Delta t \sigma^e)\Delta s} \quad (32a.1)$$

$$H_x^{n+1/2}(i,j) = C_{hxx} * H_x^{n-1/2}(i,j) + C_{hxex} * (E_z^n(i,j+1) - E_z^n(i,j)) \quad (30b)$$

Donde:

$$C_{hxx} = \frac{2\mu - \Delta t \sigma^m}{2\mu + \Delta t \sigma^m} \quad (30b.1)$$

$$C_{hxex} = \frac{-2\Delta t}{(2\mu + \Delta t \sigma^m)} \quad (30b.2)$$

$$H_y^{n+1/2}(i,j) = C_{hyh} * H_y^{n-1/2}(i,j) + C_{hyex} * (E_z^n(i+1,j) - E_z^n(i,j)) \quad (30c)$$

Donde:

$$C_{hyh} = \frac{2\mu - \Delta t \sigma^m}{2\mu + \Delta t \sigma^m} \quad (30c.1)$$

$$C_{hyez} = \frac{2\Delta t}{(2\mu + \Delta t \sigma^m)\Delta s} \quad (30c.2)$$

De las ecuaciones (32a), (32b) y (32c) es importante aclarar que, en esta formulación, los campos eléctricos y los campos magnéticos están intercalados en el tiempo.

2.5 LA CONVOLUCIÓN EN EL MÉTODO DE LAS CAPAS PERFECTAMENTE ACOPLADAS (CPML)

Otra propuesta seleccionada es el método *Convolutional PML* (CPML)¹⁹. Se presenta como una variante de las *PML*, que logra resolver los problemas en la absorción de las ondas evanescentes que presenta el método clásico *PML*. En consecuencia, se dice, que esta propuesta permite ubicar las fronteras más cerca de las fuentes u objetos dentro del espacio problema. Además, desde la formulación *CPML* no es necesario dividir los campos para lograr los parámetros constitutivos del medio que permiten evaluar las condiciones de acople de impedancia. Esta última revisión de la *PML* reduce el número de ecuaciones en diferencias que se logran en la formulación *PML*, y utiliza la convolución para resolver los campos en el dominio temporal. Esta descripción se conoce como *Convolutional PML* o *CPML*.

¹⁹ J. Alan Roden, S. D. Convolution PML (CPML): An efficient FDTD implementation of the CFS-PML for arbitrary media. *Microwave and Optical Technology Letters*, 27, (2000). P. 334-339.

2.5.1 Formulación CPML. Con el fin de absorber las ondas evanescentes se ideó una forma estrictamente causal de las *PML* (Mittra, 1996), desplazando los polos dependientes de la frecuencia mediante el uso de las coordenadas *stretched* en las ecuaciones de Maxwell²⁰. Se considera la siguiente ecuación en el dominio de la frecuencia sobre la cual se deducirá la ecuación de actualización del campo eléctrico desde la formulación *CPML*:

$$j\omega\epsilon_0 E_z + \sigma_z^e E_z = \frac{1}{S_{ex}} \frac{\partial H_y}{\partial x} - \frac{1}{S_{ey}} \frac{\partial H_x}{\partial y} \quad (33)$$

Donde

$$S_{ex} = k_{ex} + \frac{\sigma_{ex}}{\alpha_{ex} + j\omega\epsilon_0} \quad (33.1)$$

$$S_{ey} = k_{ey} + \frac{\sigma_{ey}}{\alpha_{ey} + j\omega\epsilon_0} \quad (33.2)$$

σ_z^e : conductividad eléctrica del medio *PML*.

S_{ei} : coordenadas Stretched.

Los términos S_{ei} , son parte de la propuesta que formula la solución al problema de absorción de las ondas evanescentes que presenta el método de Berenger²¹.

Las fronteras *PML*, aseguran un decaimiento exponencial para valores de la conductividad reales y positivos de las ondas que viajan a través de las capas absorbentes, sin embargo, agregan una oscilación a la solución evanescente, sin afectar la tasa de atenuación de ésta. Por otro lado, el método *CPML* incorpora el parámetro α_i desde los términos S_i , para acelerar la absorción de las ondas evanescentes.

Además de lo discutido, el método *CPML* usa capas absorbentes que rodean el espacio problema. Estas capas consisten en medios con pérdidas, conocidos particularmente como medios *PML*. Bajo esta descripción, se deduce sobre este nuevo método la necesidad de discutir el problema descrito en 3.3.2, que se

²⁰ Marshall, U. S. (s.f.). *Numerical Electromagnetics: The FDTD Method*. 2011. P. 1-385.

²¹ Berenger, J.-P. A perfectly matched layer for the absorption of electromagnetic waves. *Journal of computational physics*. (1994). P. 185-200

resuelve considerando una relación lineal entre los parámetros constitutivos. Una nueva condición que introduce los parámetros ki y αi de la expresión más general de las coordenadas *stretched* y *asegura* la propagación sin reflexión en la interfaz entre los medios vacío-PML, se presenta en ecuación (34) (35) y (36):

$$k_{ei} = k_{mi} \quad (34)$$

Y además:

$$\frac{\sigma^e}{\varepsilon_0} = \frac{\sigma^m}{\mu_0} \quad (35)$$

$$\frac{\alpha_e}{\varepsilon_0} = \frac{\alpha_m}{\mu_0} \quad (36)$$

Las ecuaciones (34) (35) y (36) se conocen como la condición de adaptación del método CPML.

2.5.2 Ecuaciones en el dominio del tiempo . La ecuación (33) se presentó en el dominio de la frecuencia. Sin embargo, es necesario conseguir la expresión en el tiempo para obtener las ecuaciones de actualización de los campos. Para lograrlo, se aplica la transformada inversa de Fourier sobre (33), obteniendo:

$$\varepsilon_0 \frac{\partial E_z}{\partial t} + \sigma E_z = \overline{Sx} * \frac{\partial H_y}{\partial x} - \overline{Sy} * \frac{\partial H_x}{\partial y} \quad (37)$$

Donde

\overline{S}_i : es la transformada inversa de Fourier del recíproco del parámetro *stretch*, S_{ei}

$$\begin{aligned} \overline{S}_i(t) &= \frac{\delta(t)}{ki} - u(t) \frac{\sigma i}{\varepsilon_0 k i^2} e^{-\left(\frac{\sigma i}{\varepsilon_0 k i} + \frac{\alpha i}{\varepsilon_0}\right)t} \\ &= \frac{\delta(t)}{ki} + \zeta(t) \end{aligned} \quad (38)$$

De esta manera la ecuación (36), ahora se convierte en:

$$\epsilon_0 \frac{\partial E_z}{\partial t} + \sigma E_z = \frac{1}{k_x} \frac{\partial H_y}{\partial x} - \frac{1}{k_y} \frac{\partial H_x}{\partial y} + \zeta_x(t) * \frac{\partial H_y}{\partial x} - \zeta_y(t) * \frac{\partial H_x}{\partial y} \quad (39)$$

Como se observa en (39), la ecuación en el tiempo involucra la convolución de la transformada inversa del recíproco de los parámetros S_{ei} con las derivadas espaciales de los campos magnéticos. Esta descripción es característica de la formulación *CPML* y difiere por lo tanto de la realizada en la sección 2.1. De manera que las condiciones *CPML* basan su desarrollo en campos de convolución, mientras que las condiciones *PML* lo hacen desdoblado los campos en subcomponentes.

2.5.3 Implementación de las condiciones CPML . Desde la ecuación (39), no es fácil deducir directamente una ecuación en diferencias correspondiente para la simulación FDTD, por esto es necesario antes, resolver las operaciones de convolución presentes. El método de la convolución recursiva permite representar esta operación en una expresión más familiar para la implementación FDTD, partiendo de la respuesta al impulso discreto de los campos $\zeta_i(t)$:

$$\psi_{ez,y} \left[\begin{array}{c} n + \frac{1}{2} \\ i + \frac{1}{2}, j \end{array} \right] = b_y \psi_{ez,y} \left[\begin{array}{c} n - 1/2 \\ i + \frac{1}{2}, j \end{array} \right] + a_y \frac{H_x \left[\begin{array}{c} n + \frac{1}{2} \\ i + \frac{1}{2}, j + \frac{1}{2} \end{array} \right] - H_x \left[\begin{array}{c} n + 1/2 \\ i + \frac{1}{2}, j - 1/2 \end{array} \right]}{\Delta y} \quad (40a)$$

$$\psi_{ez,x} \left[\begin{array}{c} n + \frac{1}{2} \\ i + \frac{1}{2}, j \end{array} \right] = b_y \psi_{ez,x} \left[\begin{array}{c} n - 1/2 \\ i + \frac{1}{2}, j \end{array} \right] + a_y \frac{H_y \left[\begin{array}{c} n + \frac{1}{2} \\ i + \frac{1}{2}, j + \frac{1}{2} \end{array} \right] - H_y \left[\begin{array}{c} n + 1/2 \\ i - \frac{1}{2}, j + 1/2 \end{array} \right]}{\Delta x} \quad (40b)$$

Donde:

$$a_i = \frac{\sigma_i}{\sigma_i k_i + k_i^2 \alpha_i} \left(e^{-\left(\frac{\sigma_i}{k_i} + \alpha_i\right)(\Delta t / \epsilon_0)} - 1 \right) \quad (41)$$

$$b_i = e^{-\left(\frac{\sigma_i}{k_i} + \alpha_i\right)(\Delta t / \epsilon_0)} \quad (42)$$

Los campos expresados en (40), se deben sumar a la ecuación de actualización en el vacío del campo eléctrico.

Las constantes que aparecen en 4.1 y 4.2 dependen los parámetros α_i , σ_i y k_i . Sobre estos parámetros se deben evaluar los perfiles de pérdidas respectivos. Sin embargo sí se establece el valor del parámetro $k_i = 1$ y el valor de α_i se asume despreciable con respecto a los valores de conductividad, se obtiene formas simples de 4.1 y 4.2 que dependen de la conductividad. De esta manera solo bastaría definir una función de pérdidas para los parámetros de conductividad (ver sección 2.1.4). Bajo las consideraciones anteriores se mostrará la manera como se realizó el bucle de iteración principal donde se calculan las ecuaciones de actualización de los campos [anexo].

```
%=====
=====
%=====<Run
time>=====
for n = 1:nsteps
%=====
=====
%=====<Actualizando Campos
Eléctricos>=====
    Pey(x1,y1)= bey(x1,y1).*Pey(x1,y1) + ...
                aey(x1,y1).*(Hx(x1,y1) - Hx(x1,y1-1))/dy;
```

```
Pex(x1,y1)= bex(x1,y1).*Pex(x1,y1) + ...
           ex(x1,y1).*(Hy(x1,y1) - Hy(x1-1,y1))/dx;
```

```
Ez(x1,y1) = Ez(x1,y1)...
           - dt/(eo*dy)*(Hx(x1,y1) - Hx(x1,y1-1))...
           + dt/(eo*dx)*(Hy(x1,y1) - Hy(x1-1,y1))...
           + dt/eo*(Pex(x1,y1) - Pey(x1,y1));
```

```
%=====
=====
```

```
%=====<Insertando
```

```
Fuente>=====
```

```
ma      =((n-to)/tao)*((n-to)/tao);
pulse   = Am*(n-to)*exp(-ma);
Ez(ic,jc) = pulse;
MonitorEzcpml(n) = Ez(ic+8,jc+8);
```

```
%=====
=====
```

```
%=====<Mostrar
```

```
Grafica>=====
```

```
if setting(1) == true
    set(h, 'ZData', Ez)
    drawnow;
end
```

```
%=====
=====
```

```
%=====<Actualizando Campos
```

```
Magnéticos>=====
```

```
Phy(x2,y2)= bhy(x2,y2).*Phy(x2,y2) + ...
           ahy(x2,y2).*(Ez(x2,y2+1) - Ez(x2,y2))/dy;
Phx(x2,y2)= bhx(x2,y2).*Phx(x2,y2) + ...
           ahx(x2,y2).*(Ez(x2+1,y2) - Ez(x2,y2))/dx;
```

$$\begin{aligned}
 H_x(x_2, y_2) &= H_x(x_2, y_2) - dt/(dy \cdot u_0) \cdot (E_z(x_2, y_2 + 1) \dots \\
 &\quad - E_z(x_2, y_2)) - dt/u_0 \cdot P_{hy}(x_2, y_2); \\
 H_y(x_2, y_2) &= H_y(x_2, y_2) + dt/(dx \cdot u_0) \cdot (E_z(x_2 + 1, y_2) \dots \\
 &\quad - E_z(x_2, y_2)) + dt/u_0 \cdot P_{hx}(x_2, y_2);
 \end{aligned}$$

end

3. RESULTADOS OBTENIDOS DE LA COMPARACIÓN DE LAS CONDICIONES DE FRONTERAS

En este capítulo mostraremos todos los resultados obtenidos de la plataforma realizada en Matlab, se analizarán las distintas graficas de error, energía y tiempo de cómputo para cada método y se finalizará mostrando en una tabla donde se expondrán estos resultados eligiendo el mejor método de acuerdo a las condiciones previamente analizadas.

3.1 MEDIDAS DEL COEFICIENTE DE REFLEXIÓN

Para la medida de reflexión en las tres condiciones de frontera elegidas, ABC-MUR, *PML* y *CPML* se asumió el problema de propagación de onda electromagnética en 2D en el modo transversal magnético, TMz.

3.1.1 Contexto de medición. A continuación, se resumirá los parámetros de simulación utilizados para la comparación de los tres métodos mediante el método de reflexión.

Tabla 1. Abreviatura de los parámetros usados en la simulación

PARAMETROS		
FDTD	<i>Tamaño de Celda</i>	ds
	<i>Tamaño de mallado</i>	Tm
	<i>Párametro de Courant</i>	S
	<i>Paso de tiempo</i>	Dt
	<i>Número de iteraciones</i>	Nsteps
ABC	<i>Orden</i>	M
	<i>Número de capas</i>	Npml
Fuente	<i>Tipo de onda</i>	Derivada Normalizada Gaussiana
	<i>Frecuencia máxima</i>	Fmax
	<i>Número de celdas por longitud de onda</i>	Nc
	<i>Desplazamiento Temporal</i>	To
	<i>Ancho de pulso</i>	Tao

Tabla 2. Parámetros de simulación utilizados en las pruebas

PARAMETROS		MUR	PML	CPML	DOMINIO REFERENCIA
FDTD	Tamaño de Celda	0,01	0,01	0,01	0,01
	Tamaño de mallado	36*36	36*36	36*36	600*600
	Parámetro de Courant	0,3536	0,3536	0,3536	0,3536
	Paso de tiempo	2,12E-12	2,12E-12	2,12E-12	2,12E-12
	Número de iteraciones	900	900	900	900
Monitor	Ubicación del monitor	(26,26)	(26,26)	(26,26)	(26,26)
ABC	Orden	2	2	2	NA
	Número de capas	NA	4,6,8,10	4,6,8,10	NA
Fuente	Tipo de onda	Derivada Gaussian a Normalizada	Derivada Gaussian a Normalizada	Derivada Gaussian a Normalizada	Derivada Gaussian Normalizada
	Frecuencia máxima	15[GHz]	15[GHz]	15[GHz]	15[GHz]
	Número de celdas por longitud de onda	20	20	20	20
	Desplazamiento Temporal	70,7107	70,7107	70,7107	70,7107
	Ancho de pulso	15,7135	15,7135	15,7135	15,7135

3.1.2 Resultados. En esta sección se discutirán los resultados obtenidos en la metodología revisada en el capítulo 1, para la medición de la reflexión de las ondas electromagnéticas.

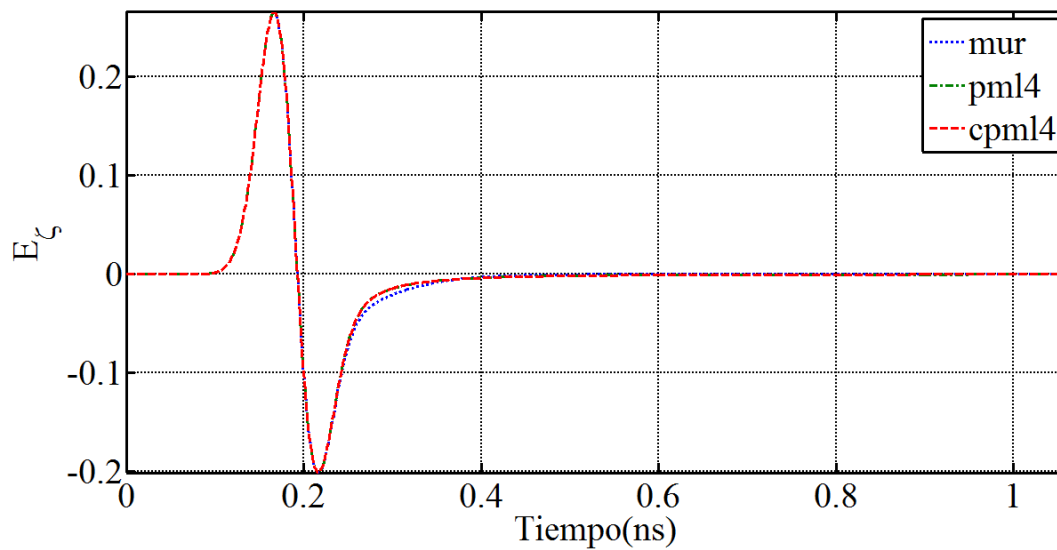
En la siguiente tabla se describen las pruebas realizadas en la medición de la reflexión.

Tabla 3. Pruebas realizadas en la plataforma para la medición del error

PRUEBAS	DESCRIPCIÓN	GRAFICAS MOSTRADAS
PRUEBA 1	Comparación de las tres condiciones de frontera elegidas, MUR, PML, CPML.	Señal del campo muestra, Espectro de la señal del campo muestra, Error medida reflexión- Tiempo, Error medida reflexión- Frecuencia.
PRUEBA 2	Comparación PML con 6, 8, 10 capas absorbentes	Error medida reflexión- Tiempo, Error medida reflexión- Frecuencia.
PRUEBA 3	Comparación CPML con 6, 8, 10 capas absorbentes	Error medida reflexión- Tiempo, Error medida reflexión- Frecuencia.
PRUEBA 4	Comparación CPML y PML capas absorbentes, orden 3 y 4	Error medida reflexión- Tiempo, Error medida reflexión- Frecuencia.

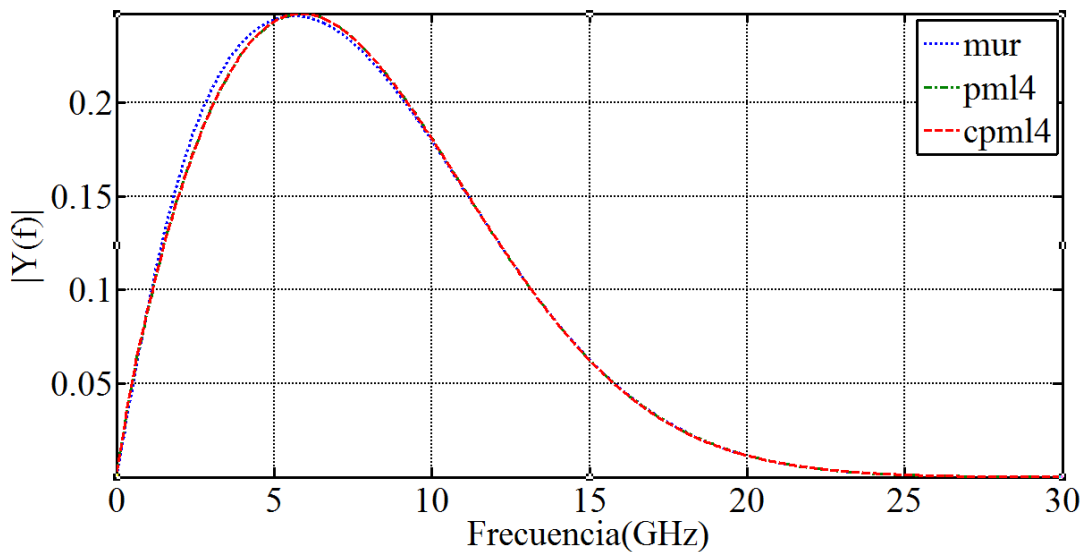
Resultados: Prueba 1

Figura 11. Grafica del pulso gaussiano respecto al tiempo.



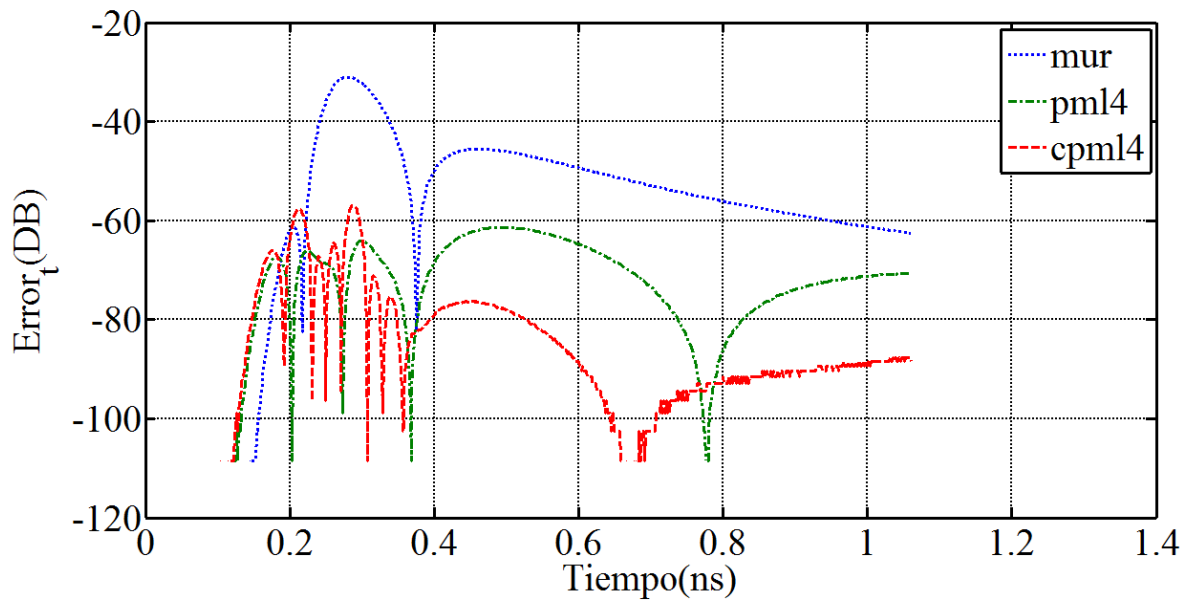
En la figura 11, se muestra el pulso simulado en cada medio computacional con fronteras absorbentes.

Figura 12. Representación del espectro de la señal de entrada.



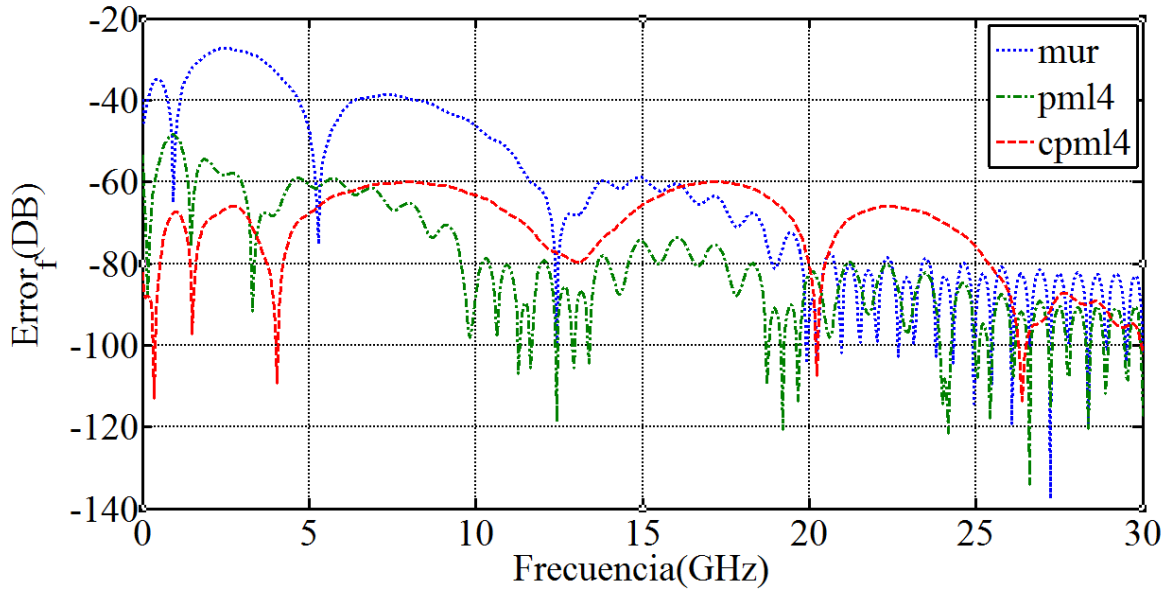
En la figura 12, se representa el espectro de la señal de entrada, donde indica la frecuencia de corte.

Figura 13. Error de las condiciones de frontera respecto al tiempo.



En la figura 13 **Error! No se encuentra el origen de la referencia.**, se representa la comparación de las condiciones de frontera absorbentes, donde se analiza la condición de Mur de orden 2, las condiciones PML de 4 capas y la condición CPML de 4 capas, respecto al tiempo.

Figura 14. Error en dB, de las fronteras absorbentes, en función de la frecuencia.



En la figura 14 , se obtiene la comparación de las condiciones de frontera PML y CPML de 4 capas y Mur de segundo orden en función de la frecuencia.

Tabla 4. Resultados de la primera prueba

PRUEBA 1		
Frontera	Max(Error-Tiempo) [dB]	Max(Error-Frecuencia) [dB]
MUR	-31,007	-27,228
PML	-61,383	-48,489
CPML	-56,916	-59,973

Para representar cuantitativamente los resultados obtenidos en las figuras 13 y 14, se calcula el valor medio del error de cada señal medida desde el monitor en cada método CFA (condición de frontera absorbente) con respecto a la señal medida desde el monitor del dominio referencia a través de la siguiente expresión:

$$\overline{error_t} = \frac{1}{nsteps} \sum \frac{|monitorcfa-monitorref|}{\max|monitorref|} \quad (43)$$

$$\overline{error_f} = \frac{1}{nsteps} \sum \frac{|F(monitorcfa)-F(monitorref)|}{\max(F(monitorref))} \quad (44)$$

Donde,

$monitor_{cfa}$, es la señal medida desde el monitor de cada CFA; $monitor_{ref}$ es la medida desde el monitor en el dominio de referencia y $F(x)$, es la transformada de Fourier de la señal x .

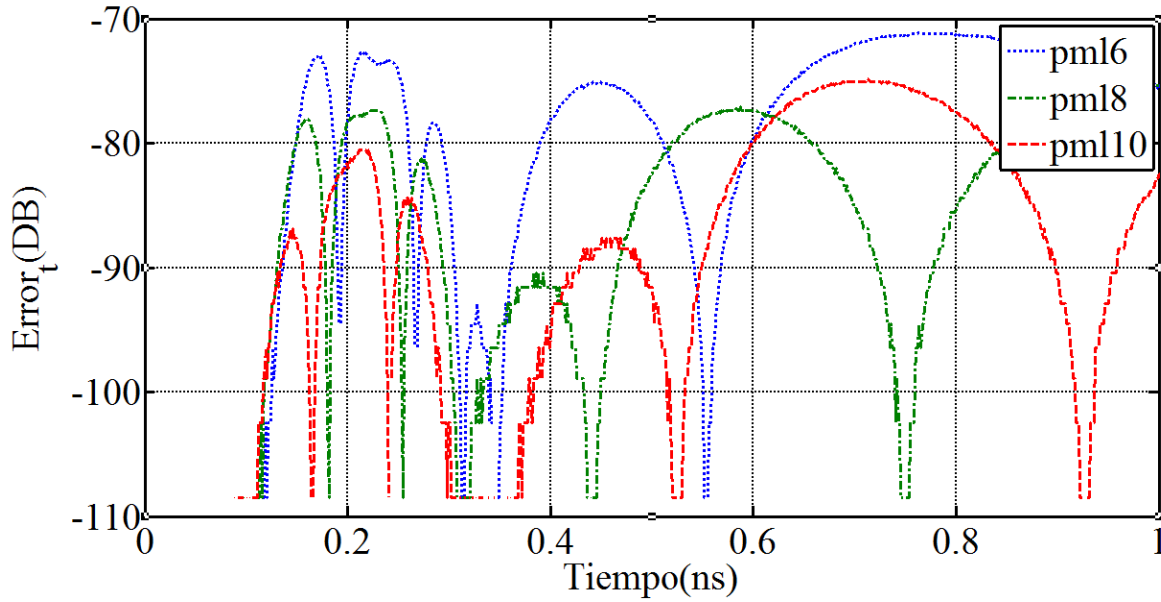
En la tabla 4, se resumen los resultados obtenidos en el cálculo del error de acuerdo a la ecuación (43) y (44):

Tabla 5. Resultados de la primera prueba

PRUEBA 1		
Frontera	Error Medio - Tiempo[dB]	Error Medio - Frecuencia[dB]
MUR	-111,592	-95,283
PML	-161,728	-152,322
CPML	-179,558	-151,021

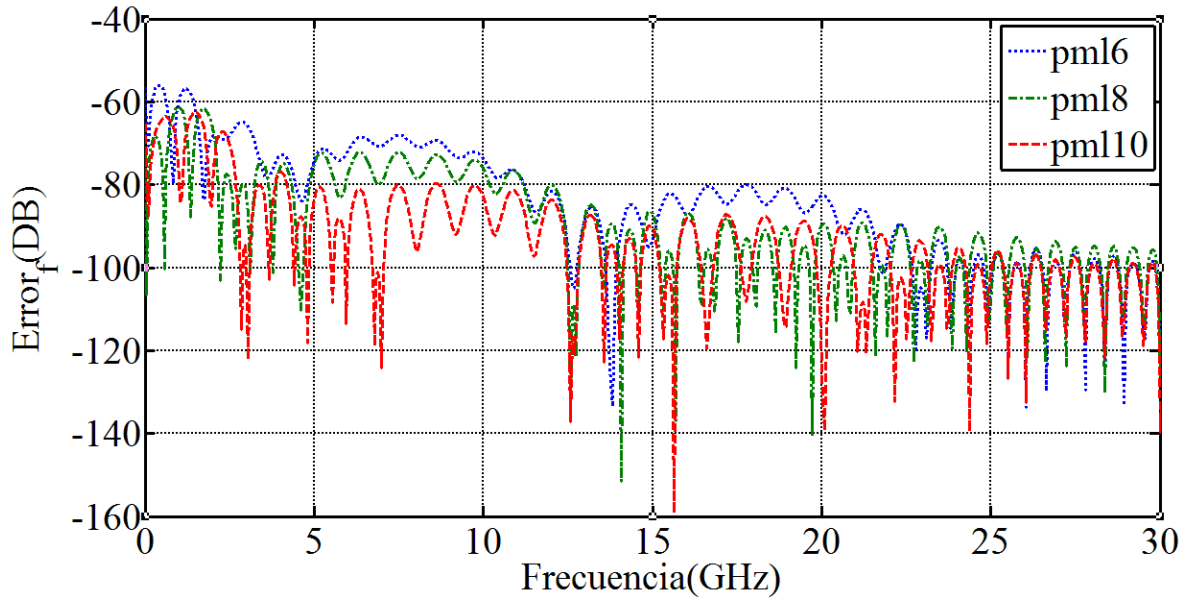
Resultados: Prueba 2

Figura 15. Error en dB, de las fronteras absorbentes PML 6 capas, PML 8 capas y PML 10 capas, en función del tiempo.



En la figura 15, se muestran los resultados obtenidos en la plataforma de la condición de frontera PML, variando el número de capas, 6, 8 y 10 respectivamente

Figura 16. Error en dB, de las fronteras absorbentes PML 6 capas, PML 8 capas y PML 10 capas, en función de la frecuencia.



En análisis en frecuencia de la comparación de PML, con 6, 8 Y 10 capas se representa en la tabla 6.

Tabla 6. Resultado de la prueba 2, error máximo

PRUEBA 2		
Frontera	<i>Max(Error-Tiempo[dB])</i>	<i>Max(Error-Frecuencia[dB])</i>
PML6	-71,085	-55,998
PML8	-75,024	-61,175
PML10	-74,841	-62,436

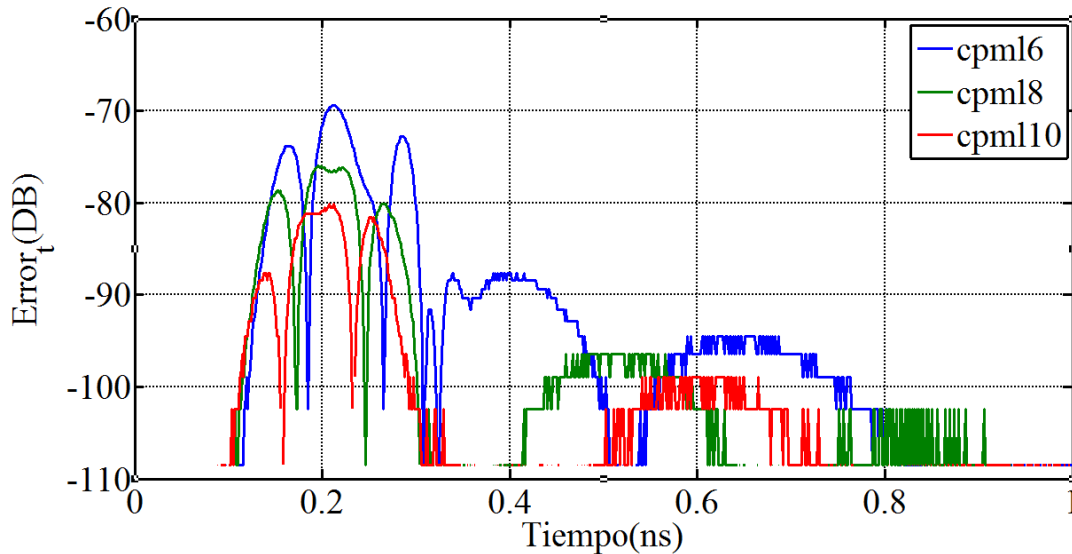
Tabla 7. Resultado de la prueba 2, error medio.

PRUEBA 2		
Frontera	<i>Error Medio - Tiempo[dB]</i>	<i>Error Medio - Frecuencia[dB]</i>
PML6	-177,300	-167,250
PML8	-189,897	-179,810
PML10	-194,024	-186,807

En este segundo ejercicio se evalúa el desempeño de las PML, al aumentar el número de capas. De las tablas 6 y 7 se puede ver que al pasar de cuatro a ocho capas el pico máximo de error en el tiempo disminuye -49 a casi -76 dB, demostrando una mejora. Sin embargo, al establecer el número de éstas en doce, el pico máximo se establece en -17 dB.

Resultados: Prueba 3

Figura 17. Error de la condición de frontera CPML, variando sus capas, en función del tiempo



En la figura 17 se observa el análisis en tiempo de la condición de frontera CPML, esta vez variando el número de capas, en este caso, 6 8 y 10 respectivamente.

Figura 18. Error en función de la frecuencia, comparación de CPML con 6 capas, CPML con 8 capas y CPML con 10 capas.

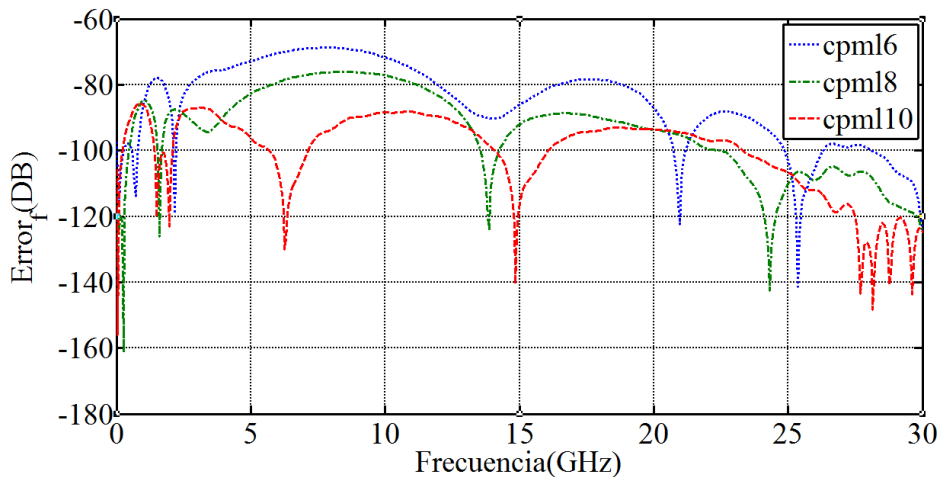


Tabla 8. Resultados de la prueba 3, error máximo.

PRUEBA 3

Frontera	<i>Max(Error-Tiempo[dB])</i>	<i>Max(Error-Frecuencia[dB])</i>
CPML6	-69,381	-68,699
CPML8	-76,001	-76,035
CPML10	-80,166	-85,683

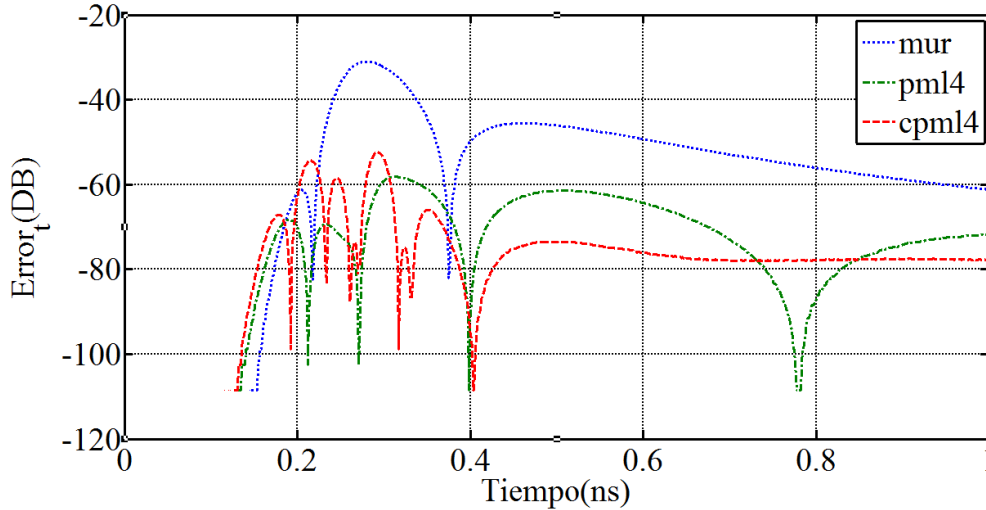
Tabla 9. Resultados de la prueba 3, error medio.

PRUEBA 3		
Frontera	<i>Error Medio - Tiempo[dB]</i>	<i>Error Medio - Frecuencia[dB]</i>
CML6	-204,661	-175,508
CPML8	-217,497	-193,393
CPML10	-227,174	-214,660

Desde los resultados de la prueba 3, se consiguió disminuir en casi 192 *dB* el error cuadrático medio aumentando el número de capas de cuatro a ocho, confirmando una mejora en el desempeño de las CPML como la observada en la prueba 2 con las PML.

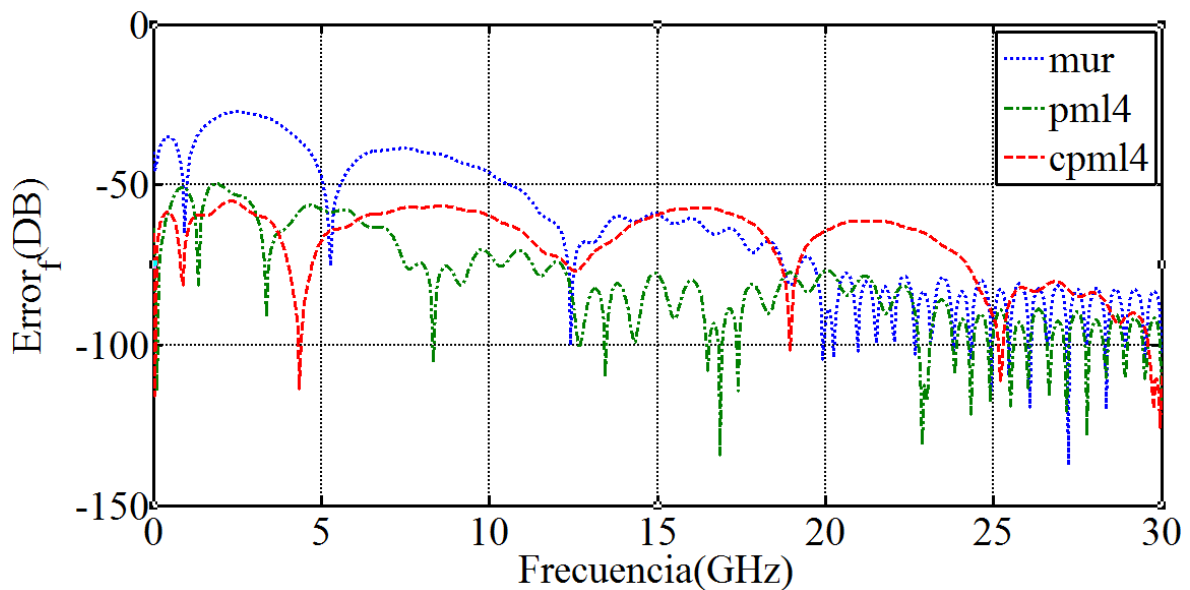
Resultados: Prueba 4

Figura 19. Error en función del tiempo, para Mur de segundo orden , PML 4 capas y CPML 4 capas.



En la figura 19, se obtiene la gráfica de error en función del tiempo, para las condiciones de frontera, pero esta vez cambiando el orden de 2 a un orden 3 , de las condiciones de frontera CPML y PML.

Figura 20. Error en función de la frecuencia, para PML y CPML de tercer orden y Mur de segundo orden.



En la figura 20 se muestra los resultado obtenidos de la comparación de las condiciones de frontera de Mur segundo orden y las condiciones CPML Y PML de tercer orden

Tabla 10. Resultados de la prueba 4, error máximo para orden 3

PRUEBA 4 (ORDEN 3)		
Frontera	<i>Max(Error-Tiempo[dB])</i>	<i>Max(Error-Frecuencia[dB])</i>
PML4	-58,148	-49,692
CPML4	-52,356	-55,058

Tabla 11. Resultados de la prueba 4, error medio para orden 3

PRUEBA 4 (ORDEN 3)		
Frontera	<i>Error Medio - Tiempo[dB]</i>	<i>Error Medio - Frecuencia[dB]</i>
PML4	-159,917	-149,910
CPML4	-167,538	-141,843

Figura 21. Error en función del tiempo, de la comparación de Mur, PML 4 capas y CPML 4 capas de 4 orden

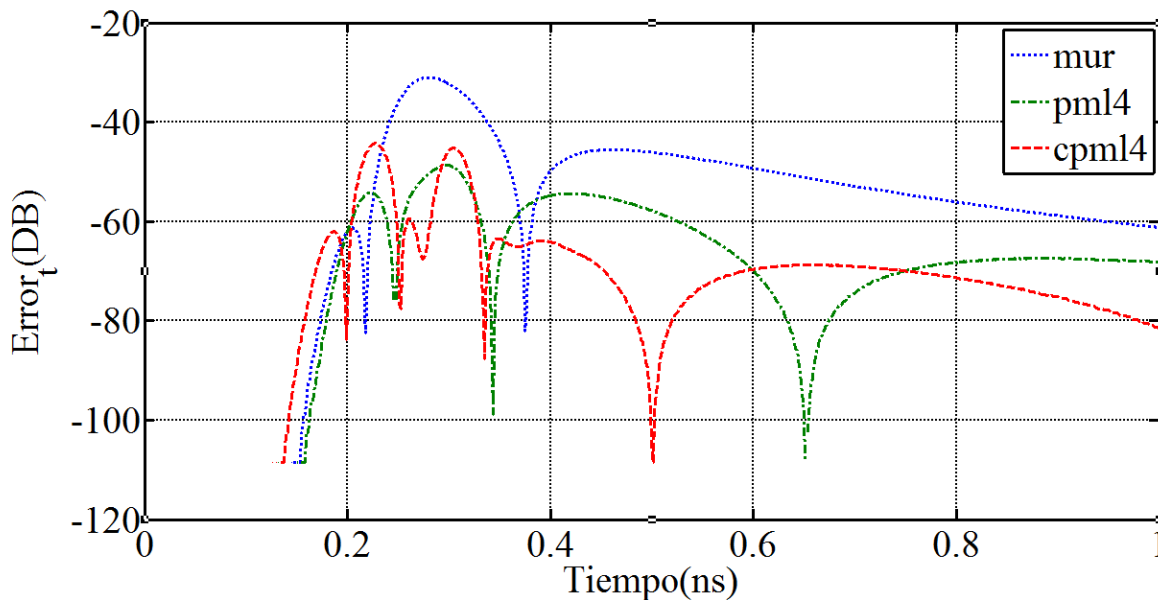
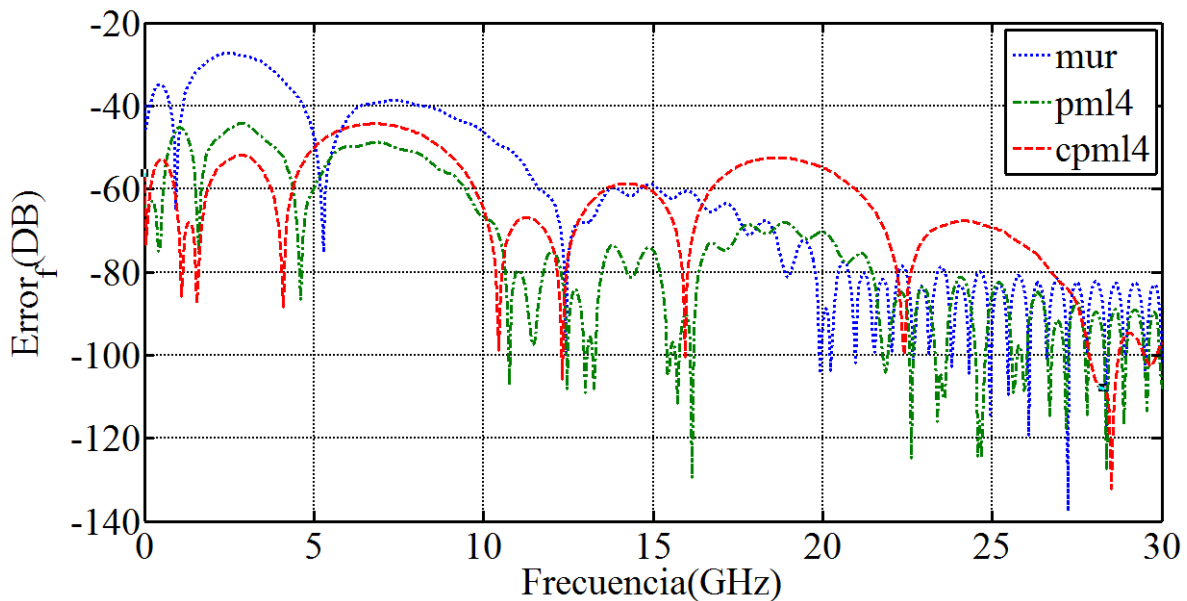


Figura 22. Error en función de la frecuencia comparando Mur, PML Y CPML de 4 capas de orden 4



En la figura 22, se obtiene el resultado de la comparación de Mur, CPML 4 CAPAS Y PML 4 CAPAS de orden 4

Tabla 12. Resultados de la prueba 4, error máximo para orden 4

PRUEBA 4 (ORDEN 4)		
Frontera	<i>Max(Error-Tiempo[dB])</i>	<i>Max(Error-Frecuencia[dB])</i>
PML4	-48,686	-44,235
CPML4	-44,308	-44,289

Tabla 13. Resultados de la prueba 4, error medio para orden 4

PRUEBA 4 (ORDEN 4)		
Frontera	<i>Error Medio - Tiempo[dB]</i>	<i>Error Medio - Frecuencia[dB]</i>
PML4	-145,143	-130,854
CPML4	-149,910	-125,003

En los resultados obtenidos en la prueba 1, la medida de error medio de las fronteras CPML difieren en 13,56 dB en el tiempo y de 21,71 dB en frecuencia con respecto a las medidas realizadas sobre las PML. Aquí se puede verificar que las CPML implementadas con ocho capas, logran una diferencia de 2 dB y

1,74dB con las *PML*, también de ocho capas, en las medidas del error en el tiempo y la frecuencia respectivamente.

3.2 MEDICION DE ENERGÍA

Para la medición de la energía, se analiza la condición *PML* y *CPML*, comparándolas con las fronteras PEC

3.2.1 Formulación sobre la medición de la energía en el mallado. A continuación, se pondrá en detalle sobre el método utilizado para la medición de la energía de las distintas condiciones de frontera, utilizando la formulación realizada por Hayt en su libro de teoría electromagnética²²:

Para encontrar el flujo de potencia de la onda electromagnética, se utilizará el teorema de Pointing, partiendo de las ecuaciones rotacionales de maxwell tenemos la siguiente ecuación:

$$\rho_{\mu} = \frac{\epsilon_0}{2} |E|^2 + \frac{\mu_0}{2} |H|^2 \quad (45)$$

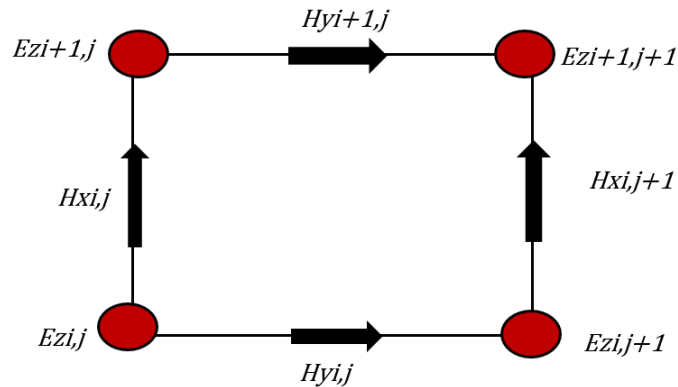
Donde $H = H_x + H_y$, y $E = E_x$, reescribiendo la ecuación:

$$\rho_{\mu} = \frac{\epsilon_0}{2} E_z^2 + \frac{\mu_0}{2} (H_x^2 + H_y^2) \quad (46)$$

Para hallar la energía total, inicialmente se hallará la energía de cada celda del mallado computacional figura 5,

²² William H. Hayt, J. J. *Engineering Electromagnetics*. The McGraw-Hill Companies. 2001. P. 1-567

Figura 23. Representación de los campos Hx, Hy y Ez en una celda de Yee.



Fuente: Atef Z. Elsherbeni, V. D. *The Finite-Difference Time-Domain Method for Electromagnetics with MATLAB Simulations*. SciTech Publishing, Inc. (2009). P. 207

Como se observa en la figura 23, en cada celda se representan los campos H_x , H_y y E_z y sobre esta se aplica la ecuación (46) para hallar la energía, este proceso se aplica a todas las celdas del dominio computacional, y obtener así obtener la energía total del dominio.

Utilizando una escala logarítmica para la representación de la energía dada por la siguiente expresión

$$E = 10 \log_{10} \left(\frac{\epsilon_0}{2} E_z^2 + \frac{\mu_0}{2} (H_x^2 + H_y^2) \right) \quad (47)$$

Donde la ecuación (45), representa la energía total en el espacio problema en decibeles.

3.2.1 Contexto de medición. A continuación, se resumirá los parámetros de simulación utilizados para la comparación de las condiciones CFA, PML; CPML y PEC mediante el método de medida de la energía.

Tabla 14 Parámetros de simulación para el análisis de la energía.

PARAMETROS		PML	CPML	PEC
FDTD	Tamaño de Celda	0,01	0,01	0,01
	Tamaño de mallado	20*20	20*20	20*20
	Parámetro de Courant	0,6364	0,6364	0,6364
	Paso de tiempo	2,12E-12	2,12E-12	2,12E-12
	Número de iteraciones	1e4	1e4	1e4
ABC	Orden	2	2	NA
	Número de capas	4	4	NA
Fuente	Tipo de onda	Derivada Gaussiana Normalizada	Derivada Gaussiana Normalizada	Derivada Gaussiana Normalizada
	Frecuencia máxima	15[GHz]	15[GHz]	15[GHz]
	Número de celdas por longitud de onda	20	20	20
	Desplazamiento Temporal	70,7107	70,7107	70,7107
	Ancho de pulso	15,7135	15,7135	15,7135

3.1.2 Resultados. En esta sección se discutirán los resultados obtenidos en la metodología revisada en el capítulo 1, para la medición de la cantidad de energía remanente en el mallado 2D.

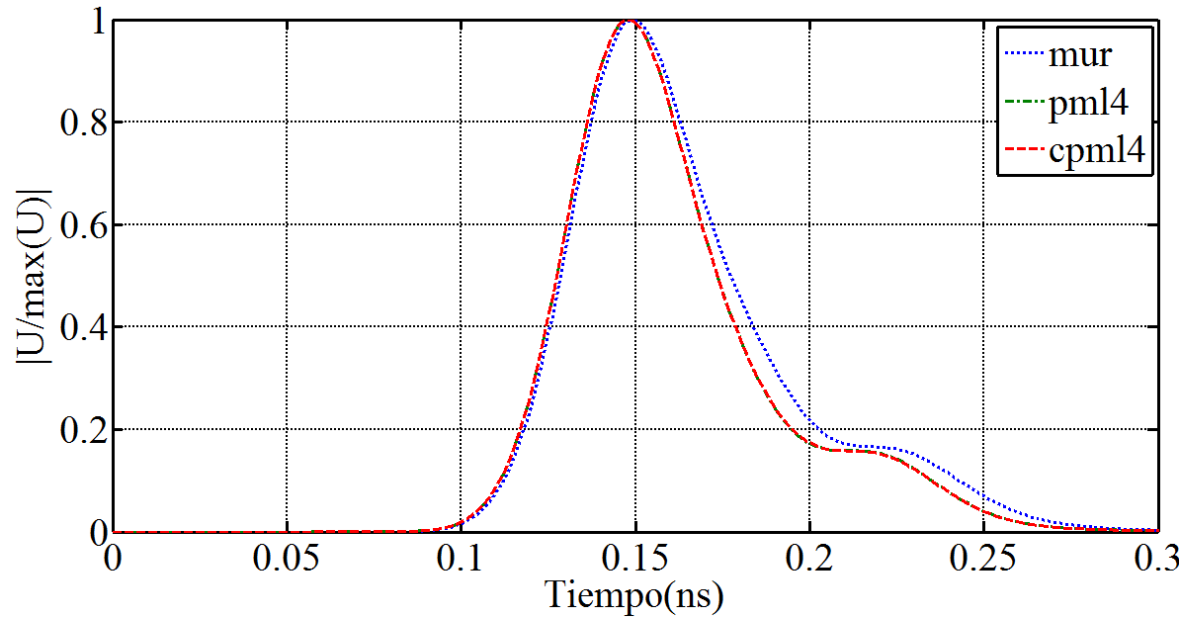
En la siguiente tabla se describen las pruebas realizadas en la medición de la energía.

Tabla 15. Descripción de la pruebas para el análisis de la energía.

PRUEBAS	DESCRIPCIÓN	GRAFICAS MOSTRADAS
PRUEBA 1	Comparación MUR, PML, CPML y Fronteras PEC (paredes conductoras)	Cantidad de Energía en el mallado

PRUEBA 1

Figura 24. Cantidad de energía en el medio computacional, representado en dB para cada frontera



Como se puede observar en esta figura, la energía presente en el mallado computacional con condiciones de frontera *PML CPML Y Mur* es absorbida por las mismas.

4. CONCLUSIONES

En la tabla 4 y Y la tabla 5, se puede observar la concurrencia de la señal fuente y su espectro con las medidas de los monitores en las condiciones *PML* y *CPML*. De los valores de error medio resumidos en la Tabla 4. Resultados de la primera prueba, se demuestra un mejor desempeño en la implementación *CPML*, al acercarse al comportamiento ideal de una onda propagándose sobre un medio infinito para el intervalo de simulación de 0-1[ns] , logrando el valor de $-179,558$ *dB* en la media, mejorando en $17,83$ *dB* y $67,97$ *dB* sobre las condiciones *PML* y *MUR* respectivamente. Por otro lado en esta implementación *PML* de cuatro capas se logró un pico máximo de $-61,383$ *dB* garantizando un desempeño para todo el tiempo de simulación de menos de -60 *dB*, mejorando sobre el criterio de pico máximo del error en el tiempo descrito en el apartado 2.2.1.

Desde los resultados de la prueba 2, de la medida del coeficiente de reflexión, la implementación de las *PML* de seis, ocho y diez capas superan por más de 40 *dB*, el desempeño de la condición de *MUR*, cumpliendo con un buen comportamiento de acuerdo al criterio mencionado en la sección 2.2.1. Además, al aumentar el número de capas de la región *PML*, el error con respecto a la medida sobre el dominio computacional de referencia disminuye tanto en tiempo como en frecuencia según lo muestra los valores consignados en la tabla (prueba 2 error medio).

Al relacionar las tablas (prueba 2 y prueba 3) y compararlas, se observa que a medida que el número de capas aumenta, para la implementación *PML* y *CPML*, ambas tienen un mejor desempeño, pero la condición de frontera *CPML* supera la condición *PML*, en cada capa. Para la capa 6 hay 27.331 *dB* de diferencia, 27.6 *dB* para 8 capas y 33.15 *dB* para 10 capas.

Sobre la prueba 4 se tomaron las condiciones de PML y CPML cada una con cuatro capas y se varió el orden del polinomio de la función de pérdidas de 3 a 4 orden. Se demuestra de acuerdo a la prueba 4, se demuestra que al aumentar el orden del polinomio aumenta el error para la señal en el tiempo en $14,77dB$ en el caso PML y de $17,62dB$ para el caso CPML, mientras que el error de la señal en la frecuencia aumenta en $19,056 dB$ y $16,84 dB$ para los casos PML y CPML respectivamente.

Además de las ya mencionadas pruebas sobre la medida del coeficiente de reflexión mediante el uso de un dominio computacional de referencia, se realizaron pruebas sobre la medida de la cantidad de energía en todo el dominio problema. Desde allí se calculó la tasa de absorción de la energía de cada condición de frontera MUR, PML y CPML en un intervalo de $0.15 - 0.25$ [ns]. A partir de esto se dice que las PML y CPML tienen una tasa de absorción de $95,64\%$, mientras que la condición de MUR presenta una razón de absorción de $93,04\%$. Es importante aclarar que estos valores se deducen desde los resultados mostrados en capítulo 1. Sobre este intervalo de tiempo logramos obtener resultados de la tasa de absorción que difiere en $3,92\%$ de los citados en la sección 2.2.2 sobre la implementación PML.

al realizar una prueba del tiempo de computo *sobre* el bucle principal del programa, *run time* en un mallado de 200×200 celdas y 20 capas PML, se deduce que las condiciones PML y CPML son 2 veces más costosas que las condiciones Mur.

TRABAJOS FUTUROS

Se sugiere para trabajos futuros, discutir acerca del problema de absorción de onda evanescente de las fronteras PML, y como la formulación CPML mejora el desempeño sobre las PML en este tipo de problemas. Usando la plataforma suministrada y la implementación del método CPML encontrar una relación de los parámetros α_i y ki , que se mencionan en la sección 3.4.1, en el problema de absorción de onda evanescente.

Por otro lado, se invita a estudiar los efectos en el desempeño de las condiciones MUR, PML y CPML en la absorción de ondas planas con incidencia oblicua.

Además, se propone ampliar el número de pruebas sobre las metodologías sugeridas en este libro usando la plataforma 2D suministrada.

REFERENCIAS BIBLIOGRÁFICAS

Atef Z. Elsherbeni, V. D. *The Finite-Difference Time-Domain Method for Electromagnetics with MATLAB Simulations*. SciTech Publishing, Inc. (2009). P.187-229

Berenger, J.-P. A perfectly matched layer for the absorption of electromagnetic waves. *Journal of computational physics*, (1994). P. 185-200.

D.S. Katz, E. T. Validation and extension to three dimensions of the Berenger PML absorbing boundary condition for FD-TD meshes. *IEEE Microwave and Guided Wave Letters*, 4. (1994). P. 268 - 270.

J. Alan Roden, S. D. Convolution PML (CPML): An efficient FDTD implementation of the CFS-PML for arbitrary media. *Microwave and Optical Technology Letters*, 27, (2000).P. 334-339.

Majda, B. E. bsorbing Boundary Conditions for the Numerical Simulation of Waves. *Mathematics of Computation*, 31(139), (1977). P.629-651.

Marshall, U. S. (s.f.). Numerical Electromagnetics: The FDTD Method.(2011). P. 1-385

Martin, D. K. An unsplit convolutional perfectly matched layer improved. *Geophysics*, 72. (2007). P. 155-167.

Mitra, M. K. Frequency dependence of the constitutive parameters of causal perfectly. *IEEE Microwave and Guided Wave Letters*,, 6(12),. (1996). P. 447-449

Mur, G. Absorbing Boundary Conditions for the Finite-Difference Approximation of the Time-Domain Electromagnetic-Field Equations. *IEEE Transactions on Electromagnetic Compatibility*,. (1981).P. 377 - 382.

Roger Matsumoto Moreira, R. B. Optimization of absorbing boundary methods for acoustic wave modelling. *Brazilian Congress of Thermal Sciences and Engineering* . (2010). P. 1-10.

Schneider, J. B. Understanding the Finite-Difference Time-Domain. (2016).P. 1-403

Taflove, A. Application of the Finite-Difference Time-Domain Method to Sinusoidal Steady-State Electromagnetic-Penetration Problems. *IEEE Transactions on Electromagnetic Compatibility*, 191 - 202. (1980). P. 191 – 202

William H. Hayt, J. J. *Engineering Electromagnetics*. The McGraw-Hill Companies. (2001). P. 1-567

Xiaojuen Yuan, D. B. Formulation and validation of Berenger's PML absorbing boundary for the FDTD simulation of acoustic scattering. *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, 44. (1997). P. 816 - 822

Yee, K. Numerical solution of initial boundary value problems involving maxwell's equations in isotropic media. *IEEE Transactions on Antennas and Propagation*, 14. (1996). P. 302 – 307

BIBLIOGRAFIA

Atef Z. Elsherbeni, V. D. The Finite-Difference Time-Domain Method for Electromagnetics with MATLAB Simulations. SciTech Publishing, Inc. (2009).

Berenger, J.-P.. A perfectly matched layer for the absorption of electromagnetic waves. Journal of computational physics, 185-200. (1994).

J. Alan Roden, S. D.. Convolution PML (CPML): An efficient FDTD implementation of the CFS-PML for arbitrary media. Microwave and Optical Technology Letters, 27, 334-339. (2000).

Mur, G. Absorbing Boundary Conditions for the Finite-Difference Approximation of the Time-Domain Electromagnetic-Field Equations. IEEE Transactions on Electromagnetic Compatibility, 377 - 382. (1981). P. 377 - 382.

Xiaojuen Yuan, D. B. Formulation and validation of Berenger's PML absorbing boundary for the FDTD simulation of acoustic scattering. IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control, 44. (1997).

Yee, K. Numerical solution of initial boundary value problems involving maxwell's equations in isotropic media. IEEE Transactions on Antennas and Propagation, 14. (1996). P. 302 - 307.

ANEXOS

ANEXOS A. CODIGOS UTILIZADOS

En esta sección de mostraran los códigos utilizados para la realización de este proyecto,

ANEXO A.1 CONDICIÓN DE FRONTERA ABSORBENTE: MUR

```
%Bucaramanga Nov. 25 de 2016
%Jesus y Sebastian
%E3T, UIS
% 2-D TMz
% ABC 2° order
clear all; clc ; close all;
disp('run mur...')
setting = [0 1]; %[[Motrar Pelicula,Generar ficheros}
%=====
=====
%=====<Parametros
FDTD>=====
nx = 32;
ny = 32;
nsteps = 1e4;
eo = 8.854e-12;
uo = 4*pi*1e-7;
n0 = sqrt(uo/eo);
ic = (nx)/2;
jc = (ny)/2;
dx = .001;
dy = .001;
```

```

ds = dx;
co = 1/sqrt(eo*uo);
S = 0.5/sqrt(2);
dt = S*ds/co;
...xend = nx + 1;
...yend = ny + 1;
x1 = 2:nx-1;
y1 = 2:ny-1;
x2 = 1:nx-1;
y2 = 1:ny-1;
%=====
=====
%=====<Definiendo Constantes
ABC>=====
%definicion constantes abc 2° order
cab0=-1/(1/S+2+S);
cab1=1/S-2+S;
cab2=2*(S-1/S);
cab3=4*(1/S+S);
%=====
=====
%=====<Iniciando
Campos>=====
%**arreglos ABC 2-D Ez
ezoldleft11=zeros(3,nx);
ezoldleft22=zeros(3,nx);
ezoldright11=zeros(3,nx);
ezoldright22=zeros(3,nx);
ezolddown11=zeros(ny,3);
ezolddown22=zeros(ny,3);
ezoldup11=zeros(ny,3);

```

```

ezoldup22=zeros(ny,3);

Ez=zeros(nx,ny);
Hx=zeros(nx,ny);
Hy=zeros(nx,ny);
%definicion de las constantes de actualizacion

c1=dt/(uo*dx);
c2=dt/(eo*dx);
%=====<...Iniciando
Matrices...>=====
%=====<.....de
Medida.....>=====
if setting(2)== true
    Energiamur = zeros(1,nsteps);
    MonitorEzmur = zeros(nsteps,1);
    EEZ = zeros(nx,ny,nsteps);
    HHX = zeros(nx,ny,nsteps);
    HHY = zeros(nx,ny,nsteps);
end
%=====
=====
%=====<Parametros de la
Fuente>=====
nc = 20;
dsmax = ds;
tao = ((nc*dsmax)/(2*co))/dt;
to = 4.5*tao;
fmax = co/(nc*dsmax);
Am = -(sqrt(2*exp(1))/tao);

```

```

%=====
=====
%===== <Parametros de la
Grafica>=====
if setting(1) == true
    afFigurePosition = [500 100 300.0 300.0];
    h=surf(Ez);
    shading interp;
    colormap(jet(1000))
    lighting gouraud
    view(90,90)
    caxis ([0 0.5])
    axis ([0 nx 0 ny -1 1])
    xlabel('N_{x}','FontSize',14)
    ylabel('N_{y}','FontSize',14)
    zlabel('E_{\zeta}','FontSize',14)
    title({'\fontsize{14}Propagation Mode TM_{z}';...
        '\fontsize{14}(MUR case)'})
    set(gcf,'Position',afFigurePosition);
    axis off
    grid on
    hold on
end
%=====
=====
%===== <Run
Time>=====
for n=1:nsteps
%=====
=====

```

```

%=====<Actualizando Campos
Magneticos>=====
for i=2:nx-1
    for j=1:ny-1
        Hx(i,j)=Hx(i,j)-c1*(Ez(i,j+1)-Ez(i,j));
    end
end

for i=1:nx-1
    for j=2:ny-1
        Hy(i,j)=Hy(i,j)-c1*(Ez(i,j)-Ez(i+1,j));
    end
end

%=====
=====
%=====<Actualizando Campos
Electricos>=====
for i=2:nx-1
    for j=2:ny-1
        Ez(i,j)=Ez(i,j)+c2*(Hy(i,j)-Hy(i-1,j)-Hx(i,j)+Hx(i,j-1));
    end
end

%=====
=====
%=====<Insertando
Fuente>=====
ma    =((n-to)/tao)*((n-to)/tao);
pulse = Am*(n-to)*exp(-ma);
Ez(ic,jc) = pulse;
MonitorEzmur(n) = Ez(ic+8,jc+8);

```

```

%=====
=====
%===== <Mostrar
Grafica>=====
    if setting(1) == true
        set(h, 'ZData', Ez)
        drawnow;
    end
%=====
=====
%===== <ABC para
Ez>=====
    for j=2:ny-1

Ez(1,j)=cab0*(cab1*(Ez(3,j)+ezoldleft22(1,j))+cab2*(ezoldleft11(1,j)+ezoldleft11
(3,j)-Ez(2,j)-ezoldleft22(2,j))-cab3*ezoldleft11(2,j))-ezoldleft22(3,j);
        Ez(nx,j)=cab0*(cab1*(Ez(nx-
2,j)+ezoldright22(1,j))+cab2*(ezoldright11(1,j)+ezoldright11(3,j)-Ez(nx-1,j)-
ezoldright22(2,j))-cab3*ezoldright11(2,j))-ezoldright22(3,j);
    end

    for i=2:nx-1

Ez(i,1)=cab0*(cab1*(Ez(i,3)+ezolddown22(i,1))+cab2*(ezolddown11(i,1)+ezoldd
own11(i,3)-Ez(i,2)-ezolddown22(i,2))-cab3*ezolddown11(i,2))-ezolddown22(i,3);
        Ez(i,ny)=cab0*(cab1*(Ez(i,ny-
2)+ezoldup22(i,1))+cab2*(ezoldup11(i,1)+ezoldup11(i,3)-Ez(i,ny-1)-
ezoldup22(i,2))-cab3*ezoldup11(i,2))-ezoldup22(i,3);
    end

    for m=1:3

```

```

for j=2:ny-1
    ezoldleft22(m,j)=ezoldleft11(m,j);
    ezoldleft11(m,j)=Ez(m,j);

    ezoldright22(m,j)=ezoldright11(m,j);
    ezoldright11(m,j)=Ez(nx-m+1,j);
end
end

for m=1:3
    for i=2:nx-1
        ezolddown22(i,m)=ezolddown11(i,m);
        ezolddown11(i,m)=Ez(i,m);

        ezoldup22(i,m)=ezoldup11(i,m);
        ezoldup11(i,m)=Ez(i,ny-m+1);
    end
end

%=====
=====
%===== <Almacenando
Datos>=====

if setting(2)== true
    EEZ(:, :, n) = Ez;
    HHX(:, :, n) = Hx;
    HHY(:, :, n) = Hy;
end
end
if setting(2)==true
    for n=1:nsteps
        Energiamur(n) = sum(sum((eo/2)*(EEZ(:, :, n).*EEZ(:, :, n))+...

```

```

        (uo/2)*(HHX(:,:,n).*HHX(:,:,n) + HHY(:,:,n).*HHY(:,:,n)));
    end
    Energiamur = (Energiamur/max(Energiamur));
    fid1 = fopen('MonitorEzmur.txt','w');
    fid2 = fopen('Energiamur.txt','w');
    for i=1:nsteps
        fprintf(fid1,'%f\t\n',MonitorEzmur(i));
        fprintf(fid2,'%f\t\n',Energiamur(i));
    end
    fclose(fid1);
    fclose(fid2);
end
disp('successful')

```

ANEXO A.2 CONDICIÓN DE FRONTERA ABSORBENTE PML

%Modo TMZ PML-Berenger

```
clear all; clc ; close all;
```

```
disp('run pml...')
```

```
setting = [0 1]; % {[Mostrar Pelicula,Generar ficheros]
```

```
%=====
```

```
=====
```

```
%=====<Parametros
```

```
FDTD>=====
```

```
nx = 36;
```

```
ny = 36;
```

```
nsteps = 1e4;
```

```
eo = 8.854e-12;
```

```
uo = 4*pi*1e-7;
```

```
n0 = sqrt(uo/eo);
```

```

ic = (nx)/2;
jc = (ny)/2;
dx = .001;
dy = .001;
ds = dx;
co = 1/sqrt(eo*uo);
S = 0.5/sqrt(2);
dt = S*ds/co;
x1 = 2:nx-1;
xe = 4:nx-4;
y1 = 2:ny-1;
ye = 4:ny-4;
x2 = 1:nx-1;
y2 = 1:ny-1;
%=====
=====
%=====<Paramatros
CPML>=====
m = 2;
npml = 4;
%R = 1e-8;
%sigma_max = -(m+1)*eo*co*log(R)/(2*ds*npml);
sigma_factor = 1.5;
sigma_max = sigma_factor*(m+1)/(150*pi*ds);
sigma1_max = (uo/eo)*sigma_max;
%=====
=====
%=====<Definiendo perfil de
perdidas>=====
sigma_m = zeros(nx,ny);
for j=1:npml

```

```

    sigma_m(j,:) = sigma1_max*((npml+1.0-j)/(npml+0.5))^m;
end
for j=ny-npml:ny
    sigma_m(j,:) = sigma1_max*((npml+1.0-ny+j)/(npml+0.5))^m;
end
sigma_e = zeros(nx,ny);
for j=1:npml+1
    sigma_e(:,j)= sigma_max*((npml+1.5-j)/(npml+0.5))^m;
end
for j=ny-npml:ny
    sigma_e(:,j)= sigma_max*((j-ny+npml+0.5)/(npml+0.5))^m;
end
%=====
%=====
%=====<Definiendo Coeficientes de los
Campos>=====
chyh = (2*uo-dt*sigma_m)./(2*uo+dt*sigma_m);
chyez = (2*dt)./((2*uo+dt*sigma_m)*ds);
chxh = chyh';
chxez = -chyez';

cezye = (2*eo-dt*sigma_e)./(2*eo+dt*sigma_e);
cezyhx = -(2*dt)./((2*eo+dt*sigma_e)*ds);
cezxe = cezye';
cezxhy = -cezyhx';
%=====
%=====
%=====<Iniciando
Campos>=====
Ez = zeros(ny,nx);
Ezx = zeros(ny,nx);

```

```

Ezy = zeros(ny,nx);
Hx = zeros(ny,nx);
Hy = zeros(ny,nx);
%=====
=====
%===== <Parametros de la
Fuente>=====
nc = 20;
dsmax = ds;
tao = ((nc*dsmax)/(2*co))/dt;
to = 4.5*tao;
fmax = co/(nc*dsmax);
Am = -(sqrt(2*exp(1))/tao);
%===== <...Iniciando
Matrices...>=====
%===== <.....de
Medida.....>=====
if setting(2)== true
    Energiapml = zeros(1,nsteps);
    MonitorEzpml = zeros(nsteps,1);
    EEZ = zeros(nx,ny,nsteps);
    HHX = zeros(nx,ny,nsteps);
    HHY = zeros(nx,ny,nsteps);
end
%=====
=====
%===== <Parametros de la
Grafica>=====
if setting(1) == true
    afFigurePosition = [500 100 300.0 300.0];
    h=surf(Ez);

```

```

shading interp;
colormap(jet(1000))
lighting gouraud
view(90,90)
caxis ([0 0.5])
axis ([0 nx 0 ny -1 1])
xlabel('N_{x}','FontSize',14)
ylabel('N_{y}','FontSize',14)
zlabel('E_{\zeta}','FontSize',14)
title({'\fontsize{14}Propagation Mode TM_{z}';...
      '\fontsize{14}(PML case)'})
set(gcf,'Position',afFigurePosition);
axis off
grid on
hold on
end
window = 20;
%=====
=====
%=====<Run
Time>=====
for n = 1:nsteps
%=====
=====
%=====<Actualizando Campos
Electricos>=====
Ezx(x1,y1) = cezxe(x1,y1).*Ezx(x1,y1) + ...
            cezxhy(x1,y1).*(Hy(x1,y1) - Hy(x1-1,y1));
Ezy(x1,y1) = cezye(x1,y1).*Ezy(x1,y1) + ...
            cezyhx(x1,y1).*(Hx(x1,y1) - Hx(x1,y1-1));
Ez(x1,y1) = Ezx(x1,y1) + Ezy(x1,y1);

```

```

%=====
=====
%=====<Insertando
Fuente>=====
    ma      = ((n-to)/tao)*((n-to)/tao);
    source  = Am*(n-to)*exp(-ma);
    Ez(ic,jc) = source;
    MonitorEzpml(n) = Ez(ic+8,jc+8);
%=====
=====
%=====<Mostrar
Grafica>=====
    if setting(1) == true
        set(h, 'ZData', Ez)
        drawnow;
    end
%=====
=====
%=====<Actualizando Campos
Magneticos>=====
    Hx(x2,y2) = chxh(x2,y2).*Hx(x2,y2) + chxez(x2,y2).*(Ez(x2,y2+1) - Ez(x2,y2));
    Hy(x2,y2) = chyh(x2,y2).*Hy(x2,y2) + chyez(x2,y2).*(Ez(x2+1,y2) - Ez(x2,y2));
%=====
=====
%=====<Almacenando
Datos>=====
    if setting(2)== true
        EEZ(:, :,n) = Ez;
        HHX(:, :,n) = Hx;
        HHY(:, :,n) = Hy;
    end

```

```

end
%=====
=====
%===== <Generando
Ficheros>=====
if setting(2)==true
    for n=1:nsteps
        Energiapml(n) = sum(sum((eo/2)*(EEZ(xe,ye,n).*EEZ(xe,ye,n))+...
            (uo/2)*(HHX(xe,ye,n).*HHX(xe,ye,n) + HHY(xe,ye,n).*HHY(xe,ye,n))));
    end
    Energiapml = (Energiapml/max(Energiapml));
    fid1 = fopen(strcat('MonitorEzpml',int2str(npml),'.txt'),'w');
    fid2 = fopen(strcat('Energiapml',int2str(npml),'.txt'),'w');
    for i=1:nsteps
        fprintf(fid1,'%f\t\n',MonitorEzpml(i));
        fprintf(fid2,'%f\t\n',Energiapml(i));
    end
    fclose(fid1);
    fclose(fid2);
end
disp('successful')

```

ANEXO A.3 CONDICIÓN DE FRONTERA ABSORBENTE CPML

%Modo CPMLTMZ

```

clear all; clc ; close all;
disp('run cpml...')
setting = [0 1]; % {[Motrar Pelicula,Generar ficheros}
%=====
===== %===== <Parametros
FDTD>=====
xx = 36;
yy = 36;
nsteps = 1e4;
eo = 8.854e-12;
uo = 4*pi*1e-7;
n0 = sqrt(uo/eo);
ic = (xx)/2;
jc = (yy)/2;
dx = .001;
dy = .001;
ds = dx;
co = 1/sqrt(eo*uo);
S = 0.5/sqrt(2);
dt = S*ds/co;
xend = xx + 1;
yend = yy + 1;
x1 = 2:xx-1;
xe = 4:xx-4;
y1 = 2:yy-1;
ye = 4:yy-4;
x2 = 1:xx-1;
y2 = 1:yy-1;
%=====
===== %===== <Parametros
CPML>=====

```

```

m = 2;
npml = 4;
%R = 1e-8;
kmax = 7;
ph_max = 0.05;
ph_min = 0;
%9.1679
sigma_factor = 1.5;
sigma_max = sigma_factor*(m+1)/(150*pi*ds);
sigma1_max = (uo/eo)*sigma_max;
clear R
%=====
===== <Definiendo perfil de
perdidas>=====
sigma_mx = zeros(xend,yend);
sigma_ex = zeros(xend,yend);
kex = ones(xend,yend);
kmx = ones(xend,yend);
ph_ex = ph_max*ones(xend,yend);
ph_mx = ph_max*ones(xend,yend);
for j=1:npml
    sigma_mx(j,:)= sigma1_max*((npml+1.0-j)/(npml+0.5))^m;
    kmx(j,:) = 1 + (kmax - 1)*((npml+1.0-j)/(npml+0.5))^m;
    ph_mx(j,:) = (uo/eo)*...
                (ph_min +(ph_max - ph_min)*(1-(npml+1.0-j)/(npml+0.5)));
end
for j=xx-npml:xx
    sigma_mx(j,:)= sigma1_max*((npml+1.0-xx+j)/(npml+0.5))^m;
    kmx(j,:) = 1 + (kmax - 1)*((npml+1.0-xx+j)/(npml+0.5))^m;
    ph_mx(j,:) = (uo/eo)*...
                (ph_min +(ph_max - ph_min)*(1-(npml+1.0-xx+j)/(npml+0.5)));

```

```

end
for j=1:npml+1
    sigma_ex(j,:)= sigma_max*((npml+1.5-j)/(npml+0.5))^m;
    kex(j,:)    = 1 + (kmax - 1)*((npml+1.5-j)/(npml+0.5))^m;
    ph_ex(j,:)  = ...
                (ph_min +(ph_max - ph_min)*(1-(npml+1.5-j)/(npml+0.5)));
end
for j=xx-npml:xx
    sigma_ex(j,:)= sigma_max*((j-xx+npml+0.5)/(npml+0.5))^m;
    kex(j,:)    = 1 + (kmax - 1)*((j-xx+npml+0.5)/(npml+0.5))^m;
    ph_ex(j,:)  = ...
                (ph_min +(ph_max - ph_min)*(1-(j-xx+npml+0.5)/(npml+0.5)));
end
clear m
sigma_my = sigma_mx';
sigma_ey = sigma_ex';
ph_my   = ph_mx';
ph_ey   = ph_ex';
key     = kex';
kmy     = kmx';

aex = (sigma_ex./(sigma_ex.*kex + kex.*kex.*ph_ex))...
      .*( exp(-(sigma_ex./kex + ph_ex)*dt/eo) - 1 );
aey = (sigma_ey./(sigma_ey.*key + key.*key.*ph_ey))...
      .*( exp(-(sigma_ey./key + ph_ey)*dt/eo) - 1 );
ahx = (sigma_mx./(sigma_mx.*kmx + kmx.*kmx.*ph_mx))...
      .*( exp(-(sigma_mx./kmx + ph_mx)*dt/uo) - 1 );
ahy = (sigma_my./(sigma_my.*kmy + kmy.*kmy.*ph_my))...
      .*( exp(-(sigma_my./kmy + ph_my)*dt/uo) - 1 );

bex = exp(-(sigma_ex./kex + ph_ex)*dt/eo);

```

```

bey = exp(-(sigma_ey./key + ph_ey)*dt/eo);
bhx = exp(-(sigma_mx./kmx + ph_mx)*dt/uo);
bhy = exp(-(sigma_my./kmy + ph_my)*dt/uo);
clear sigma_ex sigma_ey sigma_mx sigma_my
%=====
===== <Iniciando
Campos>=====
Ez = zeros(xend,yend);
Pey = zeros(xend,yend);
Pex = zeros(xend,yend);

Hx = zeros(xend,yend);
Phx = zeros(xend,yend);

Hy = zeros(xend,yend);
Phy = zeros(xend,yend);
%===== <...Iniciando
Matrices...>=====
%===== <.....de
Medida.....>=====
if setting(2)== true
    Energiacpml = zeros(1,nsteps);
    MonitorEzcpml = zeros(nsteps,1);
    EEZ = zeros(xend,yend,nsteps);
    HHX = zeros(xend,yend,nsteps);
    HHY = zeros(xend,yend,nsteps);
end
%=====
===== %===== <Parametros de la
Fuente>=====
nc = 20;

```

```

dsmax = ds;
tao = ((nc*dsmax)/(2*co))/dt;
to = 4.5*tao;
fmax = co/(nc*dsmax);
Am = -(sqrt(2*exp(1))/tao);
%=====
===== %===== <Parametros de la
Grafica>=====
if setting(1) == true
    afFigurePosition = [500 100 300.0 300.0];
    h=surf(Ez);
    shading interp;
    colormap(jet(1000))
    lighting gouraud
    view(90,90)
    caxis ([0 0.5])
    axis ([0 yy 0 xx -1 1])
    xlabel('N_{x}','FontSize',14)
    ylabel('N_{y}','FontSize',14)
    zlabel('E_{\zeta}','FontSize',14)
    title({'\fontsize{14}Propagation Mode TM_{z}';...
        '\fontsize{14}(CPML case)'})
    set(gcf,'Position',afFigurePosition);
    axis off
    grid on
    hold on
end
clear afFigurePosition
clear xx yy
window = 20;

```

```

%=====
===== %===== <Run
Time>=====
for n = 1:nsteps
%=====
===== %===== <Actualizando Campos
Electricos>=====
    Pey(x1,y1)= bey(x1,y1).*Pey(x1,y1) + ...
                aey(x1,y1).*(Hx(x1,y1) - Hx(x1,y1-1))/dy;
    Pex(x1,y1)= bex(x1,y1).*Pex(x1,y1) + ...
                aex(x1,y1).*(Hy(x1,y1) - Hy(x1-1,y1))/dx;

    Ez(x1,y1) = Ez(x1,y1)...
        - dt/(eo*dy)*(Hx(x1,y1) - Hx(x1,y1-1))./key(x1,y1)...
        + dt/(eo*dx)*(Hy(x1,y1) - Hy(x1-1,y1))./kex(x1,y1)...
        + dt/eo*(Pex(x1,y1) - Pey(x1,y1));
%=====
===== <Insertando
Fuente>=====
    ma      =((n-to)/tao)*((n-to)/tao);
    source  = Am*(n-to)*exp(-ma);
    Ez(ic,jc) = source;
    MonitorEzcpml(n) = Ez(ic+8,jc+8);
%=====
===== <Mostrar
Grafica>=====
if setting(1) == true
    set(h, 'ZData', Ez)
    drawnow;
end

```

```

%=====
=====
%=====<Actualizando Campos
Magneticos>=====
Phy(x2,y2)= bhy(x2,y2).*Phy(x2,y2) + ...
            ahy(x2,y2).*(Ez(x2,y2+1) - Ez(x2,y2))/dy;
Phx(x2,y2)= bhx(x2,y2).*Phx(x2,y2) + ...
            ahx(x2,y2).*(Ez(x2+1,y2) - Ez(x2,y2))/dx;

Hx(x2,y2) = Hx(x2,y2) - dt/(dy*uo)*(Ez(x2,y2+1)...
            - Ez(x2,y2))./kmy(x2,y2) - dt/uo*Phy(x2,y2);
Hy(x2,y2) = Hy(x2,y2) + dt/(dx*uo)*(Ez(x2+1,y2)...
            - Ez(x2,y2))./kmx(x2,y2)+ dt/uo*Phx(x2,y2);

%=====
=====
%=====<Almacenando
Datos>=====
if setting(2)== true
    EEZ(:,n) = Ez;
    HHX(:,n) = Hx;
    HHY(:,n) = Hy;
end
end
clear Ez Hy Hx

%=====
=====<Generando
Ficheros>=====
if setting(2)==true
    for n=1:nsteps
        Energiacpml(n) = sum(sum((eo/2)*(EEZ(xe,ye,n).*EEZ(xe,ye,n))+...
            (uo/2)*(HHX(xe,ye,n).*HHX(xe,ye,n) + HHY(xe,ye,n).*HHY(xe,ye,n))));
    end
end

```

```

Energiacpml = (Energiacpml/max(Energiacpml));
fid1 = fopen(strcat('MonitorEzcpml',int2str(npml),'.txt'),'w');
fid2 = fopen(strcat('Energiacpml',int2str(npml),'.txt'),'w');
for i=1:nsteps
    fprintf(fid1,'%f\t\n',MonitorEzcpml(i));
    fprintf(fid2,'%f\t\n',Energiacpml(i));
end
fclose(fid1);
fclose(fid2);
end
disp('successful')

```

ANEXO A.4 PLATAFORMA DE SIMULACION

```

clc,clear all,close all

```

```

disp('run plots...')

```

```

%Elija entre las siguientes opciones [mur,pml,{cpml}]...

```

```

strViewGoals = {'mur';'pml4';'cpml4'};

```

```

%Elija entre las siguientes opciones true:Reflexion , false: Energia

```

```

HowtoPlots = false;

```

```

strDefaultCondition = 'ref';

```

```

%Vector booleano para elegir grafica a mostrar en pantalla:

```

```

%[{TimeSignal} , FrequencySignal, ErrorTSignal, ErrorFsignal]...

```

```

vcShowGraphics = [1 0 1 1];

```

```

%-----

```

```

vcSizeString = size(strViewGoals);

```

```

vcIterationFile = 1:vcSizeString(1);

```

```

if HowtoPlots == true
    FILE = strcat('MonitorEz',strViewGoals(vclterationFile,:),'.txt');
    strDefaultName = strcat('MonitorEz',strDefaultCondition, '.txt');
else
    FILE = strcat('Energia',strViewGoals(vclterationFile,:),'.txt');
    strDefaultName = strcat('Energia',strDefaultCondition, '.txt');
end

%Estableciendo propiedades de plots...
afPlotLineWidth = [2.0, 1.3];
astrPlotLineStyle = [{'_.'},{:}, {'--'}];
aafPlotLineColor = ['b';'r';'g'];
vcXTick = linspace(0,2,5);

%Estableciendo posicion y dimensiones de la figura sobre la pantalla:
%[pos_x pos_y width_x width_y].
afFigurePosition = [[0.5 1.2 20.0 10.0];[18.0 1.0 20.0 10.0];...
                    [0.5 7.0 20.0 10.0];[18.0 7.0 20.0 10.0]];

%Estableciendo propiedades de axes...
iFontSize = 20;
strFontUnit = 'points'; %[{points}]normalized[inches|centimeters|pixels]
strFontName = 'Times' ; %[{Times}]Courier]
strFontWeight = 'normal'; %[light|normal|demi|bold]
strFontAngle = 'normal'; %[{normal}]italic|oblique]
strInterpreter = 'latex'; %[{tex}]|latex]
fLineWidth = 2.0; %width of the line of te axes
strLineWidth = 2.0;

%-----
%Definiendo parametros...
if HowtoPlots
    nsteps = 900;
else

```

```

nsteps = 1e4;
end
eo = 8.854e-12;    %permitividad electrica del vacio.
uo = 4*pi*1e-7;   %permeabilidad magnetica del vacio.
ds = .001;        %tamaño de la celda.
co = 1/sqrt(eo*uo);
S = 0.5/sqrt(2);  %Parametro de Courant
Ts = S*ds/co;     %Paso de tiempo.
Fs = 1/Ts;
L = 10*nsteps;    %Longitud de la onda
NFFT = 2^nextpow2(L);
f = Fs/2*linspace(0,1,NFFT/2+1);

factor.ns = 1e-9;
factor.StrTn = 'ns';
factor.StrFn = 'nHz';
factor.M = 1e6;
factor.StrTM = 'Ms';
factor.StrFM = 'MHz';
factor.G = 1e9;
factor.StrTG = 'Gs';
factor.StrFG = 'GHz';
factor.T = 1e12;
factor.StrTT = 'Ts';
factor.StrFT = 'THz';

axesTime = ((0:nsteps-1)*Ts)/factor.ns ;
axesFrecuency = f/factor.G;

Data = struct('Value',[],'Name',[],'Color',[],'Tfourier',[]);
Default = struct('Value',load(strDefaultName),'Tfourier',[],...

```

```
'Name',strDefaultCondition,'Color','k');
```

```
for i= vcIterationFile
```

```
    Data(i).Value = load(FILE {i,1});
```

```
    Data(i).Name = strViewGoals{i};
```

```
...    Data(i).Color = aafPlotLineColor(i,:);
```

```
    if HowtoPlots
```

```
        Data(i).Tfourier = fft(Data(i).Value,NFFT)/100;
```

```
        Data(i).Tfourier = 2*abs(Data(i).Tfourier(1:NFFT/2+1));
```

```
    end
```

```
end
```

```
if HowtoPlots == true
```

```
    mxabcT = zeros(1,vcSizeString(1));
```

```
    mxabcF = zeros(1,vcSizeString(1));
```

```
    error2T = zeros(1,vcSizeString(1));
```

```
    error2F = zeros(1,vcSizeString(1));
```

```
else
```

```
    por100u = zeros(1,vcSizeString(1));
```

```
end
```

```
if HowtoPlots
```

```
    Default.Tfourier = fft(Default.Value,NFFT)/100;
```

```
    Default.Tfourier = 2*abs(Default.Tfourier(1:NFFT/2+1));
```

```
    Error = struct('Time',[],'Name',[],'Color',[],'Frec',[]);
```

```
    for i=vcIterationFile
```

```
        Error(i).Time = 20*log10(abs(Data(i).Value - Default.Value)...  
            /max(abs(Default.Value)));
```

```
        Error(i).Name = strViewGoals {i};
```

```

Error(i).Color = aafPlotLineColor(i,:);
Error(i).Frec = 20*log10(abs(Data(i).Tfourier - Default.Tfourier )...
    /max(Default.Tfourier));
mxabcT(i) = max(Error(i). Time);
error2T(i)= 20*log(mean(abs(Data(i).Value - Default.Value)...
    /max(abs(Default.Value))));
error2F(i)= 20*log(mean(abs(Data(i).Tfourier(1:388) -
Default.Tfourier(1:388) )...
    /max(Default.Tfourier(1:388) ));
mxabcF(i) = max(Error(i).Frec);

end
else
for i=vcIterationFile
    por100u(i) = (max(Data(i).Value - min(Data(i).Value))/nsteps)*100;
end
end

%-----%.....Time
Signal Plot.....
%-----

if vcShowGraphics(1) == true
switch vcSizeString(1)
case 1
    STime = plot(axesTime',Data(1).Value,Data(1).Color,'-.b',...
        axesTime',Default.Value,Default.Color);
    legend(Data(1).Name,Default.Name);
case 2
    STime = plot(axesTime',Data(1).Value,Data(1).Color,...
        axesTime',Data(2).Value,Data(2).Color);
    legend(Data(1).Name,Data(2).Name);

```

```

case 3
    STime = plot(axesTime',Data(1).Value,'!',...
                axesTime',Data(2).Value,'-!',...
                axesTime',Data(3).Value,'--');
    legend(Data(1).Name,Data(2).Name,Data(3).Name);
otherwise
    STime = plot(axesTime',Data(1).Value,'-*',...
                axesTime',Data(2).Value,'-!',...
                axesTime',Data(3).Value,'--',...
                axesTime',Data(4).Value,'!');
    legend(Data(1).Name,Data(2).Name,Data(3).Name,Data(4).Name);

end
settingPlots(STime);
% axis([0 axesTime(end) min(Default.Value) max(Default.Value)])
xlabel(strcat('Tiempo','(',factor.StrTn,')'))
if HowtoPlots == true
    ylabel('E_{\zeta}')
else
    ylabel('dB(\phi)')
end
end
%-----
%.....Frecuency Signal Plot.....
%-----

if vcShowGraphics(2)&& HowtoPlots
    figure
    switch vcSizeString(1)
    case 1
        SFrec = plot(axesFrecuency,Data(1).Tfourier,Data(1).Color,...

```

```

        axesFrequency,Default.Tfourier,Default.Color);
    legend(Data(1).Name,Default.Name);
case 2
    SFrec = plot(axesFrequency,Data(1).Tfourier,Data(1).Color,...
        axesFrequency,Data(2).Tfourier,Data(2).Color);
    legend(Data(1).Name,Data(2).Name);
otherwise
    SFrec = plot(axesFrequency,Data(1).Tfourier,'!',...
        axesFrequency,Data(2).Tfourier,'-!',...
        axesFrequency,Data(3).Tfourier,'--');

    legend(Data(1).Name,Data(2).Name,Data(3).Name);
end

settingPlots(SFrec);
xlabel(strcat('Frecuencia','(',factor.StrFG,')'))
ylabel('|Y(f)|')
axis([0 axesFrequency(end) min(Default.Tfourier) max(Default.Tfourier)])
end

%-----%.....Error
Time Signal Plot.....
%-----

if vcShowGraphics(3)&& HowtoPlots
    figure
    switch vcSizeString(1)
        case 1
            ErrorT = plot(axesTime',Error(1).Time,Error(1).Color);
            legend(Error(1).Name);
        case 2
            ErrorT = plot(axesTime',Error(1).Time,'!',...
                axesTime',Error(2).Time,'-!');

```

```

        legend(Error(1).Name,Error(2).Name);
    otherwise
        SerrorT = plot(axesTime',Error(1).Time,'!',...
            axesTime',Error(2).Time,'-!',...
            axesTime',Error(3).Time,'--');
        legend(Error(1).Name,Error(2).Name,Error(3).Name);
    end
    settingPlots(SerrorT);
    xlabel(strcat('Tiempo','(',factor.StrTn,')'))
    ylabel('Error_{t}(DB)')
end
%-----%.....Error
Frec Signal Plot.....
%-----
if vcShowGraphics(4)&& HowtoPlots
    figure
    switch vcSizeString(1)
        case 1
            SerrorF = plot(axesFrecuency,Error(1).Frec,Error(1).Color);
            legend(Error(1).Name);

        case 2
            SerrorF = plot(axesFrecuency,Error(1).Frec,'!',...
                axesFrecuency,Error(2).Frec,'-!');
            legend(Error(1).Name, Error(2).Name);
        otherwise
            SerrorF = plot(axesFrecuency,Error(1).Frec,'!',...
                axesFrecuency,Error(2).Frec,'-!',...
                axesFrecuency,Error(3).Frec,'--');
            legend(Error(1).Name, Error(2).Name,Error(3).Name);
        end
    end
end

```

```

    settingPlots(SerrorF);
    xlabel(strcat('Frecuencia','(',factor.StrFG,')'))
    ylabel('Error_{f}(DB)')
end
disp('successful')

```

ANEXO A.5 DOMINIO COMPUTACIONAL DE REFERENCIA

```

%Modo TMZ PML-Berenger
clear all; clc ; close all;
disp('run ref...')
strChooseRef ='pec';
setting = [0 1]; % {[Motrar Pelicula,Generar ficheros}
%=====
=====
%===== <Parametros
FDTD>=====
if strcmp(strChooseRef,'pec')
    nx = 32;
    ny = 32;
elseif strcmp(strChooseRef,'ref')
    nx = 600;
    ny = 600;
else
    nx = 600;
    ny = 600;
end
nsteps = 1e4;
eo = 8.854e-12;

```

```

uo = 4*pi*1e-7;
n0 = sqrt(uo/eo);
ic = (nx)/2;
jc = (ny)/2;
dx = .001;
dy = .001;
ds = dx;
co = 1/sqrt(eo*uo);
S = 0.5/sqrt(2);
dt = S*ds/co;
x1 = 2:nx-1;
y1 = 2:ny-1;
x2 = 1:nx-1;
y2 = 1:ny-1;
%=====
=====
%===== <Definiendo perfil de
perdidas>=====
sigma_m = zeros(nx,ny);
sigma_e = zeros(nx,ny);
%=====
===== %===== <Definiendo Coeficientes de los
Campos>=====
chyh = (2*uo-dt*sigma_m)./(2*uo+dt*sigma_m);
chyez = (2*dt)./((2*uo+dt*sigma_m)*ds);
chxh = chyh';
chxez = -chyez';

ceze = (2*eo-dt*sigma_e)./(2*eo+dt*sigma_e);
cezhy = (2*dt)./((2*eo+dt*sigma_e)*ds);
cezhx = -(2*dt)./((2*eo+dt*sigma_e)*ds)';

```

```

%=====
===== %===== <Iniciando
Campos>=====
Ez = zeros(ny,nx);
Hx = zeros(ny,nx);
Hy = zeros(ny,nx);
%=====
===== %===== <Parametros de la
Fuente>=====
nc = 20;
dsmax = ds;
tao = ((nc*dsmax)/(2*co))/dt;
to = 4.5*tao;
fmax = co/(nc*dsmax);
Am = -(sqrt(2*exp(1))/tao);
%===== <...Iniciando
Matrices...>=====
%===== <.....de
Medida.....>=====
if setting(2)== true
    MonitorEzref = zeros(nsteps,1);
    if strcmp(strChooseRef,'pec')
        Energiaref = zeros(1,nsteps);
        EEZ = zeros(nx,ny,nsteps);
        HHX = zeros(nx,ny,nsteps);
        HHY = zeros(nx,ny,nsteps);
    end
end
%=====
=====

```

```

%===== <Parametros de la
Grafica>=====
if setting(1) == true
    afFigurePosition = [500 100 300.0 300.0];
    h=surf(Ez);
    shading interp;
    colormap(jet(1000))
    lighting gouraud
    view(90,90)
    caxis ([0 0.5])
    axis ([0 nx 0 ny -1 1])
    xlabel('N_{x}','FontSize',14)
    ylabel('N_{y}','FontSize',14)
    zlabel('E_{zeta}','FontSize',14)
    title({'\fontsize{14}Propagation Mode TM_{z}';...
        '\fontsize{14}(REF case)'})
    set(gcf,'Position',afFigurePosition);
    axis off
    grid on
    hold on
end
window = 20;
%=====
===== <Run
Time>=====
for n = 1:nsteps
%=====
===== <Actualizando Campos
Electricos>=====
    Ez(x1,y1) = ceze(x1,y1).*Ez(x1,y1)...
        + cezhy(x1,y1).*(Hy(x1,y1) - Hy(x1-1,y1))...

```

```

+ cezhx(x1,y1).*(Hx(x1,y1) - Hx(x1,y1-1));
%=====
===== <Insertando
Fuente>=====
    ma      = ((n-to)/tao)*((n-to)/tao);
    source  = Am*(n-to)*exp(-ma);
    Ez(ic,jc) = source;
    MonitorEzref(n) = Ez(ic+8,jc+8);
%=====
===== <Mostrar
Grafica>=====
    if setting(1) == true
        set(h, 'ZData', Ez)
        drawnow;
    end
%=====
===== <Actualizando Campos
Magneticos>=====
    Hx(x2,y2) = chxh(x2,y2).*Hx(x2,y2) + chxez(x2,y2).*(Ez(x2,y2+1) - Ez(x2,y2));
    Hy(x2,y2) = chyh(x2,y2).*Hy(x2,y2) + chyez(x2,y2).*(Ez(x2+1,y2) - Ez(x2,y2));
%=====
===== <Almacenando
Datos>=====
    if setting(2)== true
        if strcmp(strChooseRef,'pec')
            EEZ(:,n) = Ez;
            HHX(:,n) = Hx;
            HHY(:,n) = Hy;
        end
    end
end
end
end

```

```

%=====
=====
%=====<Generando
Ficheros>=====
if setting(2)==true
    if strcmp(strChooseRef,'pec')
        for n=1:nsteps
            Energiaref(n) = sum(sum((eo/2)*(EEZ(:,:,n).*EEZ(:,:,n))+...
                (uo/2)*(HHX(:,:,n).*HHX(:,:,n) + HHY(:,:,n).*HHY(:,:,n))));
        end
        Energiaref = (Energiaref/max(Energiaref));
        fid2 = fopen('Energiaref.txt','w');
        for i=1:nsteps
            fprintf(fid2,'%f\t\n',Energiaref(i));
        end
        fclose(fid2);
    end
    fid1 = fopen('MonitorEzref.txt','w');
    for i=1:nsteps
        fprintf(fid1,'%f\t\n',MonitorEzref(i));
    end
    fclose(fid1);
end
disp('successful')

```