

ESCUELA POLITÉCNICA NACIONAL

**FACULTAD DE INGENIERÍA ELÉCTRICA Y
ELECTRÓNICA**

**DESARROLLO DE UNA HERRAMIENTA DE SOFTWARE PARA LA
EVALUACIÓN DE CAMPOS ELECTROSTÁTICOS EN
ESTRUCTURAS COMPLEJAS**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
INGENIERO ELÉCTRICO**

STALIN FABRICIO PINOS ORDÓÑEZ

stalin_pinos@hotmail.com

DIRECTOR: Dr. FRANKLIN LENIN QUILUMBA GUDIÑO

franklin.quilumba@epn.edu.ec

Quito, Noviembre 2017

AVAL

Certifico que el presente trabajo fue desarrollado por Stalin Fabricio Pinos Ordóñez, bajo mi supervisión.

Dr. FRANKLIN LENIN QUILUMBA GUDIÑO
DIRECTOR DEL TRABAJO DE TITULACIÓN

DECLARACIÓN DE AUTORÍA

Yo, Stalin Fabricio Pinos Ordóñez, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Stalin Fabricio Pinos Ordóñez

DEDICATORIA

Dedico el presente trabajo de titulación a quienes permanentemente me apoyaron y contribuyeron de una u otra manera para lograr las metas y objetivos planteados.

A Dios y la virgen del Quinche, por guiarme y brindarme salud para lograr mis objetivos.

A mis padres Rodrigo y Digna, por su apoyo incondicional y sobre todo por su amor.

A mis hermanos Sebastián y Dayanna por estar presentes en todas las etapas de mi vida, los quiero mucho.

A mis abuelitos, tíos, primos y amigos, gracias por su apoyo y consejos.

A mis maestros por enseñarme que los objetivos se logran con esfuerzo.

De igual manera dedico este triunfo a dos personas muy especiales. A mí hijo Aiden por acompañarme estos últimos años y desprender luz en mi vida que me brinda fuerza cada día. A mí enamorada Anabel por su comprensión y por ser la mejor mamá que mi hijo puede tener.

AGRADECIMIENTO

A Dios y la virgen del Quinche, por guiarme y haberme permitido llegar hasta este punto, por brindarme salud para lograr mis objetivos.

A mis padres Rodrigo y Digna, por su apoyo incondicional, por los ejemplos de perseverancia y constancia que han infundido siempre en mí, por sus consejos, sus valores, sus enseñanzas que han permitido convertirme en una persona de bien y sobre todo por su amor.

A mis hermanos Sebastián y Dayanna, por su apoyo en todo momento, por haber estado presentes en todas las etapas de mi vida, por las alegrías, anécdotas, discusiones que vivimos juntos, los quiero mucho.

A mi hijo Aiden por ser la fuerza que me ayuda a seguir adelante cada día, por ser la responsabilidad y compromiso más duro y más bonito que he tenido que pasar.

A mí enamorada Anabel por su comprensión hacia mí y hacia nuestro hijo.

A mis abuelitos, tíos, primos y amigos que estuvieron pendientes desde el principio hasta el fin de este camino y que ahora lo estoy culminando, gracias por su apoyo y consejos.

A mis maestros que, desde la escuela, luego en el colegio y en la universidad me enseñaron que los objetivos se logran con esfuerzo, a todos ellos un Dios les pague.

De igual forma agradecer al personal docente, administrativo y a todos los que conforman la Escuela Politécnica Nacional en especial al Msc. Fausto Valencia y al Dr. Franklin Quilumba por su colaboración en el proceso de aprendizaje, por la amistad brindada y por haberme ofrecido las facilidades necesarias para realizar este trabajo de titulación.

Gracias.

ÍNDICE DE CONTENIDO

AVAL	I
DECLARACIÓN DE AUTORÍA.....	II
DEDICATORIA.....	III
AGRADECIMIENTO.....	IV
ÍNDICE DE CONTENIDO.....	V
RESUMEN	VIII
ABSTRACT	VIIIX
1. INTRODUCCIÓN	1
1.1 Objetivos.....	2
1.2 Alcance.....	3
1.3 Marco Teórico.....	3
1.3.1 Ecuaciones de Poisson y de Laplace	3
1.3.2 Expansión en series de Taylor.....	6
1.3.2.1 Proximidad de la primera derivada mediante diferencias centradas	7
1.3.2.2 Proximidad de la segunda derivada mediante diferencias centradas ..	8
1.3.2.3 Derivadas en dos dimensiones	9
1.3.2.4 Derivadas en tres dimensiones	10
1.3.3 Limitaciones en la solución analítica para resolver la ecuación de Laplace	12
2. MÉTODO DE LAS DIFERENCIAS FINITAS GAUSS – SEIDEL Y SU IMPLEMENTACIÓN EN PYTHON 3.5	14
2.1 Método de las diferencias finitas	14
2.1.1 Aproximación de la derivada parcial en términos algebraicos de voltaje	14
2.1.2 Método de Gauss - Seidel	17
2.1.3 Método de relajación	19
2.2 Implementación del método de las diferencias finitas en Python 3.5.....	21
2.2.1 Condiciones de frontera.....	22
2.2.2 Algoritmo del método de las diferencias finitas Gauss – Seidel con método de relajación.....	26
2.2.3 Campo Electrostático.....	29
3. FUNCIONAMIENTO DE LA HERRAMIENTA DE SOFTWARE.....	35
3.1 Interacción Usuario – Herramienta de software.....	35
3.2 Funcionamiento del Complemento en Excel	36

3.3	Funcionamiento del código fuente	37
4.	EJEMPLOS DE APLICACIÓN Y ANÁLISIS DE RESULTADOS	38
4.1	Complicaciones de utilizar el método de las diferencias finitas en estructuras complejas	38
4.1.1	Implementación de un círculo	38
4.1.2	Implementación de una circunferencia	39
4.2	Ejemplos de aplicación	41
4.2.1	Implementación de la herramienta de software a un problema simétrico de teoría electromagnética	41
4.2.2	Implementación de la herramienta de software en la parte transversal de un cable coaxial	46
4.2.2.1	Cable de un núcleo conductor y un material aislante	46
4.2.2.2	Cable con un núcleo conductor y dos capas de aislamiento	52
4.2.3	Implementación de la herramienta de software a un problema asimétrico de teoría electromagnética	56
4.2.3.1	Representación del problema.....	56
4.2.3.2	Datos ingresados a la herramienta de software	57
4.2.3.3	Resultados	58
4.2.4	Implementación de la herramienta de software a un transformador monofásico de potencia	66
4.2.4.1	Datos Generales del Transformador	67
4.2.4.1.1	Valores nominales.....	67
4.2.4.1.2	Dimensiones de la geometría interna del transformador	67
4.2.4.2	Modelación del transformador	68
4.2.4.2.1	Forma gráfica del problema.....	68
4.2.4.2.2	Voltajes en los devanados.....	70
4.2.4.3	Datos ingresados a la herramienta de software	70
4.2.4.4	Resultados	71
5.	CONCLUSIONES Y RECOMENDACIONES.....	81
5.1	Conclusiones	81
5.2	Recomendaciones.....	82
6.	REFERENCIAS BIBLIOGRÁFICAS	83
7.	ANEXOS.....	84
	ANEXO I.....	84
	USO DE CEDIF	84
	ANEXO II.....	92

CÓDIGO DE PROGRAMA EN PYTHON 3.5	92
ORDEN DE EMPASTADO	103

RESUMEN

En el presente trabajo de titulación se desarrolla una herramienta de software, denominada Campo Electroestático con Diferencias Finitas (CEDIF), mediante el lenguaje de programación Python 3.5, con el fin de obtener el comportamiento del campo eléctrico en estado estacionario para entornos genéricos bidimensionales y tridimensionales.

Los datos geométricos, de voltaje y permitividades que caracterizan el problema son almacenados en matrices localizadas en el código de programa, estos datos se muestran de forma gráfica tal como el usuario ingresó en Excel. De la misma manera los datos son utilizados para determinar superficies equipotenciales resolviendo la ecuación de Laplace mediante el método de las diferencias finitas, obteniendo la distribución del voltaje en el entorno.

El campo eléctrico se encuentra mediante el gradiente de potencial y los resultados se muestran de forma gráfica obteniendo la distribución del campo electrostático en entornos de dos y tres dimensiones.

Ejemplos bidimensionales se implementan y los resultados se comparan con respuestas analíticas, con libros citados en la bibliografía y con simulaciones efectuadas en el programa computacional FEMM. Para las aplicaciones tridimensionales se efectúan dos ejemplos, el primero para un problema asimétrico de teoría electromagnética y el segundo para el interior de un transformador de potencia.

PALABRAS CLAVE: Campos electromagnéticos, diferencias finitas, ecuación de Laplace, métodos computacionales, software, Python.

ABSTRACT

In this work, a development of a software tool, called Electrostatic Field with Finite Differences (CEDIF), through the programming language Python 3.5 is presented. Its main functionality is the determination of the behavior of electrostatic fields for bidimensional and tridimensional systems.

The geometric data of the problem, and the voltage and permittivity that characterize the medium are stored in a matrix located in the program's code. The data are shown as a graph the same as it was entered by the user in an Excel file. Likewise, these data are used to determine the equipotential surfaces by solving the Laplace equation through the finite differences method.

The electric field is found through the gradient potential. The electrostatic field is shown as a graphic for two or three dimensions problems.

Bidimensional examples are implemented and the results are compared with analytical solutions. Two examples for tridimensional applications are shown. The first one is for an asymmetrical electromagnetic problem and the second one is the analysis of the potential field inside a power transformer.

KEYWORDS: Electromagnetic fields, finite differences, Laplace's equation, computational methods, software, Python.

1. INTRODUCCIÓN

Todos los métodos utilizados para determinar el comportamiento del campo eléctrico son válidos, la diferencia radica en la complejidad de su utilización. En otras palabras, el método para establecer la distribución del campo depende de la complicación del problema. Para ilustrar, problemas sencillos se pueden resolver mediante métodos analíticos; problemas simétricos con disposición geométrica del entorno por medio del método de las imágenes; problemas con distintos niveles de voltaje, entornos complejos y abundantes datos, a través de métodos numéricos, etc.

El análisis numérico para solucionar problemas es cada día más utilizado debido a que permite simular procesos matemáticos complejos aplicados a conocimientos de ingeniería. [1].

Generalmente el análisis del comportamiento del campo eléctrico ha estado basado en la utilización de programas que aplican métodos numéricos. Tal es el caso del programa computacional FEMM que usa el método numérico de elementos finitos limitado a dos dimensiones [2], o el programa ANSYS Maxwell, que emplea el mismo método para dos y tres dimensiones, pero es de tipo comercial. Este último software de simulación es destacado en el sector del diseño, análisis de motores eléctricos, sensores, transformadores, dispositivos electromagnéticos y electromecánicos. Por ejemplo, se ha utilizado para optimizar el equilibrio entre las pérdidas de ventiladores y el rendimiento electromagnético en motores eléctricos [3].

Existen métodos numéricos y ciertas ecuaciones que requieren de superficies cerradas para solucionar problemas matemáticos. Estas superficies se establecen como condiciones de frontera y pueden convertirse en un problema al solucionar una ecuación.

El campo eléctrico puede ser simulado aplicando el método numérico de las diferencias finitas. En él se utiliza un análisis numérico para solucionar dificultades matemáticas. Además, resuelve de forma aproximada problemas de ecuaciones diferenciales y ecuaciones en derivadas parciales [4].

La ecuación de Laplace puede expresarse mediante términos algebraicos obtenidos de las expansiones de Taylor. De esta manera la ecuación puede ser resuelta por medio del método numérico de las diferencias finitas. Existe una dificultad ya que la ecuación obtenida depende de puntos vecinos al punto en análisis. Estos puntos vecinos forman

superficies cerradas que se utilizan como condiciones de frontera. En este documento se utiliza el método numérico de las diferencias finitas para resolver la ecuación de Laplace.

La problemática que se pretende abordar radica en la dificultad de solucionar las ecuaciones de Maxwell de manera directa en entornos asimétricos, además la dificultad de encontrar el campo eléctrico y su comportamiento cuando existen electrodos complejos radica en la geometría, distribución de voltajes y el medio donde se desarrolla el problema.

Es debido a esto que, en la presente investigación, se desarrolla una herramienta de software, *Campo Electrostático con Diferencias Finitas* (CEDIF), capaz de determinar el comportamiento del campo electrostático de forma genérica en entornos bidimensionales y tridimensionales, utilizando el método numérico de las diferencias finitas, de esta manera se podrán establecer superficies equipotenciales para así aplicar el gradiente de potencial y obtener el campo eléctrico en estado estacionario.

CEDIF puede modelar varias aplicaciones en el estudio del campo eléctrico, entre las que pueden destacar: El proceder del campo eléctrico en el interior de cables monopolares o tripolares que cuentan con varias capas de aislamiento, conducta del campo eléctrico en el interior de equipos de potencia, comportamiento del campo eléctrico en sistemas donde existen equipos eléctricos que operan a ciertos niveles de voltaje.

Para la implementación de CEDIF se obtiene la ecuación de Laplace de forma general que caracteriza entornos bidimensionales y tridimensionales teniendo en cuenta la permitividad de los materiales.

En el presente documento se analiza como ejemplo el comportamiento del campo electrostático en el interior de un transformador monofásico de potencia.

1.1 Objetivos

El objetivo general de este Estudio Técnico es:

- Diseñar una herramienta de software en lenguaje Python 3.5 para la determinación del comportamiento del campo electrostático en electrodos de configuración compleja en un ambiente bidimensional y tridimensional.

Los objetivos específicos de este Estudio Técnico son:

- Identificar las limitaciones en la solución de la ecuación de Laplace cuando se usan métodos analíticos.

- Describir el método de diferencias finitas para la solución de la ecuación de Laplace en espacios bidimensionales y tridimensionales.
- Desarrollar una herramienta de software para la solución de la ecuación de Laplace mediante el método de diferencias finitas.
- Implementar la herramienta de software desarrollada para la determinación de campos electrostáticos en electrodos con configuraciones simétricas, asimétricas, y en un problema aplicativo de sistemas eléctricos.

1.2 Alcance

En este proyecto se presenta una herramienta de software desarrollada en lenguaje de programación Python 3.5 que resuelve la ecuación de Laplace en problemas electrostáticos de dos y tres dimensiones mediante la aplicación del método de Diferencias Finitas. Se aplica a un problema asimétrico de teoría electromagnética y a uno práctico de sistemas eléctricos.

1.3 Marco Teórico

En esta sección se desarrolla un análisis a la ecuación de Laplace. Se obtiene ésta de forma analítica para medios bidimensionales y tridimensionales. De la misma manera para entornos homogéneos y no homogéneos.

1.3.1 Ecuaciones de Poisson y de Laplace

La densidad de flujo eléctrico (\vec{D}) y el campo eléctrico (\vec{E}) tienen una relación que se presenta en la Ecuación 1.1.

$$\vec{D} = \varepsilon \vec{E}$$

Ecuación 1.1. Relación entre la densidad de flujo eléctrico y el campo eléctrico.

Donde:

ε : Es la permitividad eléctrica del material

Una de las ecuaciones de Maxwell para campos estáticos simbolizada por la Ecuación 1.2.

$$\nabla \vec{D} = \rho$$

Ecuación 1.2. Primera ecuación de Maxwell.

Donde:

$\nabla \vec{D}$: Es la divergencia de la densidad de flujo eléctrico (\vec{D}).

ρ : Es la densidad volumétrica de carga.

El campo eléctrico (\vec{E}) y el gradiente de potencial (∇V) se encuentran relacionados mediante la Ecuación 1.3.

$$\vec{E} = -\nabla V$$

Ecuación 1.3. Relación entre el campo eléctrico y el gradiente de potencial.

Las ecuaciones de Poisson y de la Laplace se deducen de las Ecuaciones 1.1, 1.2 y 1.3.

Al sustituir 1.1 en 1.2 se obtiene la Ecuación 1.4.

$$\rho = \nabla(\varepsilon \vec{E})$$

Ecuación 1.4. Densidad volumétrica de carga (ρ) en función de la divergencia de la multiplicación entre el campo eléctrico (\vec{E}) y la permitividad (ε).

Al sustituir 1.3 en 1.4 se obtiene la Ecuación 1.5.

$$\nabla(\varepsilon (-\nabla V)) = \rho$$

Ecuación 1.5. Densidad volumétrica de carga en función de la divergencia de la multiplicación entre la permitividad (ε) y el gradiente de potencial (∇V).

Si el medio de interés es homogéneo, entonces la permitividad (ε) sería una constante, se retira de la derivada y se obtiene la Ecuación 1.6.

$$\nabla^2 V = -\frac{\rho}{\varepsilon}$$

Ecuación 1.6. Ecuación de Poisson para medios homogéneos.

Donde:

$\nabla^2 V$: Es el laplaciano del voltaje.

Muchas veces la zona de interés está libre de carga y es homogénea (permitividad uniforme). De esta manera se obtiene la ecuación de Laplace [5] que de forma matemática se presenta en 1.7.

$$\nabla^2 V = 0$$

Ecuación 1.7. Ecuación de Laplace para medios homogéneos.

Si el medio de interés es no homogéneo, entonces no se puede excluir a la permitividad (ϵ). Si la densidad de carga volumétrica es nula ($\rho = 0$) se formaría la Ecuación 1.8 a partir de la 1.5. Esto permite que existan cargas puntuales, densidad de carga lineal y densidad de carga superficial como fuentes de campo localizadas en lugares bien definidos [6].

$$\nabla(\epsilon (-\nabla V)) = 0$$

Ecuación 1.8. Divergencia de la multiplicación entre la permitividad (ϵ) y el gradiente de potencial (∇V) cuando la densidad de carga volumétrica es cero.

La Ecuación 1.9 representa el gradiente de potencial en coordenadas cartesianas.

$$\nabla V = \frac{\partial V}{\partial x} \hat{i} + \frac{\partial V}{\partial y} \hat{j} + \frac{\partial V}{\partial z} \hat{k}$$

Ecuación 1.9. Gradiente de potencial en coordenadas cartesianas

Al reemplazar la Ecuación 1.9 en 1.8, se obtiene:

$$\nabla \left(\epsilon \left(-\frac{\partial V}{\partial x} \hat{i} - \frac{\partial V}{\partial y} \hat{j} - \frac{\partial V}{\partial z} \hat{k} \right) \right) = 0$$

Desarrollando la divergencia en coordenadas cartesianas se obtiene:

$$\frac{\partial}{\partial x} \left(\epsilon \frac{\partial V}{\partial x} \right) + \frac{\partial}{\partial y} \left(\epsilon \frac{\partial V}{\partial y} \right) + \frac{\partial}{\partial z} \left(\epsilon \frac{\partial V}{\partial z} \right) = 0$$

Utilizando la derivada de un producto:

$$\varepsilon \frac{\partial^2 V}{\partial x^2} + \frac{\partial \varepsilon}{\partial x} \frac{\partial V}{\partial x} + \varepsilon \frac{\partial^2 V}{\partial y^2} + \frac{\partial \varepsilon}{\partial y} \frac{\partial V}{\partial y} + \varepsilon \frac{\partial^2 V}{\partial z^2} + \frac{\partial \varepsilon}{\partial z} \frac{\partial V}{\partial z} = 0$$

Finalmente factorando se obtiene la ecuación de Laplace que se resolverá en este trabajo de titulación para tres dimensiones (Ecuación 1.10) y para dos dimensiones (Ecuación 1.11)

$$\varepsilon \left(\frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} + \frac{\partial^2 V}{\partial z^2} \right) + \frac{\partial \varepsilon}{\partial x} \frac{\partial V}{\partial x} + \frac{\partial \varepsilon}{\partial y} \frac{\partial V}{\partial y} + \frac{\partial \varepsilon}{\partial z} \frac{\partial V}{\partial z} = 0$$

Ecuación 1.10. Ecuación de Laplace para medios tridimensionales no homogéneos (Ecuación a solucionar).

$$\varepsilon \left(\frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} \right) + \frac{\partial \varepsilon}{\partial x} \frac{\partial V}{\partial x} + \frac{\partial \varepsilon}{\partial y} \frac{\partial V}{\partial y} = 0$$

Ecuación 1.11. Ecuación de Laplace para medios bidimensionales no homogéneos.

1.3.2 Expansión en series de Taylor

Sea $f(x)$ una función definida en (a,b) que tiene hasta la k -ésima derivada, entonces la expansión de $f(x)$ usando series de Taylor alrededor del punto x_i contenido en el interior (a,b) [7], se puede escribir como la Ecuación 1.12.

$$f(x) = f(x_i) + \frac{(x - x_i)}{1!} \left(\frac{df}{dx} \Big|_{x_i} \right) + \frac{(x - x_i)^2}{2!} \left(\frac{d^2 f}{dx^2} \Big|_{x_i} \right) + \dots + \frac{(x - x_i)^k}{k!} \left(\frac{d^k f}{dx^k} \Big|_{\varepsilon} \right)$$

Ecuación 1.12. Expansiones de Taylor.

Donde: $\varepsilon = x_i + \theta(x - x_i)$ y $0 < \theta < 1$.

Las aproximaciones de las derivadas pueden expresarse por medio de tres tipos: hacia adelante de la derivada, hacia atrás de la derivada y la diferencia central de la derivada. Esta última es a menudo la más exacta [8] y la que se utiliza en este trabajo.

1.3.2.1 Proximidad de la primera derivada mediante diferencias centradas

Para $k = 3$ en las series de Taylor y escribiendo $f(x)$ en $x = x_i + \Delta x$ (Expansión de Taylor progresiva, ε_p) y $x = x_i - \Delta x$ (Expansión de Taylor regresiva, ε_r) [7], tenemos las Ecuaciones 1.13 y 1.14.

$$f(x_i + \Delta x) = f(x_i) + \Delta x \left(\frac{df}{dx} \Big|_{x_i} \right) + \frac{\Delta x^2}{2!} \left(\frac{d^2 f}{dx^2} \Big|_{x_i} \right) + \frac{\Delta x^3}{3!} \left(\frac{d^3 f}{dx^3} \Big|_{\varepsilon_p} \right) + \dots$$

Ecuación 1.13. Expansiones de Taylor progresivas.

$$f(x_i - \Delta x) = f(x_i) - \Delta x \left(\frac{df}{dx} \Big|_{x_i} \right) + \frac{\Delta x^2}{2!} \left(\frac{d^2 f}{dx^2} \Big|_{x_i} \right) - \frac{\Delta x^3}{3!} \left(\frac{d^3 f}{dx^3} \Big|_{\varepsilon_r} \right) + \dots$$

Ecuación 1.14. Expansiones de Taylor regresivas.

Restando las ecuaciones 1.13 y 1.14 se tiene:

$$f(x_i + \Delta x) - f(x_i - \Delta x) = 2 \Delta x \left(\frac{df}{dx} \Big|_{x_i} \right) + \frac{\Delta x^3}{3!} \left[\left(\frac{d^3 f}{dx^3} \Big|_{\varepsilon_p} \right) + \left(\frac{d^3 f}{dx^3} \Big|_{\varepsilon_r} \right) \right]$$

Despejando se llega a la Ecuación 1.15.

$$\left(\frac{df}{dx} \Big|_{x_i} \right) = \frac{f(x_i + \Delta x) - f(x_i - \Delta x)}{2 \Delta x} + O_c(\Delta x^2)$$

Ecuación 1.15. Aproximación para la primera derivada.

Donde:

O_c : Es el error de truncamiento.

Las Ecuaciones 1.16 hasta la 1.18 presentan los errores de truncamiento o de discretización. [7]

$$O_c(\Delta x^2) = \frac{\Delta x^2}{3!} \left[\left(\frac{d^3 f}{dx^3} \Big|_{\varepsilon_p} \right) + \left(\frac{d^3 f}{dx^3} \Big|_{\varepsilon_r} \right) \right]$$

Ecuación 1.16. Error de truncamiento para la diferencia centralizada.

$$O_p(\Delta x) = -\frac{\Delta x^2}{2!} \left(\frac{d^2 f}{dx^2} \Big|_{\varepsilon_p} \right)$$

Ecuación 1.17. Error de truncamiento para la diferencia progresiva.

$$O_r(\Delta x) = \frac{\Delta x^2}{2!} \left(\frac{d^2 f}{dx^2} \Big|_{\varepsilon_r} \right)$$

Ecuación 1.18. Error de truncamiento para la diferencia regresiva.

El error por truncamiento es directamente proporcional a la distancia entre los puntos en análisis. Esto se puede evidenciar en las Ecuaciones 1.16, 1.17 y 1.18.

Comparando el error local de truncamiento de la diferencia centralizada con los errores de truncamiento de las diferencias progresivas y regresivas [7], se tiene que:

$$\lim_{\Delta x \rightarrow 0} O_c(\Delta x^2) < \lim_{\Delta x \rightarrow 0} O_p(\Delta x)$$

Donde es común encontrar la derivada (diferencias centradas) expresada como la Ecuación 1.19. [7]

$$\left(\frac{df}{dx} \Big|_{x_i} \right) = \frac{f(x_i + \Delta x) - f(x_i - \Delta x)}{2 \Delta x}$$

Ecuación 1.19. Derivada en función de puntos vecinos utilizando diferencias centradas.

1.3.2.2 Proximidad de la segunda derivada mediante diferencias centradas

Para $k = 4$ en las series de Taylor y escribiendo $f(x)$ en $x = x_i + \Delta x$ (Expansión de Taylor progresiva, ε_p) y $x = x_i - \Delta x$ (Expansión de Taylor regresiva, ε_r) [7], se tienen las Ecuaciones 1.20 y 1.21.

$$f(x_i + \Delta x) = f(x_i) + \Delta x \left(\frac{df}{dx} \Big|_{x_i} \right) + \frac{\Delta x^2}{2!} \left(\frac{d^2 f}{dx^2} \Big|_{x_i} \right) + \frac{\Delta x^3}{3!} \left(\frac{d^3 f}{dx^3} \Big|_{x_i} \right) + \frac{\Delta x^4}{4!} \left(\frac{d^4 f}{dx^4} \Big|_{\varepsilon_p} \right)$$

Ecuación 1.20. Expansiones de Taylor progresivas.

$$f(x_i - \Delta x) = f(x_i) - \Delta x \left(\frac{df}{dx} \Big|_{x_i} \right) + \frac{\Delta x^2}{2!} \left(\frac{d^2 f}{dx^2} \Big|_{x_i} \right) - \frac{\Delta x^3}{3!} \left(\frac{d^3 f}{dx^3} \Big|_{x_i} \right) + \frac{\Delta x^4}{4!} \left(\frac{d^4 f}{dx^4} \Big|_{\varepsilon_r} \right)$$

Ecuación 1.21. Expansiones de Taylor regresivas.

Sumando las ecuaciones 1.20 y 1.21.

$$f(x_i + \Delta x) + f(x_i - \Delta x) = 2 \left[f(x_i) + \frac{\Delta x^2}{2!} \left(\frac{d^2 f}{dx^2} \Big|_{x_i} \right) + \frac{\Delta x^4}{4!} \left(\frac{d^4 f}{dx^4} \Big|_{\varepsilon_p} \right) \right]$$

Despejando la segunda derivada se obtiene:

$$\left(\frac{d^2 f}{dx^2} \Big|_{x_i} \right) = \frac{f(x_i + \Delta x) + f(x_i - \Delta x) - 2 f(x_i)}{\Delta x^2} - \frac{\Delta x^2}{12} \left(\frac{d^4 f}{dx^4} \Big|_{\varepsilon_p} \right)$$

La aproximación de la segunda derivada usando diferencias centradas con un error de truncamiento $O_c(\Delta x^2)$ [7], se presenta en la Ecuación 1.22.

$$\left(\frac{d^2 f}{dx^2} \Big|_{x_i} \right) = \frac{f(x_i + \Delta x) + f(x_i - \Delta x) - 2 f(x_i)}{\Delta x^2}$$

Ecuación 1.22. Aproximación de la segunda derivada usando diferencias centradas.

1.3.2.3 Derivadas en dos dimensiones

Las primeras y segundas derivadas para dos dimensiones se construyen de forma analítica. Estas se obtienen usando el teorema de Taylor para funciones de dos variables x, y (Ver Ecuaciones 1.23 y 1.24)

$$f(x_i + \Delta x, y_j) = f(x_i, y_j) + \Delta x \frac{\partial f(x_i, y_j)}{\partial x} + \frac{\Delta x}{2} \frac{\partial^2 f(x_i + \theta_1 \Delta x, y_j)}{\partial x^2}$$

Ecuación 1.23. Expansión de Taylor para una función de dos variables (Δx).

$$f(x_i, y_j + \Delta y) = f(x_i, y_j) + \Delta y \frac{\partial f(x_i, y_j)}{\partial y} + \frac{\Delta y}{2} \frac{\partial^2 f(x_i, y_j + \theta_2 \Delta y)}{\partial y^2}$$

Ecuación 1.24. Expansión de Taylor para una función de dos variables (Δy).

Las primeras derivadas de una función de dos variables expresadas en aproximaciones de diferencias centradas, se presentan en las Ecuaciones 1.25 (derivada con respecto a x) y 1.26 (derivada con respecto a y).

$$\frac{\partial f(x_i, y_j)}{\partial x} \cong \frac{f(x_i + \Delta x, y_j) - f(x_i - \Delta x, y_j)}{2 \Delta x}$$

Ecuación 1.25. Aproximación en diferencias centradas para la primera derivada de x.

$$\frac{\partial f(x_i, y_j)}{\partial y} \cong \frac{f(x_i, y_j + \Delta y) - f(x_i, y_j - \Delta y)}{2 \Delta y}$$

Ecuación 1.26. Aproximación en diferencias centradas para la primera derivada de y.

Las segundas derivadas de una función de dos variables expresada en aproximaciones de diferencias centradas se presentan en las Ecuaciones 1.27 (derivada con respecto a x) y 1.28 (derivada con respecto a y).

$$\frac{\partial^2 f(x_i, y_j)}{\partial x^2} \cong \frac{f(x_i + \Delta x, y_j) + f(x_i - \Delta x, y_j) - 2 f(x_i, y_j)}{\Delta x^2}$$

Ecuación 1.27. Aproximación en diferencias centradas para la segunda derivada de x.

$$\frac{\partial^2 f(x_i, y_j)}{\partial y^2} \cong \frac{f(x_i, y_j + \Delta y) + f(x_i, y_j - \Delta y) - 2 f(x_i, y_j)}{\Delta y^2}$$

Ecuación 1.28. Aproximación en diferencias centradas para la segunda derivada de y.

1.3.2.4 Derivadas en tres dimensiones

Las primeras y segundas derivadas para tres dimensiones se construyen de forma analítica. Estas se obtienen usando el teorema de Taylor para funciones de tres variables x, y, z (Ver Ecuaciones 1.29, 1.30 y 1.31).

$$f(x_i + \Delta x, y_j, z_k) = f(x_i, y_j, z_k) + \Delta x \frac{\partial f(x_i, y_j, z_k)}{\partial x} + \frac{\Delta x}{2} \frac{\partial^2 f(x_i + \theta_1 \Delta x, y_j, z_k)}{\partial x^2}$$

Ecuación 1.29. Expansión de Taylor para una función de tres variables (Δx).

$$f(x_i, y_j + \Delta y, z_k) = f(x_i, y_j, z_k) + \Delta y \frac{\partial f(x_i, y_j, z_k)}{\partial y} + \frac{\Delta y}{2} \frac{\partial^2 f(x_i, y_j + \theta_2 \Delta y, z_k)}{\partial y^2}$$

Ecuación 1.30. Expansión de Taylor para una función de tres variables (Δy).

$$f(x_i, y_j, z_k + \Delta z) = f(x_i, y_j, z_k) + \Delta z \frac{\partial f(x_i, y_j, z_k)}{\partial z} + \frac{\Delta z}{2} \frac{\partial^2 f(x_i, y_j, z_k + \theta_3 \Delta z)}{\partial z^2}$$

Ecuación 1.31. Expansión de Taylor para una función de tres variables (Δz).

Las primeras derivadas de una función de tres variables expresadas en aproximaciones de diferencias centradas se presentan en las Ecuaciones 1.32 (derivada con respecto a x), 1.33 (derivada con respecto a y) y 1.34 (derivada con respecto a z).

$$\frac{\partial f(x_i, y_j, z_k)}{\partial x} \cong \frac{f(x_i + \Delta x, y_j, z_k) - f(x_i - \Delta x, y_j, z_k)}{2 \Delta x}$$

Ecuación 1.32. Aproximación en diferencias centradas para la primera derivada de x.

$$\frac{\partial f(x_i, y_j, z_k)}{\partial y} \cong \frac{f(x_i, y_j + \Delta y, z_k) - f(x_i, y_j - \Delta y, z_k)}{2 \Delta y}$$

Ecuación 1.33. Aproximación en diferencias centradas para la primera derivada de y.

$$\frac{\partial f(x_i, y_j, z_k)}{\partial z} \cong \frac{f(x_i, y_j, z_k + \Delta z) - f(x_i, y_j, z_k - \Delta z)}{2 \Delta z}$$

Ecuación 1.34. Aproximación en diferencias centradas para la primera derivada de z.

Las segundas derivadas de una función de tres variables expresadas en aproximaciones de diferencias centradas, se presentan en las Ecuaciones 1.35 (derivada con respecto a x), 1.36 (derivada con respecto a y) y 1.37 (derivada con respecto a z).

$$\frac{\partial^2 f(x_i, y_j, z_k)}{\partial x^2} \cong \frac{f(x_i + \Delta x, y_j, z_k) + f(x_i - \Delta x, y_j, z_k) - 2 f(x_i, y_j, z_k)}{\Delta x^2}$$

Ecuación 1.35. Aproximación en diferencias centradas para la segunda derivada de x.

$$\frac{\partial^2 f(x_i, y_j, z_k)}{\partial y^2} \cong \frac{f(x_i, y_j + \Delta y, z_k) + f(x_i, y_j - \Delta y, z_k) - 2 f(x_i, y_j, z_k)}{\Delta y^2}$$

Ecuación 1.36. Aproximación en diferencias centradas para la segunda derivada de y.

$$\frac{\partial^2 f(x_i, y_j, z_k)}{\partial z^2} \cong \frac{f(x_i, y_j, z_k + \Delta z) + f(x_i, y_j, z_k - \Delta z) - 2f(x_i, y_j, z_k)}{\Delta z^2}$$

Ecuación 1.37. Aproximación en diferencias centradas para la segunda derivada de z.

1.3.3 Limitaciones en la solución analítica para resolver la ecuación de Laplace

La ecuación de Laplace soluciona problemas de campo eléctrico desarrollando y utilizando herramientas matemáticas apropiadas. Esta ecuación permite encontrar campos de potencial dentro de regiones cerradas por potenciales conocidos.

A continuación se presenta la Ecuación de Laplace tridimensional para medios homogéneos (Ecuación 1.38) y para medios no homogéneos (Ecuación 1.39):

$$\nabla^2 V = \frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} + \frac{\partial^2 V}{\partial z^2} = 0$$

Ecuación 1.38. Ecuación de Laplace para medios homogéneos.

$$\varepsilon \left(\frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} + \frac{\partial^2 V}{\partial z^2} \right) + \frac{\partial \varepsilon}{\partial x} \frac{\partial V}{\partial x} + \frac{\partial \varepsilon}{\partial y} \frac{\partial V}{\partial y} + \frac{\partial \varepsilon}{\partial z} \frac{\partial V}{\partial z} = 0$$

Ecuación 1.39. Ecuación de Laplace para medios no homogéneos.

La ecuación de Laplace debe solucionarse sujeta a ciertas condiciones de frontera. Es por esto que todo problema físico debe contener al menos una frontera conductora con valores asignados de potencial.

El teorema de la unicidad asegura que si una respuesta satisface la ecuación de Laplace sujeta a ciertas condiciones de frontera, entonces ésta es la única respuesta posible.

Existen varios métodos analíticos para solucionar la ecuación de Laplace. Por ejemplo: El método de la integración directa (se aplica solo a problemas en “una dimensión”) y el método de la solución producto (El campo de potencial varía con más de una de las tres coordenadas) [6].

La solución analítica de la ecuación de Laplace conlleva ciertas limitaciones y complicaciones, entre las cuales se tienen:

- La solución analítica de la ecuación de Laplace para resolver un problema depende del sistema de coordenadas (rectangular, cilíndrica, esférica) el cual se elige dependiendo de la geometría del problema. Ahora bien, si se trata de un problema complejo sería difícil escoger un sistema de coordenadas, más aun solucionarlo.
- El método de la integración directa se aplica solo a problemas en “una dimensión”, es decir, a aquellos en los cuales el campo de potencial (V) es una función de sólo una de las tres coordenadas [6].
- La solución producto de la ecuación de Laplace conlleva un análisis matemático profundo para el sistema de coordenadas rectangular. Este método general es aplicable a los demás sistemas de coordenadas. Sin embargo para los sistemas de coordenadas cilíndricas y esféricas los campos de potencial quedan expresados en términos de funciones matemáticas más avanzadas, por ejemplo desarrollos en polinomios de Legendre o las funciones de Bessel [6].
- En la práctica, la mayor parte de los problemas que involucran la solución de un campo de potencial no favorecen una solución analítica. Debido a que las regiones de interés pueden tener formas irregulares o estar configuradas de tal forma que las fronteras no pueden representarse mediante superficies de coordenadas constantes dentro de un único sistema de coordenadas [6].

Debido a estas complicaciones y limitaciones se desarrolla la solución de la ecuación de Laplace por medio del método matemático de las diferencias finitas.

2. MÉTODO DE LAS DIFERENCIAS FINITAS GAUSS – SEIDEL Y SU IMPLEMENTACIÓN EN PYTHON 3.5

El método de las Diferencias finitas Gauss – Seidel soluciona de manera aproximada problemas de ecuaciones diferenciales y ecuaciones en derivadas parciales [4].

Este capítulo presenta el método de las diferencias finitas y su implementación en Python 3.5. El cual utilizará un parámetro de relajación para acelerar la convergencia.

2.1 Método de las diferencias finitas

El método de las diferencias finitas tiene como objetivo aproximar el valor de las derivadas parciales de una función por medio de términos algebraicos. Además es un método iterativo, por lo cual el sistema de ecuaciones algebraicas se resuelve pudiendo llegar a un número largo de operaciones aritméticas [8].

Los términos algebraicos se obtienen mediante las expansiones de Taylor que contienen valores de la variable dependiente en puntos seleccionados.

Los valores en los puntos seleccionados se obtienen mediante sucesiones, es decir los puntos pasan de ser incógnitas en un proceso aritmético a ser datos en el siguiente. De esta manera los valores de los puntos se utilizan en la iteración actual para obtener otros valores en la misma iteración.

La ecuación diferencial parcial (EDP) presentada en 1.10 caracteriza el problema a solucionar. Esta EDP se resolverá mediante el método de las diferencias finitas tal como se muestra a continuación.

2.1.1 Aproximación de la derivada parcial en términos algebraicos de voltaje

Los términos algebraicos presentados en la Tabla 2.1 se obtienen mediante las series de Taylor (ecuación 1.19 para la primera derivada y ecuación 1.22 para la segunda derivada), éstos caracterizan la ecuación diferencial parcial (ecuación 1.10) utilizando la Figura 2.1.

Tabla 2.1. Términos de la ecuación diferencial parcial (ecuación 1.10) utilizando la Figura 2.1 [Elaboración propia].

Términos		
De permitividad	De voltaje	
Primera derivada	Primera derivada	Segunda derivada
$\frac{\partial \varepsilon}{\partial x} = \frac{\varepsilon_1 - \varepsilon_2}{2h}$	$\frac{\partial V}{\partial x} = \frac{V_1 - V_2}{2h}$	$\frac{\partial^2 V}{\partial x^2} = \frac{V_1 + V_2 - 2V_0}{h^2}$
$\frac{\partial \varepsilon}{\partial y} = \frac{\varepsilon_3 - \varepsilon_4}{2h}$	$\frac{\partial V}{\partial y} = \frac{V_3 - V_4}{2h}$	$\frac{\partial^2 V}{\partial y^2} = \frac{V_3 + V_4 - 2V_0}{h^2}$
$\frac{\partial \varepsilon}{\partial z} = \frac{\varepsilon_5 - \varepsilon_6}{2h}$	$\frac{\partial V}{\partial z} = \frac{V_5 - V_6}{2h}$	$\frac{\partial^2 V}{\partial z^2} = \frac{V_5 + V_6 - 2V_0}{h^2}$

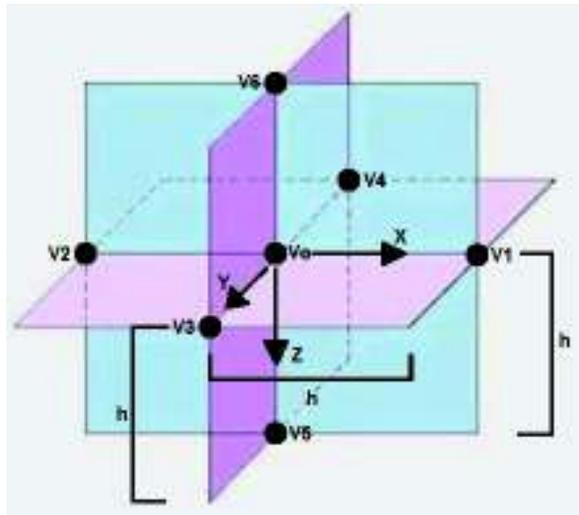


Figura 2.1. Parte de la región que contiene un campo de voltaje tridimensional dividido en cubos de longitud h por lado [Elaboración propia].

Donde:

V_0 : Es el voltaje en el punto analizado

$V_1, V_2, \varepsilon_1, \varepsilon_2$: Son los voltajes y permitividades vecinas en el eje X del punto analizado.

$V_3, V_4, \varepsilon_3, \varepsilon_4$: Son los voltajes y permitividades vecinas en el eje Y del punto analizado.

$V_5, V_6, \varepsilon_5, \varepsilon_6$: Son los voltajes y permitividades vecinas en el eje Z del punto analizado.

h : Es la longitud entre el punto vecino y el punto analizado.

Entonces, la Ecuación 1.10 queda definida en términos de voltaje y permitividades vecinas como:

$$\varepsilon_0 \left(\frac{V_1 + V_2 + V_3 + V_4 + V_5 + V_6 - 6V_0}{h^2} \right) + \frac{(\varepsilon_1 - \varepsilon_2)(V_1 - V_2) + (\varepsilon_3 - \varepsilon_4)(V_3 - V_4) + (\varepsilon_5 - \varepsilon_6)(V_5 - V_6)}{4h^2} = 0$$

La Ecuación 2.1 se obtiene al despejar V_0 . Ésta expresa el voltaje en un punto en función de voltajes y permitividades vecinas para un entorno tridimensional.

$$V_0 = \frac{1}{6} \left(V_1 + V_2 + V_3 + V_4 + V_5 + V_6 + \frac{(\varepsilon_1 - \varepsilon_2)(V_1 - V_2) + (\varepsilon_3 - \varepsilon_4)(V_3 - V_4) + (\varepsilon_5 - \varepsilon_6)(V_5 - V_6)}{4\varepsilon_0} \right)$$

Ecuación 2.1. Voltaje en un punto en función de voltajes y permitividades vecinas para tres dimensiones

De forma similar se obtiene la expresión para dos dimensiones utilizando los términos de la Tabla 2.1 en la Ecuación 1.11. Ver ecuación 2.2.

$$V_0 = \frac{1}{4} \left(V_1 + V_2 + V_3 + V_4 + \frac{(\varepsilon_1 - \varepsilon_2)(V_1 - V_2) + (\varepsilon_3 - \varepsilon_4)(V_3 - V_4)}{4\varepsilon_0} \right)$$

Ecuación 2.2. Voltaje en un punto en función de voltajes y permitividades vecinas para dos dimensiones

La Ecuación 2.3 es la representación genérica de las Ecuaciones 2.1 y 2.2. Es decir, caracteriza un entorno bidimensional y tridimensional para medios no homogéneos.

$$V_0^{(Gauss-Seidel)} = \frac{1}{cte} \left(V_1 + V_2 + V_3 + V_4 + \frac{(\varepsilon_1 - \varepsilon_2)(V_1 - V_2) + (\varepsilon_3 - \varepsilon_4)(V_3 - V_4) + ter_adi}{4\varepsilon_0} \right)$$

Ecuación 2.3. Voltaje en un punto en función de voltajes y permitividades vecinas

Donde:

$V_0^{(Gauss-Seidel)}$: valor de voltaje actual en un punto, utilizando solo el método de Gauss – Seidel.

cte : es el número de puntos vecinos al punto analizado. (cte = 4, en un caso bidimensional (ecuación 2.2) y cte = 6, en un caso tridimensional (ecuación 2.1))

Vn : voltajes vecinos al punto analizado, Vo. (Ver figura 2.1)

ϵ_n : permitividades vecinas al punto analizado, ϵ_0 . (Ver figura 2.1)

ter_adi : Son los términos que diferencian la ecuación 2.2 de la 2.1

Para un caso bidimensional:

$$cte = 4 \quad ter_adi = 0$$

Para un caso tridimensional:

$$cte = 6 \quad ter_adi = 4 \epsilon_0 (V_5 + V_6) + (\epsilon_5 - \epsilon_6)(V_5 - V_6)$$

Se debe crear una rejilla segmentada por líneas imaginarias separadas una longitud infinitesimal h (ver Figura 2.1). Ésta forma un conjunto de puntos ubicados ordenadamente a los lados, delante, atrás, arriba y abajo del otro para tres dimensiones.

En estos puntos se aplica el método de las diferencias finitas a la Ecuación 2.3 ocasionando iteraciones sucesivas.

2.1.2 Método de Gauss - Seidel

El método de Gauss – Seidel calcula valores de incógnitas y los reutiliza en la misma iteración [9]. Las iteraciones efectuadas en este método se repiten hasta que las incógnitas cumplan la tolerancia (comparación entre el valor anterior y el actual). El siguiente ejemplo se plantea para comprender de mejor manera lo enunciado anteriormente.

El ejemplo tiene dos incógnitas a y b. Cada incógnita es el promedio de los cuatro puntos vecinos. Además la tolerancia es de 0.15. La Figura 2.2 muestra el modelo de forma gráfica.

	1	1	
1	a	b	5
	1	1	

Figura 2.2. Ejemplo para el método Gauss – Seidel [Elaboración propia].

Inicialmente: (a = b = 0):

Iteración 1

Se obtienen los nuevos valores de "a" y "b":

$$a = \frac{1 + 1 + 1 + b}{4} = 0.75$$

El nuevo valor de "a" es 0.75

$$b = \frac{1 + 1 + 5 + a}{4} = 1.94$$

El nuevo valor de "b" es 1.94

Comparación de la tolerancia:

$$|a_{ant} - a_{act}| = |0 - 0.75| = 0.75$$

$0.75 \geq 0.15$ entonces la incógnita "a" necesita una nueva iteración

$$|b_{ant} - b_{act}| = |0 - 1.94| = 1.94$$

$1.94 \geq 0.15$ entonces la incógnita "b" necesita una nueva iteración

Iteración 2

Ahora: a = 0.75 y b = 1.94 (estos valores se obtuvieron en la iteración 1)

$$a = \frac{1 + 1 + 1 + 1.94}{4} = 1.24$$

El nuevo valor de "a" es 1.24

$$b = \frac{1 + 1 + 5 + a}{4} = 2.06$$

El nuevo valor de "b" es 2.06

Comparación de la tolerancia:

$$|a_{ant} - a_{act}| = |0.75 - 1.24| = 0.49$$

$0.49 \geq 0.15$ entonces la incógnita "a" necesita una nueva iteración

$$|b_{ant} - b_{act}| = |1.94 - 2.06| = 0.12$$

$0.12 < 0.15$ entonces la incógnita "b" no necesita una nueva iteración

Iteración 3

Ahora: $a = 1.24$ y $b = 2.06$ (estos valores se obtuvieron en la iteración 2)

$$a = \frac{1 + 1 + 1 + 2.06}{4} = 1.27$$

El nuevo valor de "a" es 1.27

$$b = \frac{1 + 1 + 5 + a}{4} = 2.07$$

El nuevo valor de "b" es 2.07

Comparación de la tolerancia:

$$|a_{ant} - a_{act}| = |1.24 - 1.27| = 0.03$$

$0.03 < 0.15$ entonces la incógnita "a" no necesita una nueva iteración

$$|b_{ant} - b_{act}| = |2.06 - 2.07| = 0.01$$

$0.01 < 0.15$ entonces la incógnita "b" no necesita una nueva iteración

De esta manera la solución para las incógnitas es la obtenida en la última iteración donde todos los elementos cumplen la tolerancia.

$$a = 1.27$$

$$b = 2.07$$

2.1.3 Método de relajación

El método de relajación mejora la convergencia en cada iteración potencializando la relación de recurrencia mediante un factor de relajación [10]. Este método considera un promedio ponderado de las iteraciones anteriores y actuales del método de Gauss – Seidel. Ver Ecuación 2.4.

La ecuación modificada teniendo en cuenta el promedio ponderado es:

$$x_i^{(nuevo)} = rel \ x_i^{(gauss-seidel-actual)} + (1 - rel) \ x_i^{(anterior)}$$

Ecuación 2.4. Método de relajación [11].

Donde:

$X_i^{(Nuevo)}$: valor de la incógnita obtenido en la iteración actual utilizando el parámetro de relajación

$X_i^{(Gauss-Seidel-Actual)}$: valor de la incógnita obtenido en la iteración actual utilizando solo el método de Gauss – Seidel.

$X_i^{(Anterior)}$: valor de la incógnita que se obtuvo en la iteración anterior

rel : Es el parámetro de relajación o factor de ponderación comprendido entre 0 y 2. (Ver Tabla 2.2).

Tabla 2.2. Valores que puede tomar el parámetro de relajación [12].

Rel	Método	Observación
$0 < rel < 1$	Sub-relajación	Se emplea para un sistema que no converge con Gauss – Seidel
$rel = 1$	Gauss-Seidel	Se emplea para un sistema que converge
$1 < rel < 2$	Sobre-relajación	Acelera la convergencia del sistema que si converge con Gauss – Seidel
$rel > 2$		El método diverge

De esta manera al utilizar el método de relajación (Ecuación 2.4) para el caso de estudio (Ecuación 2.3) se tiene la Ecuación 2.5.

$$V_0^{(Nuevo)} = V_0^{(Gauss-Seidel-Actual)} \times rel + (1 - rel) \times V_0^{(Anterior)}$$

Ecuación 2.5. Forma general del método de relajación para las funciones de voltaje.

Donde:

$V_0^{(Nuevo)}$: valor de voltaje obtenido en la iteración actual utilizando el parámetro de relajación

$V_0^{(Gauss-Seidel-Actual)}$: valor de voltaje obtenido en la iteración actual utilizando solo el método de Gauss – Seidel. En otras palabras es la Ecuación 2.3

$V_0^{(Anterior)}$: valor de voltaje que se obtuvo en la iteración anterior

rel : parámetro de relajación

Cuando la Ecuación 2.5 es evaluada con $rel = 1$, se obtiene el mismo valor de voltaje que el obtenido por el método de Gauss – Seidel.

Al reemplazar la Ecuación 2.3 en la 2.5 se obtiene la 2.6 que representa el método de relajación utilizado para el caso de estudio.

$$V_0^{(nuevo)} = \frac{rel}{cte} \left(V_1 + V_2 + V_3 + V_4 + \frac{(\varepsilon_1 - \varepsilon_2)(V_1 - V_2) + (\varepsilon_3 - \varepsilon_4)(V_3 - V_4) + ter_adi}{4 \varepsilon_0} \right) + (1 - rel) V_0^{(anterior)}$$

Ecuación 2.6. Voltaje en función de voltajes y permitividades vecinas utilizando el método de relajación

Para el caso bidimensional:

$$cte = 4 \quad ter_adi = 0$$

Para el caso tridimensional:

$$cte = 6 \quad ter_adi = 4 \varepsilon_0 (V_5 + V_6) + (\varepsilon_5 - \varepsilon_6)(V_5 - V_6)$$

Las condiciones de frontera se definen a lo largo del borde de una región y sirven para especificar valores constantes. Por lo tanto son necesarias para utilizar el método de las diferencias finitas.

CEDIF sigue el siguiente proceso para obtener el comportamiento del campo electrostático:

1. Aplica el método de las diferencias finitas en cada punto de rejilla a la Ecuación 2.6, de esta manera se determinan superficies equipotenciales en el entorno.
2. Utiliza el gradiente del potencial para cada punto de rejilla y así se obtiene el comportamiento del campo eléctrico en estado estable.

2.2 Implementación del método de las diferencias finitas en Python 3.5

El código fuente desarrollado en Python 3.5 lee los datos ingresados por el usuario, de esta manera establece matrices de dos y tres dimensiones para modelar el problema. Este algoritmo aplica el método de las diferencias finitas para conocer la distribución de voltajes en el entorno. Después mediante el gradiente de potencial se obtiene la distribución del campo electrostático.

2.2.1 Condiciones de frontera

Las condiciones de frontera son establecidas por el usuario a través de la matriz de condiciones frontera y la matriz de voltajes que son ingresadas por el usuario. La matriz de condiciones frontera solo puede tener dos valores: uno o cero. En la Tabla 2.3 se muestra el uso de estos dos números.

Tabla 2.3. Números válidos para ingresar en la matriz de condiciones frontera [Elaboración propia].

Número	Uso
0	Asignar a un elemento de la matriz como condición de frontera (es decir nunca variará en las iteraciones)
1	Asignar a un elemento de la matriz como punto de análisis (es decir variará en cada iteración hasta cumplir la tolerancia)

El usuario ingresa estos números en la “Matriz de condiciones frontera”.

El código de programa importa los datos y valores estipulados por el usuario en las hojas “Datos iniciales”, “Matriz de voltajes”, “Matriz de condiciones frontera” y “Matriz de permitividades”. Este algoritmo por medio de la “Matriz de voltajes” y la “Matriz de condiciones frontera” establece condiciones de frontera. Este enunciado se especifica en el mapa conceptual de la Figura 2.3.

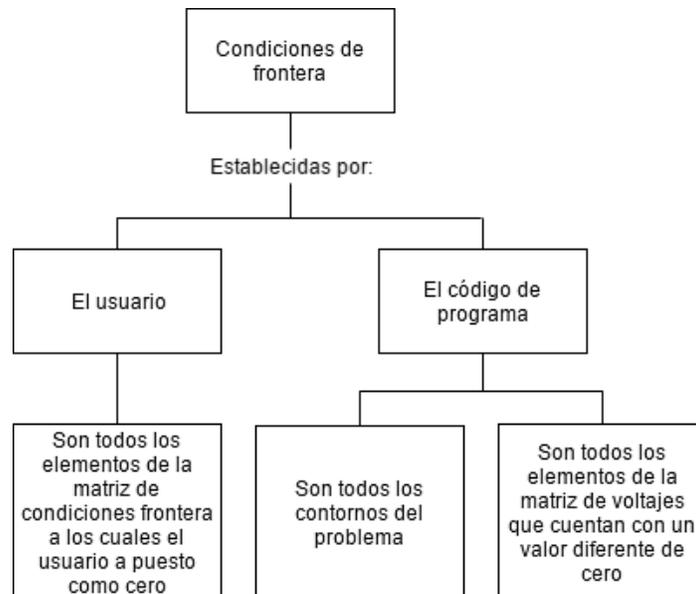


Figura 2.3. Mapa conceptual donde se muestran las formas de establecer condiciones de frontera [Elaboración propia].

Como ejemplo del desarrollo del programa se plantea el caso del campo en un entorno bidimensional con distribución de voltajes y valores de frontera, tal como se especifica en las Figuras 2.4, 2.5 y 2.6.

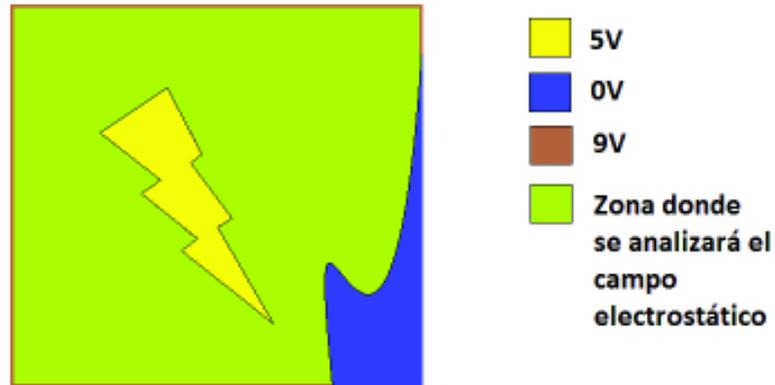


Figura 2.4. Problema planteado de dos dimensiones [Elaboración propia].

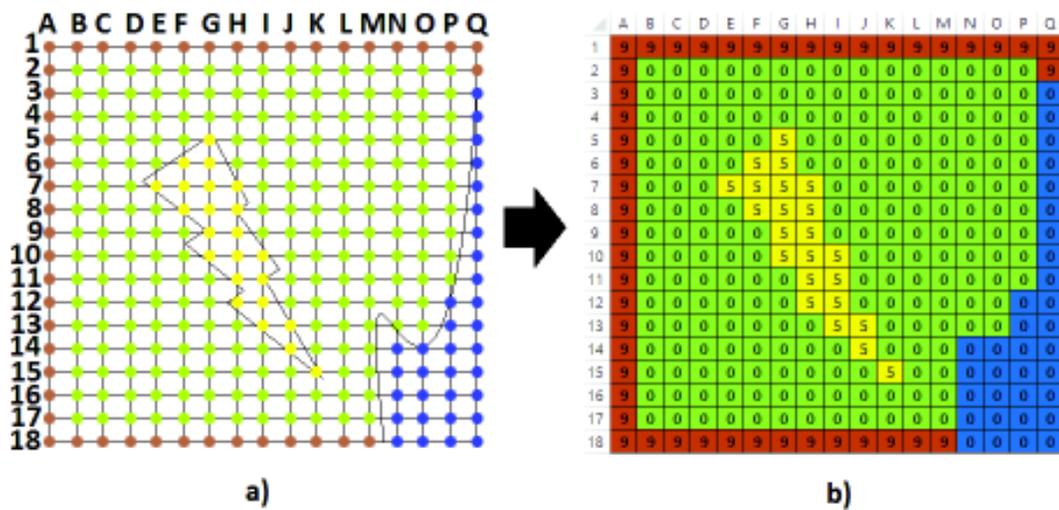


Figura 2.5. a) Problema planteado de forma gráfica ajustado con la longitud infinitesimal donde se muestran los niveles de voltaje [Elaboración propia]. **b)** Matriz de voltajes que se debe ingresar en Excel para describir el problema planteado [Impresión de pantalla de Excel].

El voltaje se va a calcular solamente en la región verde de las Figuras expuestas en 2.5.

En la Figura 2.5 b se observan voltajes nulos en la zona azul y verde. Por esta razón se debe diferenciar por medio de la “Matriz de condiciones frontera” sobre cual zona es condición de frontera (región azul) y cual es zona de análisis (región verde).

La “Matriz de condiciones frontera” es modelada por el usuario y por el código de programa tal como se muestra en la Figura 2.3. Además, el algoritmo une estas dos modelaciones en una sola tal como describe la Figura 2.7 literal b.

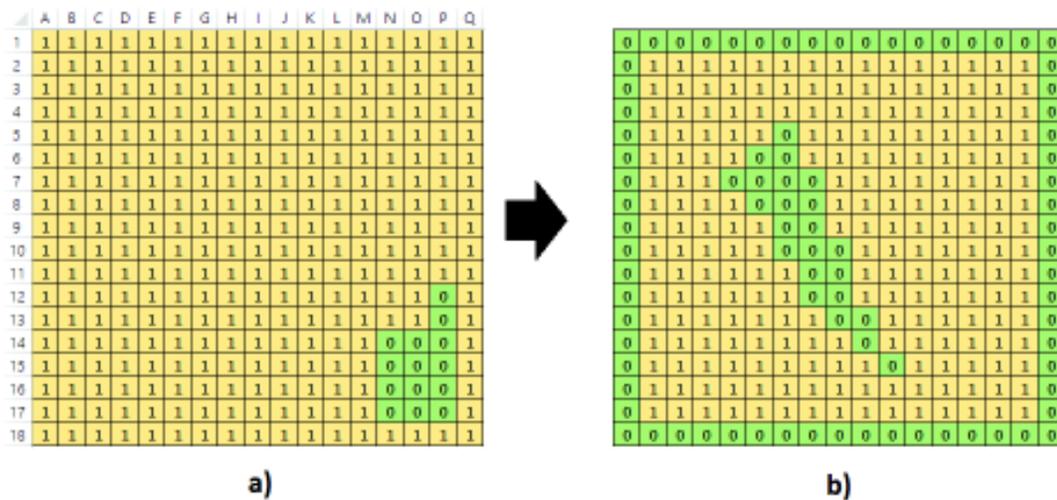


Figura 2.6. a) Matriz de condiciones frontera que debería ingresar el usuario (en Excel) [Impresión de pantalla de Excel]. **b)** Matriz de condiciones frontera que especifica el código de programa [Datos obtenidos de Python].

La matriz de condiciones frontera que el usuario debería ingresar se encuentra en la Figura 2.6 a. Esta matriz posee ceros ya que la zona azul de la Figura 2.4 se encuentra aterrizada y nunca cambiará en las iteraciones.

La matriz de condiciones frontera que establece el código de programa se observa en la Figura 2.6 b. Esta matriz se establece por medio de comandos lógicos y condicionales, los mismos que colocan ceros en los contornos del problema y en los puntos donde existe voltaje diferente de cero.

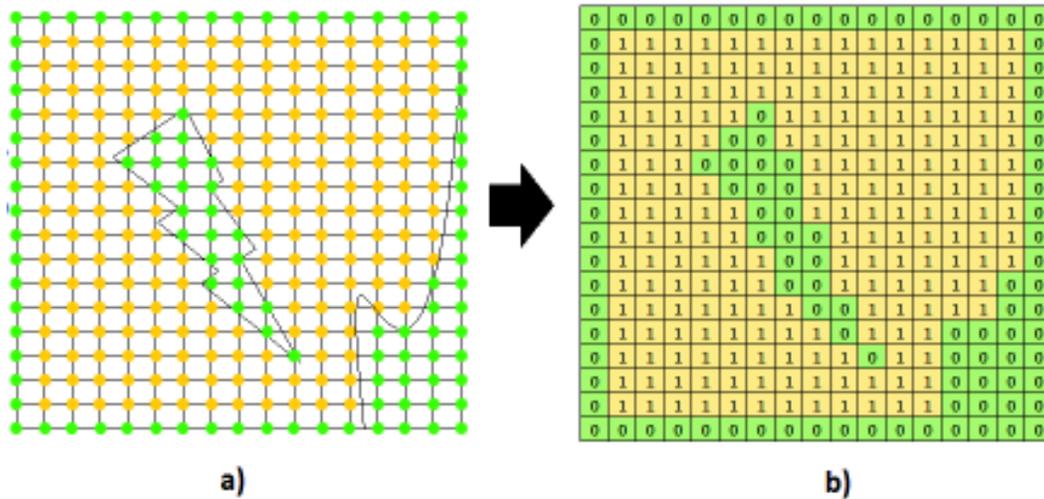


Figura 2.7. a) Entorno del problema (puntos amarillos zona de análisis, puntos verdes condiciones de frontera) [Elaboración propia]. **b)** Matriz de condiciones frontera establecida por el usuario y por el código de programa después de depurar el algoritmo [Datos obtenidos de Python].

La matriz general de condiciones frontera se muestra en la Figura 2.7. Esta se forma a partir de las matrices establecidas por el usuario (Figura 2.6 a) y por el código de programa (Figura 2.6 b). Además esta matriz se forma ya que el algoritmo adquiere y evalúa los datos ingresados por el usuario en la hoja de Excel.

El código de programa establece condiciones de frontera utilizando el siguiente algoritmo:

```
matrizC = np.ones((Y, X))
for vary in range(Y):
    for varx in range(X):
        if(vary == 0 or varx == 0 or vary == Y-1 or varx == X-1 or matrizV[vary,
                                                                    varx] != 0):
            matrizC[vary, varx] = 0
        else:
            matrizC[vary, varx] = hoja.cell(row = Y*varz+vary+1, column =
                                                                    varx+1).value
```

Donde:

- matrizC : Es la matriz de Condiciones frontera
- matrizV : Es la matriz de Voltajes

- X, Y : Son los puntos de rejilla en la dimensión X, Y respectivamente
- varx, vary: Son las variables utilizadas por el bucle for - in

De esta manera, la matriz de condiciones frontera asegura tener ceros en los límites del entorno y donde sea necesario mantener el voltaje constante.

2.2.2 Algoritmo del método de las diferencias finitas Gauss – Seidel con método de relajación.

El método iterativo Gauss – Seidel implementado en lenguaje Python utiliza el comando *while*. Este bucle se repite hasta que todos los elementos de la matriz de voltajes cumplan con el margen mínimo de error estipulado por la tolerancia. Adicionalmente en cada iteración el código de programa realiza la función que describe el método de las diferencias finitas Gauss – Seidel.

La constante *iter* aumentará su valor en pasos de uno cada vez que se utilice el bucle *while*, de esta manera se cuenta el número de iteraciones realizadas. También en cada iteración contabiliza el número de elementos de la matriz de voltajes que no cumplen con la tolerancia a través de la variable *no_tol*.

Por lo tanto, el bucle *while* se repetirá hasta que la variable *no_tol* sea cero.

Lo señalado en los párrafos anteriores se describe por medio del siguiente código:

```
iter = 0
no_tol = 1

while(no_tol != 0):
    variables = función_gauss_seidel(parámetros)
    iter = iter + 1
```

Donde:

iter : número de iteraciones.

no_tol : número de elementos de la matriz de voltajes que no cumplen con la tolerancia.

variables : valores que devuelve la función.

parámetros: datos que necesita la función para operar.

Función Gauss – Seidel con método de relajación

La Ecuación 2.6 se desarrolla en CEDIF, por esta razón es importante describir el algoritmo que aplica las diferencias finitas a esta ecuación.

Los parámetros, variables internas y retornos utilizados en la función Gauss – Seidel son:

Parámetros:

v : matriz de voltajes antes de la iteración Gauss – Seidel.

maco : matriz de condiciones frontera.

per : matriz de permitividades.

tol : tolerancia.

dim : número de dimensiones del problema.

X, Y : puntos de rejilla en la dimensión X, Y respectivamente.

iniZ : valor inicial del barrido en la dirección del eje Z.

finZ : último valor del barrido en la dirección del eje Z.

cte : número de elementos vecinos (dos dimensiones: cte = 4,
tres dimensiones: cte = 6)

rel : parámetro de relajación

Variables internas:

ter_adi: puntos vecinos en el eje Z que se adicionan en un caso tridimensional

v_ant : voltaje de un elemento antes de la iteración Gauss – Seidel

v_des : voltaje de un elemento después de la iteración Gauss – Seidel

Retornos:

v : matriz de voltajes después de la iteración Gauss – Seidel

no_tol : número de elementos de la matriz de voltajes que no cumplen con la tolerancia

La función se describe por medio del siguiente código:

- **Definición de la función**

```
def función_gauss_seidel_rel(v, maco, per, tol, dim, X, Y, iniZ, finZ, cte, rel):
```

- **Inicialización del contador**

```
no_tol = 0
```

- **Barrido de todos los elementos de la matriz 3D a excepción de los bordes**

```
for k in range(iniZ, finZ + 1):  
    for j in range(1, int(Y) - 1):  
        for i in range(1, int(X) - 1):
```

- **Condicional para detectar condiciones de frontera (es condición de frontera si el elemento maco[k, j, i] == 0)**

```
    if(maco[k, j, i] == 1):
```

- **Análisis debido al número de dimensiones**

```
        if(dim == 2):  
            ter_adi = 0
```

```
        elif(dim == 3):  
            ter_adi = ((v[k-1,j,i] + v[k+1,j,i])*4*per[k,j,i] + (v[k+1,j,i] -  
                v[k-1,j,i])*(per[k+1,j,i] - per[k-1,j,i]))
```

- **Voltaje de un elementos antes de la iteración Gauss - Seidel**

```
            v_ant = v[k, j, i]
```

- **Iteración Gauss – Seidel (con parámetro de relajación)**

```
            v[k,j,i] = ((rel / cte) * (v[k,j,i+1] + v[k,j,i-1] + v[k,j+1,i] + v[k,j-1,i] + ((per[k,j,i+1] -  
                per[k,j,i-1]) * (v[k,j,i+1] - v[k,j,i-1]) + (per[k,j+1,i] - per[k,j-1,i]) *  
                (v[k,j+1,i] - v[k,j-1,i]) + ter_adi) / (4*per[k,j,i])) + (1-rel)*v[k,j,i])
```

- **Voltaje en el mismo elemento después de la iteración Gauss - Seidel**

```
            v_des = v[k, j, i]
```

- **Análisis de tolerancia en cada elemento de la matriz**

```
            if(abs(v_des - v_ant) >= tol):  
                no_tol = no_tol + 1
```

- **Retorno de variables**

```
return v, no_tol
```

2.2.3 Campo Electrostático

La conducta del campo electrostático se observa al utilizar una función descrita en el código de programa. Esta función realiza el gradiente de potencial a cada punto del entorno para obtener el campo electrostático.

La función necesita de parámetros, variables internas y retornos que se describen a continuación:

Parámetros:

X, Y, Z: puntos de rejilla en la dimensión X, Y y Z respectivamente.

v : matriz de voltajes.

li : longitud infinitesimal.

tol : tolerancia.

lon : unidad de longitud de la longitud infinitesimal (mm, cm, dm, m).

Variables internas:

lmi : longitud en metros de la longitud infinitesimal.

num1i : voltaje del elemento a la izquierda del punto analizado. Ver Figura 2.8.

num2i : voltaje del elemento a la derecha del punto analizado. Ver Figura 2.8.

num1j : voltaje del elemento que se encuentra al frente del punto analizado. Ver Figura 2.8.

num2j : Voltaje del elemento que se encuentra atrás del punto analizado. Ver Figura 2.8.

num1k : Voltaje del elemento que se encuentra debajo del punto analizado.

num2k : Voltaje del elemento que se encuentra encima del punto analizado.

mayori: mayor voltaje entre los puntos vecinos en el eje X.

mayorj: mayor voltaje entre los puntos vecinos en el eje Y.

mayork: mayor voltaje entre los puntos vecinos en el eje Z.

sigi : dirección del campo en el eje X.

sigj : dirección del campo en el eje Y.

sigk : dirección del campo en el eje Z.

Retornos:

Ex : matriz que contiene las componentes X del campo electrostático en cada punto.

Ey : matriz que contiene las componentes Y del campo electrostático en cada punto.

Ez : matriz que contiene las componentes Z del campo electrostático en cada punto.

E : matriz que contiene la magnitud del campo electrostático en cada punto.

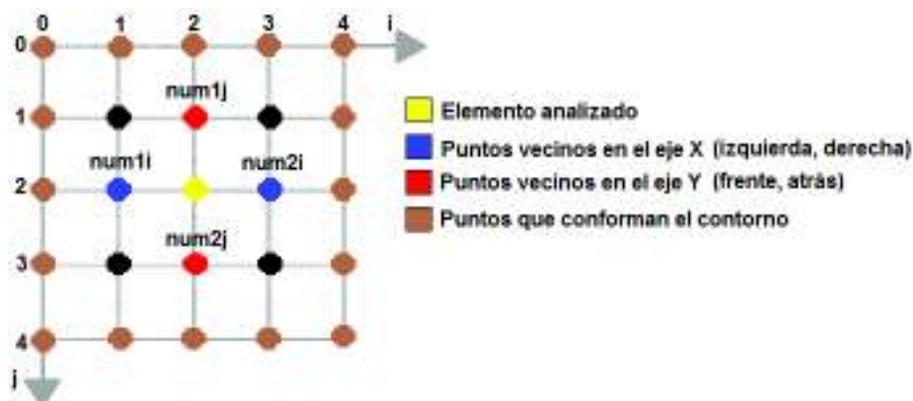


Figura 2.8. Ejemplo de una matriz de voltajes donde se observan los puntos vecinos al elemento en análisis [Elaboración propia].

La función campo obtiene las componentes X, Y, Z del campo electrostático para cada punto del problema. Estos valores se guardan en las matrices Ex, Ey, Ez para la posterior representación gráfica de las componentes x, y del campo en los planos z.

La función campo se describe por medio del siguiente algoritmo:

- **Definición de la función campo.**

```
def campo(Z, Y, X, v, li, tol, lon):
```

- **Asignación de la unidad de longitud.**

```
if(lon==str("mm")):
    lon=0.001
elif(lon==str("cm")):
    lon=0.01
elif(lon==str("dm")):
    lon=0.1
elif(lon==str("m")):
    lon=1
```

- **Longitud en metros de la longitud infinitesimal.**

```
lml = li * lon
```

- **Inicialización de las matrices de Campo Electrostático.**

```
Ex = np.zeros((Z, Y, X))
Ey = np.zeros((Z, Y, X))
Ez = np.zeros((Z, Y, X))
```

- **Barrido de todos los elementos de la matriz de voltaje que puedan contribuir a la componente X del campo electrostático.**

```
for k in range(int(Z)):
    for j in range(int(Y)):
        for i in range(1, int(X) - 1):
```

- **Designación de voltajes vecinos en el eje X.**

```
num1i = v[k, j, i - 1]
num2i = v[k, j, i + 1]
```

- **Ingresa si los voltajes vecinos en el eje X son iguales, es decir el punto analizado se encontraría en una región equipotencial.**

```
if(abs(num1i - num2i) < tol):
    mayori = v[k, j, i]
    sigi = 1
```

- **Ingresa si los voltajes vecinos en el eje X no son iguales.**

```
else:
```

- **Ingresa si el punto analizado se encuentra en el límite derecho de una región equipotencial (en la dirección de X)**

```
if(num1i == v[k, j, i]):
    mayori = num2i
    sigi = -1
```

- **Ingresa si el punto analizado se encuentra en el límite izquierdo de una región equipotencial (en la dirección de X)**

```
elif(num2i == v[k, j, i]):
    mayori = num1i
    sigi = 1
```

- **Ingresa si el punto analizado se encuentra en una región de voltaje variante (En la dirección de X)**

```
elif(num1i > num2i):
    mayori = num1i
    sigi = 1
else:
    mayori = num2i
    sigi = -1
```

- **Matriz que contiene las componentes del campo electrostático en el eje X.**

$$Ex[k, j, i] = (sigi * (mayori - v[k, j, i]) / li) / lmi$$

- **Barrido de todos los elementos de la matriz de voltaje que puedan contribuir a la componente Z del campo electrostático.**

```
for j in range(int(Y)):
    for i in range(int(X)):
        for k in range(1, int(Z) - 1):
```

- **Designación de voltajes vecinos en el eje Z.**

```
num1k = v[k-1, j, i]
num2k = v[k+1, j, i]
```

- **Ingresa si los voltajes vecinos en el eje Z son iguales, es decir el punto analizado se encontraría en una región equipotencial.**

```
if(abs(num1k - num2k) < tol):
    mayork = v[k, j, i]
    sigk = 1
```

- **Ingresa si los voltajes vecinos en el eje Z no son iguales.**

```
else:
```

- **Ingresa si el punto analizado se encuentra en el límite inferior de una región equipotencial (en la dirección de Z).**

```
if(num1k == v[k, j, i]):
    mayork = num2k
    sigk = -1
```

- **Ingresar si el punto analizado se encuentra en el límite superior de una región equipotencial (en la dirección de Z).**

```
elif(num2k == v[k, j, i]):
    mayork = num1k
    sigk = 1
```

- **Ingresar si el punto analizado se encuentra en una región de voltaje variante (En la dirección de Z).**

```
elif(num1k > num2k):
    mayork = num1k
    sigk = 1
else:
    mayork = num2k
    sigk = -1
```

- **Matriz que contiene las componentes del campo electrostático en el eje Z.**

$$Ez[k, j, i] = (sigk * (mayork - v[k, j, i]) / li) / lmi$$

- **Barrido de todos los elementos de la matriz de voltaje que puedan contribuir a la componente Y del campo electrostático.**

```
for i in range(int(X)):
    for k in range(int(Z)):
        for j in range(1, int(Y) - 1):
```

- **Designación de voltajes vecinos en el eje Y.**

```
num1j = v[k, j - 1, i]
num2j = v[k, j + 1, i]
```

- **Ingresar si los voltajes vecinos en el eje Y son iguales, es decir el punto analizado se encontraría en una región equipotencial.**

```
if(abs(num1j - num2j) < tol):
    mayorj = v[k, j, i]
    sigj = 1
```

- **Ingresar si los voltajes vecinos en el eje Y no son iguales.**

```
else:
```

- **Ingresar si el punto analizado se encuentra en el límite posterior de una región equipotencial (en la dirección de Y).**

```
if(num1j == v[k, j, i]):
    mayorj = num2j
    sigj = -1
```

- **Ingresar si el punto analizado se encuentra en el límite frontal de una región equipotencial (en la dirección de Y).**

```
elif(num2j == v[k, j, i]):
    mayorj = num1j
    sigj = 1
```

- **Ingresar si el punto analizado se encuentra en una región de voltaje variante (En la dirección de Y).**

```
elif(num1j > num2j):
    mayorj = num1j
    sigj = 1
else:
    mayorj = num2j
    sigj = -1
```

- **Matriz que contiene las componentes del campo electrostático en el eje Y.**

$$E_y[k, j, i] = (\text{sigj} * (\text{mayorj} - v[k, j, i]) / l_i) / |mli$$

- **Generación de la matriz campo electrostático.**

```
E[:, :, :] = np.sqrt((Ex[:, :, :])**2 + (Ey[:, :, :])**2 + (Ez[:, :, :])**2)
```

- **Devolución de las matrices que contienen las componentes del campo electrostático.**

```
return Ex, Ey, Ez, E
```

El llamado a esta función se realiza de la siguiente manera:

```
Ex, Ey, Ez, E = campo(Z, Y, X, v, li, tol, lon)
```

3. FUNCIONAMIENTO DE LA HERRAMIENTA DE SOFTWARE

La herramienta de software tiene el objetivo de continuar con el estudio del campo electrostático dando como resultado la distribución del mismo en sistemas complejos. Este proceso se lleva a cabo utilizando el método numérico de las diferencias finitas.

CEDIF está conformada por dos partes importantes para garantizar su funcionamiento. La primera es un documento Excel llamado *“Datos del campo electrostático”* de donde se obtienen los valores de voltaje en cada uno de los puntos de rejilla que describen el problema. La segunda es un código fuente desarrollado en lenguaje Python 3.5 que determina el comportamiento del campo electrostático en entornos bidimensionales o tridimensionales.

En el presente capítulo se detalla el funcionamiento de CEDIF, por medio de diagramas de flujo.

3.1 Interacción Usuario – Herramienta de software

Se exhibe en la Figura 3.1 un diagrama de flujo que describe de la manera simple la interacción entre el usuario y la nombrada herramienta.

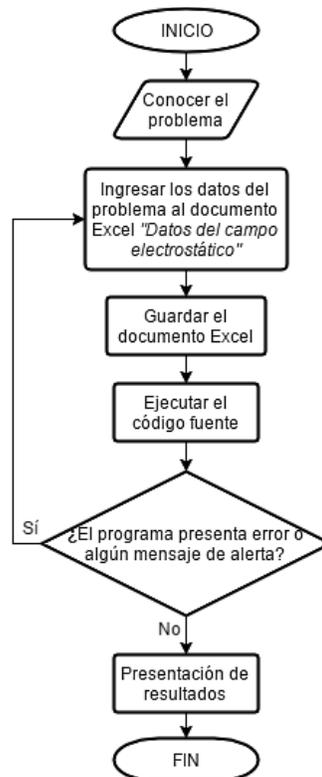


Figura 3.1. Diagrama de flujo que describe de forma simple la interacción usuario – CEDIF [Elaboración propia].

3.2 Funcionamiento del Complemento en Excel

CEDIF utiliza un documento en Excel llamado “Datos del campo electrostático”, en este documento se ingresan los datos del problema. Entre los datos a ingresar están: número de dimensiones, longitud de cada dimensión, longitud infinitesimal, tolerancia, parámetro de relajación, matriz de voltajes, matriz de condiciones frontera y matriz de permitividades.

En la Figura 3.2 se presenta un diagrama de flujo que describe el proceso de como ingresar datos del problema al nombrado documento.

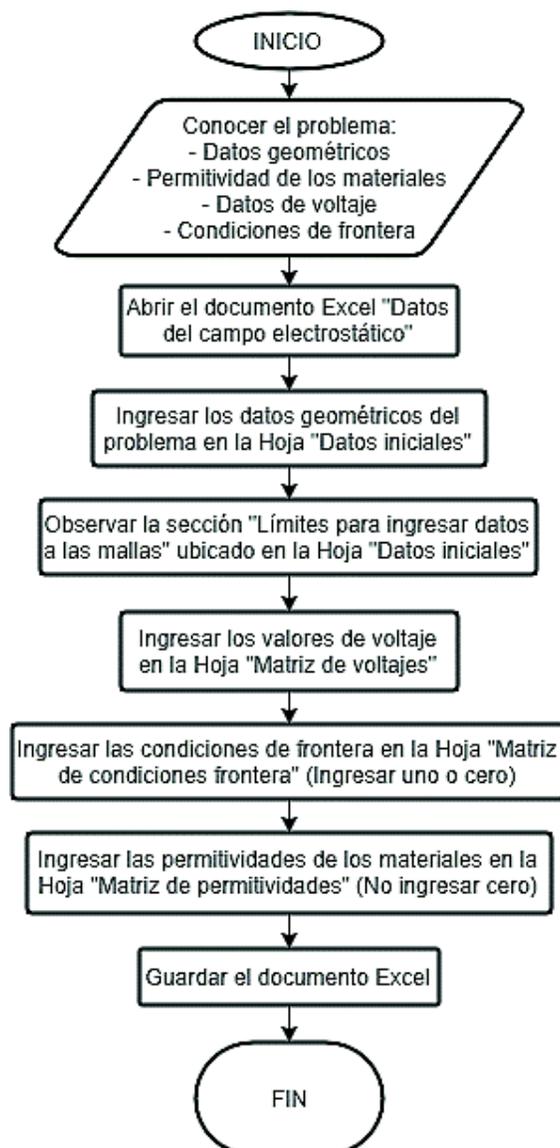


Figura 3.2. Funcionamiento e ingreso de datos al documento Excel “Datos del campo electrostático” [Elaboración propia].

3.3 Funcionamiento del código fuente

El algoritmo implementado adquiere, guarda, procesa, grafica y presenta resultados e información del problema que el usuario desea analizar. En todo este proceso se mide el tiempo de ejecución.

La Figura 3.3 muestra un diagrama de flujo que describe el funcionamiento del algoritmo.

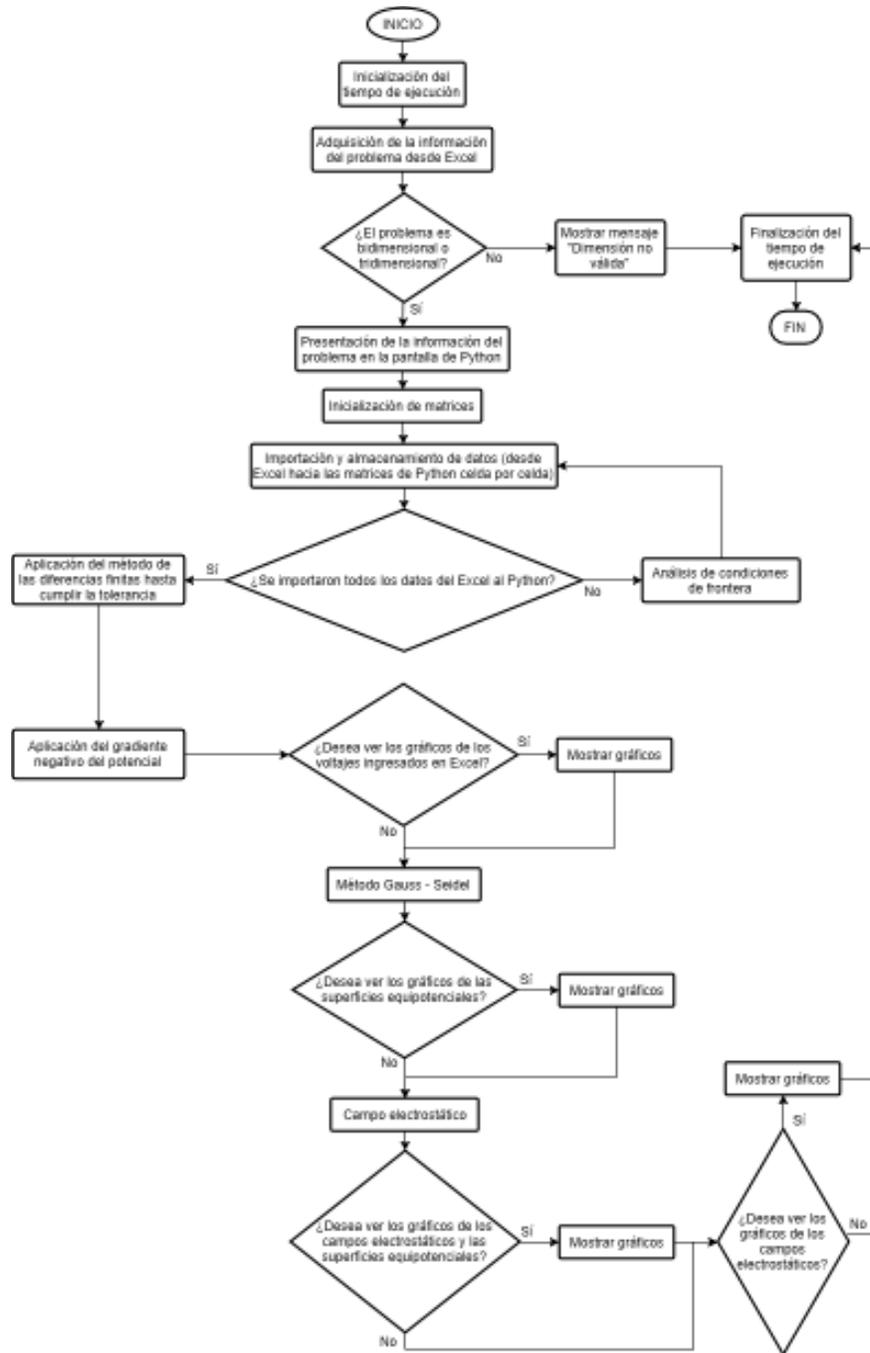


Figura 3.3. Resumen del algoritmo desarrollado en Python 3.5 [Elaboración propia].

4. EJEMPLOS DE APLICACIÓN Y ANÁLISIS DE RESULTADOS

En este capítulo se citan algunas dificultades al modelar regiones complejas. Además, se presentan los resultados obtenidos al utilizar CEDIF en varias aplicaciones.

4.1 Complicaciones de utilizar el método de las diferencias finitas en estructuras complejas

Detallar una estructura compleja para utilizar el método de las diferencias finitas es complicado porque esta metodología usa una rejilla caracterizada por la distancia entre sus rectas paralelas. Por lo tanto, se hace difícil concretar una figura curva, línea curva, círculos, circunferencias, esferas, esferas huecas, etc.

Para mayor comprensión se estudiará con un ejemplo la implementación de un círculo y una circunferencia.

4.1.1 Implementación de un círculo

Círculo es el “área o superficie plana contenida dentro de una circunferencia” [13].

La modelación de un círculo en una rejilla de 8x8 se puede observar en la Figura 4.1.

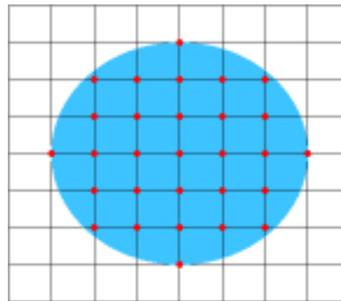


Figura 4.1. Círculo en una rejilla de 8x8 [Elaboración propia].

El método de las diferencias finitas utiliza una rejilla regular. El enrejado cuenta con rectas paralelas distanciadas a una longitud específica, por esta razón modelar curvas se vuelve complicado.

Las rejillas utilizadas para modelar figuras requieren de longitudes infinitesimales. Cuando estas longitudes son lo suficientemente pequeñas se conoce como refinamiento de la discretización y este refinamiento favorece al modelamiento de una figura.

De forma matemática la longitud infinitesimal es directamente proporcional al error por discretización e inversamente proporcional a la resolución de la imagen. Entonces, mientras menor sea la longitud infinitesimal menor será el error por discretización y mejor será la resolución de la imagen. De igual manera mientras la longitud infinitesimal sea más grande aumentará el error por discretización y menor será la calidad de la figura.

En cuanto a la resolución del círculo (Figura 4.1) se puede decir que:

- La resolución del círculo es menor cuando la distancia entre los puntos de rejilla es mayor.
- La resolución del círculo es mayor cuando la distancia entre los puntos de rejilla es menor.

4.1.2 Implementación de una circunferencia

Circunferencia es una “curva plana, cerrada, cuyos puntos son equidistantes de otro, el centro, situado en el mismo plano”. [13]

Una circunferencia se modela en una rejilla de 40x40. Ver Figura 4.2.

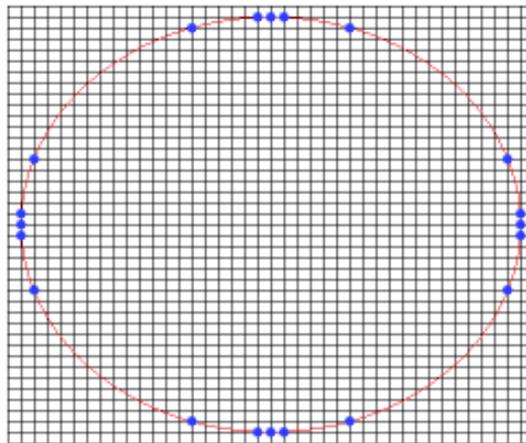


Figura 4.2. Circunferencia en una rejilla de 40x40 (sin considerar grosor en la línea que conforma la circunferencia) [Elaboración propia].

La circunferencia como el círculo son figuras geométricas curvas, por lo que se cumple lo ya estipulado en la sección anterior.

Una circunferencia modelada en una rejilla depende de la longitud infinitesimal. Tal es el caso de una circunferencia inscrita en una rejilla de 40x40 (Figura 4.2) donde se observa que solo 20 puntos cortan exactamente la intersección entre los puntos de rejilla y la curva que forma la circunferencia.

Una rejilla regular tan grande con longitudes infinitesimales tan pequeñas es la mejor forma de lograr la perfección de una circunferencia, pero controlaría muchísimos datos. Por lo que se recomienda considerar una solución que mantenga un número de datos prudentes y que modelen bien a las figuras.

El usuario deberá plantear un grosor específico a la línea curva que forma la circunferencia. Así, se obtendrá una aproximación a mencionada imagen tal como se muestra en la Figura 4.3.

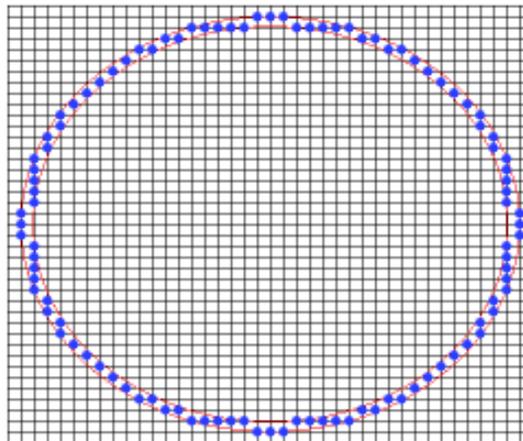


Figura 4.3. Circunferencia en una rejilla de 40x40 (considerando grosor en la línea que conforma la circunferencia) [Elaboración propia].

Las figuras curvas necesitan poseer un cierto grosor en las líneas que las conforman. Este enunciado favorece a la reducción de datos, mejorando la modelación pero obteniendo errores de truncamiento.

El error por modelamiento o de truncamiento se produce por tomar atribuciones en la modelación de figuras. Esta falta es directamente proporcional a la longitud infinitesimal y se debe tomar en cuenta en las aplicaciones donde se utilice el método de las diferencias finitas.

Las Tablas 4.1 y 4.2 muestran las consecuencias del modelamiento de una circunferencia.

Tabla 4.1. Consecuencias del grosor de la línea que conforma una circunferencia [Elaboración propia].

Grosor de la línea que conforma la circunferencia	Modelamiento de la figura
Grande	En vez de parecer una circunferencia parecerá un disco
Pequeño	Se puede perder el modelamiento

Tabla 4.2. Consecuencias de la longitud infinitesimal en el modelamiento de una circunferencia [Elaboración propia].

Longitud infinitesimal	Resolución de la figura	Número de datos	Número de cálculos	Tiempo de ejecución	Error por redondeo	Error por discretización
Grande			Disminuye			Aumenta
Pequeña			Aumenta			Disminuye

4.2 Ejemplos de aplicación

La herramienta de software desarrollada en este trabajo de titulación se utilizará en cinco ejemplos. El primero en un problema bidimensional propuesto en [6] con el fin de comparar resultados de potencial. El segundo y el tercero en la parte transversal de un cable coaxial con el afán de comparar resultados con los obtenidos en [14] donde se utiliza el programa computacional FEMM. El cuarto para un sistema asimétrico de teoría electromagnética, y el quinto en un transformador monofásico de potencia. Los ejemplos planteados en este capítulo tienen el fin de comprobar el correcto funcionamiento de CEDIF.

4.2.1 Implementación de la herramienta de software a un problema simétrico de teoría electromagnética

Este ejemplo pretende comparar los resultados obtenidos en la matriz de voltajes con el ejemplo planteado en [6], donde se estima valores de voltaje en diferentes puntos de un enrejado.

El ejemplo presentado en esta sección se estructura como un problema bidimensional en el cual el potencial no cambia con la coordenada z . Así se forma una sección transversal conformada por cuadros con lados de longitud h . De tal manera que esta región es donde se desea conocer el potencial.

Los datos del problema son:

- Región cuadrada con fronteras conductoras.
- El potencial en la parte superior es 100 V, el de los lados y el fondo es 0 V.
- Los voltajes en el centro de la rejilla son nulos.

En primer lugar, la región se divide en 16 cuadrados, es decir se forma una rejilla de 4x4. Ver Figura 4.4.

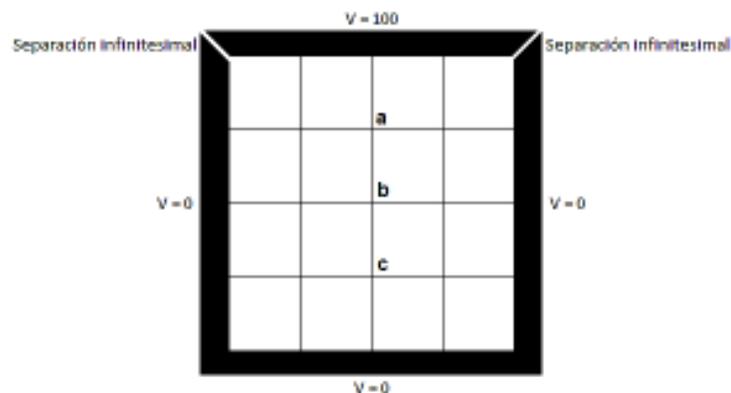


Figura 4.4. Enrejado de 4x4 [6].

En segundo lugar, el potencial es estimado aplicando análisis numérico para un enrejado de 4x4, 8x8 y 16x16 [6]. Los resultados obtenidos en [6] se presentan en la Tabla 4.3.

Los resultados obtenidos por CEDIF para el problema anteriormente planteado se presentan en la Tabla 4.3.

Tabla 4.3. Resultados de voltaje obtenidos en [6] y en la herramienta de software [Elaboración propia].

Enrejado	Voltaje en referencia (V) [6].			Voltaje con CEDIF (V) [Elaboración propia].				
	a	b	c	a	b	c	Número de datos	Tiempo de ejecución (s)
4x4	52.6	25	9.8	52.68	25	9.82	25	11.41
8x8	53.6	25	9.7	53.61	25	9.63	81	12
16x16	53.93	25	9.56	53.93	25	9.56	289	12.71
32x32				54.02	25	9.55	1089	18.93
64x64				54.04	25	9.54	4225	29.62
Exacto	54.05	25	9.54					

Cabe mencionar que CEDIF analiza la permitividad de los materiales. Por esta razón para el ejemplo se consideró bordes conductores de cobre ($\epsilon_r = 1$) y para el enrejado el vacío, es decir permitividad relativa de uno.

La Tabla 4.3 comprueba el correcto accionar de la herramienta de software para encontrar voltajes en los diferentes puntos de rejilla.

La Tabla 4.4 muestra los resultados obtenidos para el enrejado de 16x16. Estos resultados se mostraron en la Tabla 4.3.

Tabla 4.4. Resultados de voltaje obtenidos por CEDIF en todos los puntos de la rejilla (enrejado de 16x16) [Datos obtenidos de Python].

100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00	100,00
0,00	49,57	68,90	77,73	82,32	84,91	86,39	87,17	87,41	87,17	86,39	84,91	82,32	77,73	68,90	49,57	0,00
0,00	29,39	48,28	59,70	66,65	70,92	73,49	74,86	75,30	74,86	73,49	70,92	66,65	59,70	48,28	29,39	0,00
0,00	19,69	35,15	46,14	53,65	58,63	61,77	63,51	64,06	63,51	61,77	58,63	53,65	46,14	35,15	19,69	0,00
0,00	14,23	26,48	36,08	43,18	48,18	51,47	53,33	53,93	53,33	51,47	48,18	43,18	36,08	26,48	14,23	0,00
0,00	10,76	20,45	28,51	34,81	39,44	42,59	44,41	45,01	44,41	42,59	39,44	34,81	28,51	20,45	10,76	0,00
0,00	8,35	16,06	22,70	28,09	32,19	35,04	36,72	37,27	36,72	35,04	32,19	28,09	22,70	16,06	8,35	0,00
0,00	6,57	12,73	18,16	22,67	26,18	28,67	30,15	30,64	30,15	28,67	26,18	22,67	18,16	12,73	6,57	0,00
0,00	5,21	10,13	14,53	18,25	21,19	23,31	24,58	25,00	24,58	23,31	21,19	18,25	14,53	10,13	5,21	0,00
0,00	4,12	8,05	11,59	14,61	17,03	18,79	19,85	20,20	19,85	18,79	17,03	14,61	11,59	8,05	4,12	0,00
0,00	3,25	6,34	9,16	11,58	13,53	14,96	15,82	16,11	15,82	14,96	13,53	11,58	9,16	6,34	3,25	0,00
0,00	2,52	4,92	7,12	9,02	10,56	11,69	12,38	12,61	12,38	11,69	10,56	9,02	7,12	4,92	2,52	0,00
0,00	1,90	3,72	5,38	6,82	8,00	8,86	9,39	9,56	9,39	8,86	7,99	6,82	5,38	3,72	1,90	0,00
0,00	1,36	2,66	3,86	4,89	5,74	6,36	6,75	6,88	6,75	6,36	5,74	4,89	3,86	2,66	1,36	0,00
0,00	0,88	1,72	2,49	3,16	3,71	4,11	4,36	4,45	4,36	4,11	3,71	3,16	2,49	1,72	0,88	0,00
0,00	0,43	0,84	1,22	1,55	1,82	2,02	2,14	2,18	2,14	2,02	1,82	1,55	1,22	0,84	0,43	0,00
0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00

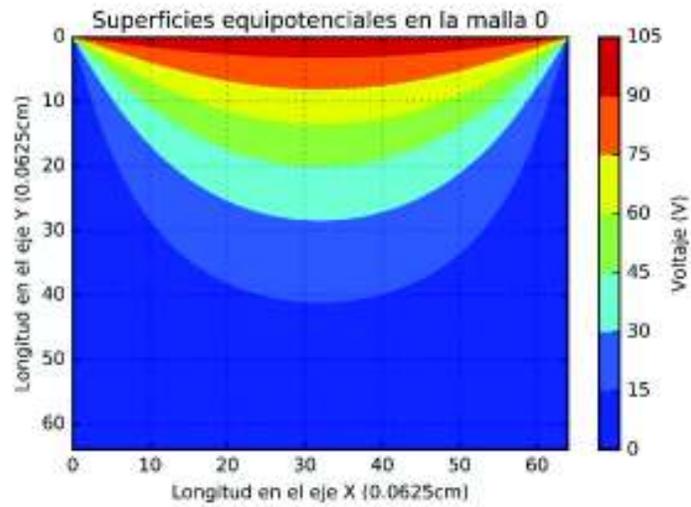


Figura 4.5. Voltaje en un enrejado de 64x64 [Resultado obtenido por CEDIF].

La Figura 4.5 muestra de forma factible los valores de la Tabla 4.4, así se hace más fácil observar las superficies equipotenciales.

CEDIF produce el comportamiento del campo electrostático y superficies equipotenciales. En la Figura 4.6 se observan las superficies equipotenciales y el campo electrostático. En la Figura 4.7 se observa la magnitud del campo electrostático, ambos casos para el ejemplo bidimensional simétrico de teoría electromagnética.

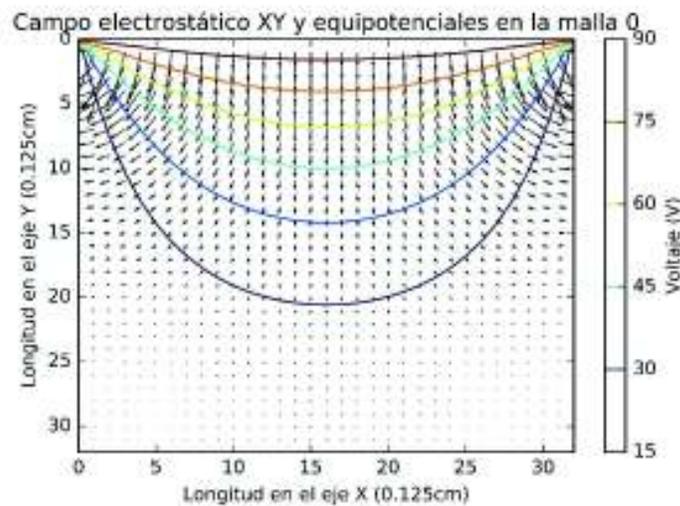


Figura 4.6. Campo electrostático y superficies equipotenciales en un enrejado de 32x32 [Resultado obtenido por CEDIF].

El campo electrostático que se obtiene en este problema bidimensional surge desde el centro de la barra a 100 V (parte superior del gráfico) y se extiende simétricamente por todo el entorno dirigiéndose hacia los lados a 0 V. Además se observa que el campo electrostático cruza perpendicularmente cada superficie equipotencial.

En las esquinas superiores donde existe una mayor diferencia de potencial (100 V a 0 V) se observa una mayor cantidad de líneas de flujo.

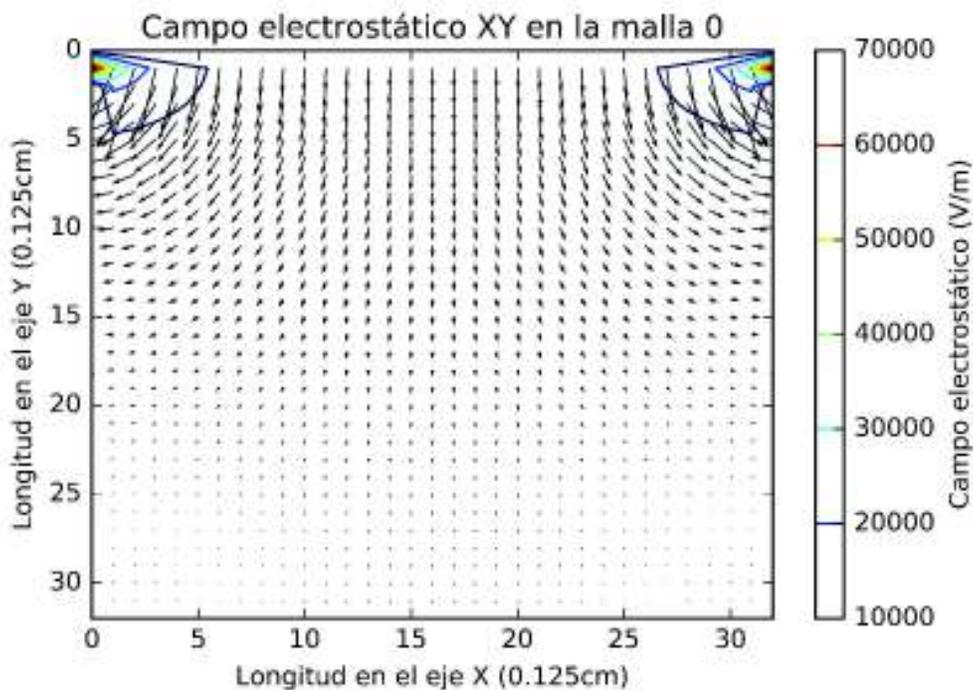


Figura 4.7. Campo electrostático en un enrejado de 32x32 [Resultado obtenido por CEDIF].

En la Figura 4.7 se muestra el comportamiento del campo electrostático y la magnitud del mismo en el entorno planteado. Se nota que la mayor magnitud del campo se encuentra en las esquinas donde hay mayor cantidad de líneas de flujo, es decir, mayor gradiente. Además el campo es menor en las zonas de bajo voltaje (Esquinas inferiores del entorno).

4.2.2 Implementación de la herramienta de software en la parte transversal de un cable coaxial

La evaluación de la herramienta se realiza mediante su utilización en un cable coaxial. Para este ejemplo se ingresará el valor de permitividad en el aislante y en el conductor del cable. Los resultados obtenidos se compararán con los obtenidos en las simulaciones del programa FEMM [14].

4.2.2.1 Cable de un núcleo conductor y un material aislante

El problema de un cable monopolar se especifica con los siguientes datos [14]:

- Cable de un solo núcleo conductor y un solo material aislante.
- Voltaje del conductor 66 kV ($\epsilon_r = 1$)
- Material aislante XLPE de permitividad eléctrica $\epsilon_r = 2.5$
- Radio del conductor de 1.65 cm y espesor del aislamiento de 2.85 cm

Los resultados obtenidos por CEDIF en este ejemplo se comprobarán con los obtenidos en el programa FEMM [14], con el fin de evaluar el comportamiento de la misma. Los resultados conseguidos se muestran las Figuras 4.8 y 4.9.

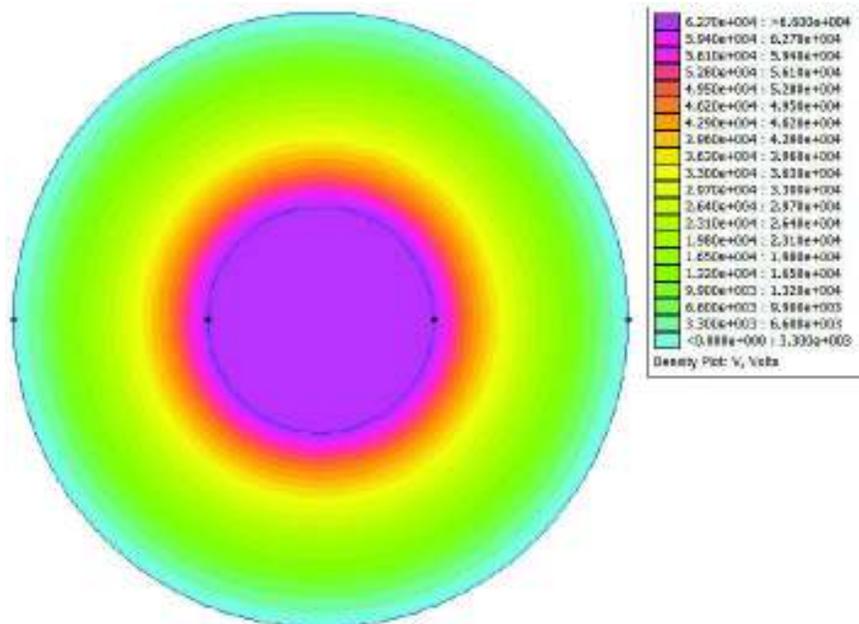


Figura 4.8. Resultados de voltaje en un cable monopolar con aislamiento XLPE simulando en el programa FEMM. [14]

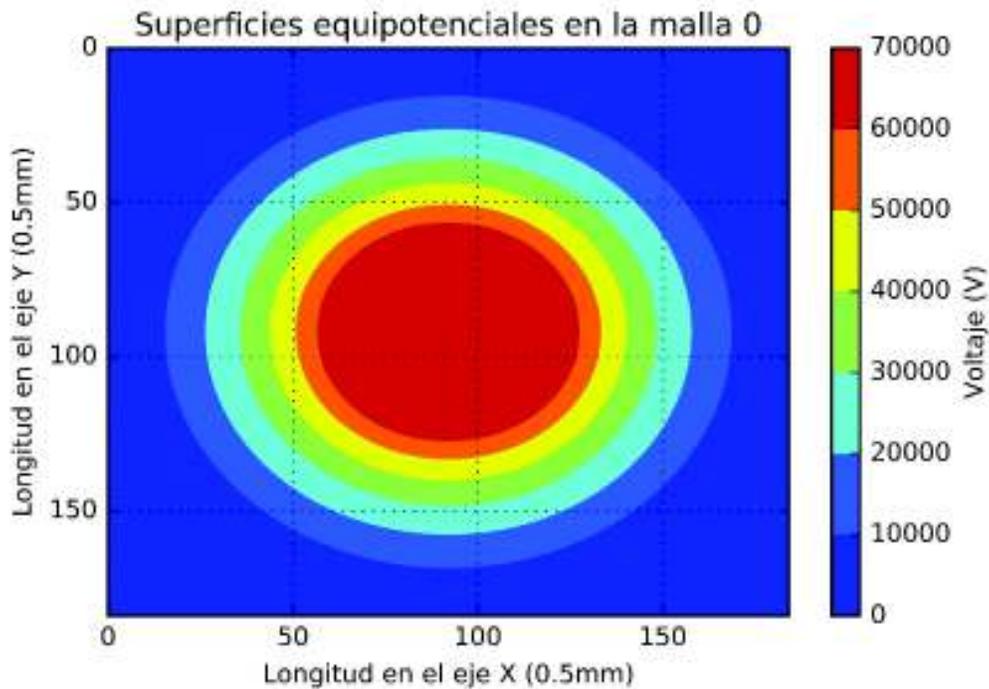


Figura 4.9. Voltaje en un cable monopolar con aislamiento XLPE [Resultado obtenido por CEDIF].

Este ejemplo se puede comprobar por medio de la ecuación de Laplace aplicada a una línea de transmisión coaxial. La ecuación 4.1 describe el voltaje en función de los radios y en función del voltaje del conductor [6].

$$V = V_o \times \frac{\ln\left(\frac{b}{r}\right)}{\ln\left(\frac{b}{a}\right)}$$

Ecuación 4.1. Voltaje en el aislamiento de un cable coaxial

Donde:

- V : voltaje en cualquier punto del aislante
- Vo: voltaje del conductor
- r : radio en cualquier punto del aislante ($b \geq r \geq a$)
- a : radio del conductor
- b : radio del aislante ($b > a$)

Las respuestas de voltaje obtenidas de forma analítica, por medio del programa FEMM y por medio de CEDIF se muestran en la Figura 4.10.

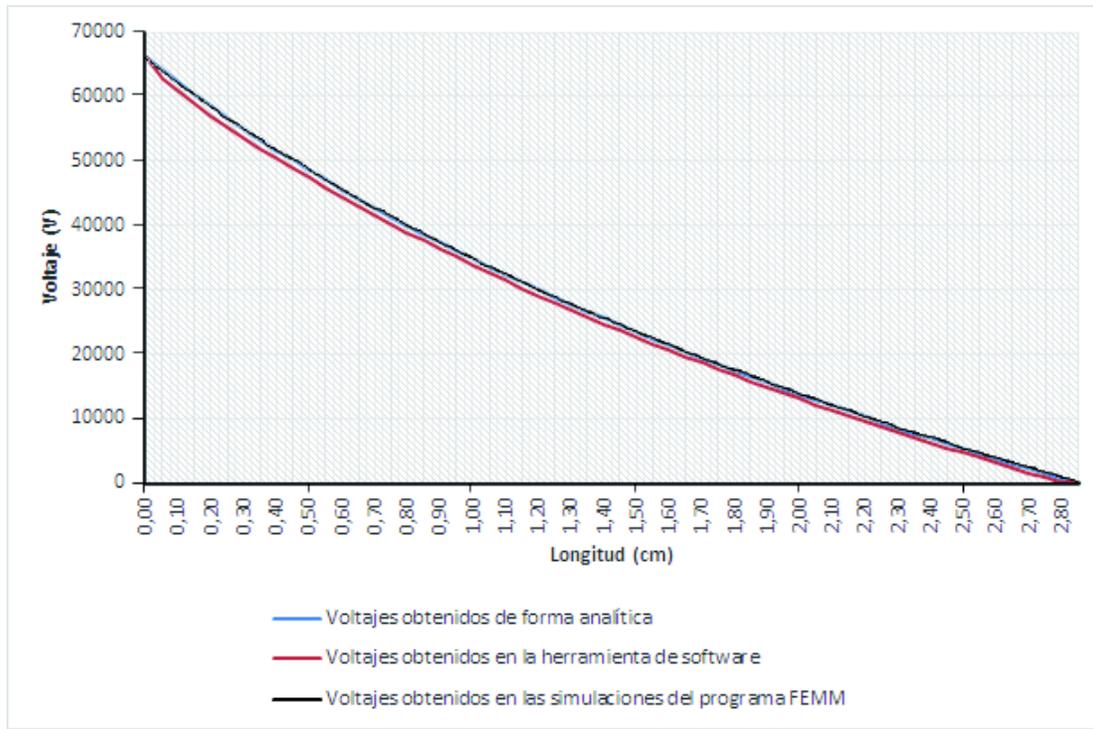


Figura 4.10. Comparación de las respuestas de voltaje [Elaboración propia].

La longitud está tomada desde el borde del conductor hasta la periferia del aislamiento, es decir las gráficas corresponden al comportamiento del voltaje a lo largo del aislamiento del cable.

Los resultados de voltaje son parecidos a los obtenidos analíticamente y a los obtenidos mediante simulaciones en el programa FEMM (Figura 4.10). Entonces CEDIF valida su funcionamiento hasta la obtención de voltajes.

Ahora bien, CEDIF debe comprobar su funcionamiento en cuanto a los resultados del campo eléctrico. Esta comprobación se realiza al mismo ejemplo del cable monopolar, en el que la respuesta analítica viene dada por la Ecuación 4.2. [6]

$$E = \frac{V_0}{r} \times \frac{1}{\ln\left(\frac{b}{a}\right)}$$

Ecuación 4.2. Campo eléctrico en el aislamiento de un cable coaxial.

Donde:

- E : campo eléctrico en cualquier punto del aislante.
- V_0 : voltaje del conductor.
- r : radio en cualquier punto del aislante ($b \geq r \geq a$).
- a : radio del conductor.
- b : radio del aislante ($b > a$).

Los resultados de campo eléctrico obtenidos se presentan en las Figuras 4.11 y 4.12.

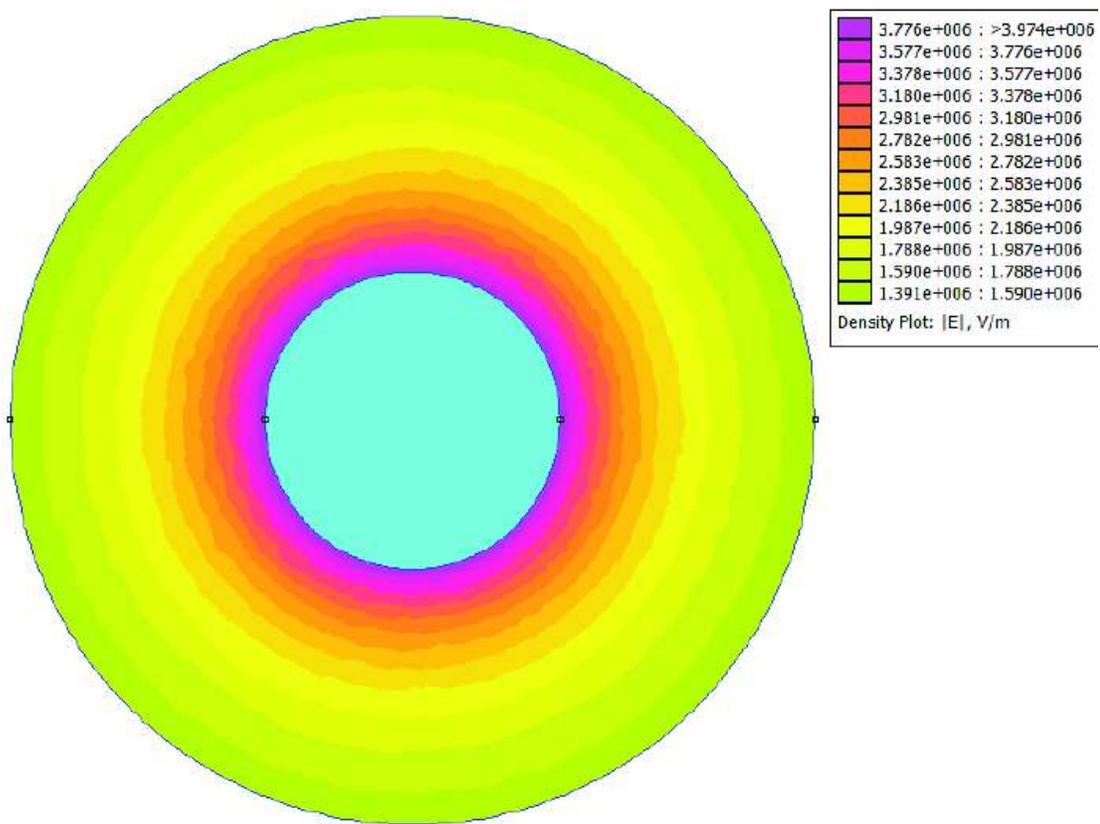


Figura 4.11. Resultados de campo eléctrico en un cable monopolar con aislamiento XLPE simulando en el programa FEMM. [14]

La Figura 4.11 muestra los resultados obtenidos del cable monopolar a través de la simulación del programa FEMM. En esta se puede observar la magnitud del campo en todo

el aislante. Es decir el aislamiento posee valores desde $1.391 \times 10^6 \left(\frac{V}{m}\right)$ hasta valores que superan $3.776 \times 10^6 \left(\frac{V}{m}\right)$.

Específicamente las magnitudes del campo eléctrico en los límites del aislamiento se presentan en la Tabla 4.5. Estos valores son los mismos que los mostrados en las Figuras 4.11 y 4.12. Además estos resultados servirán para aceptar los obtenidos por CEDIF (Figura 4.12)

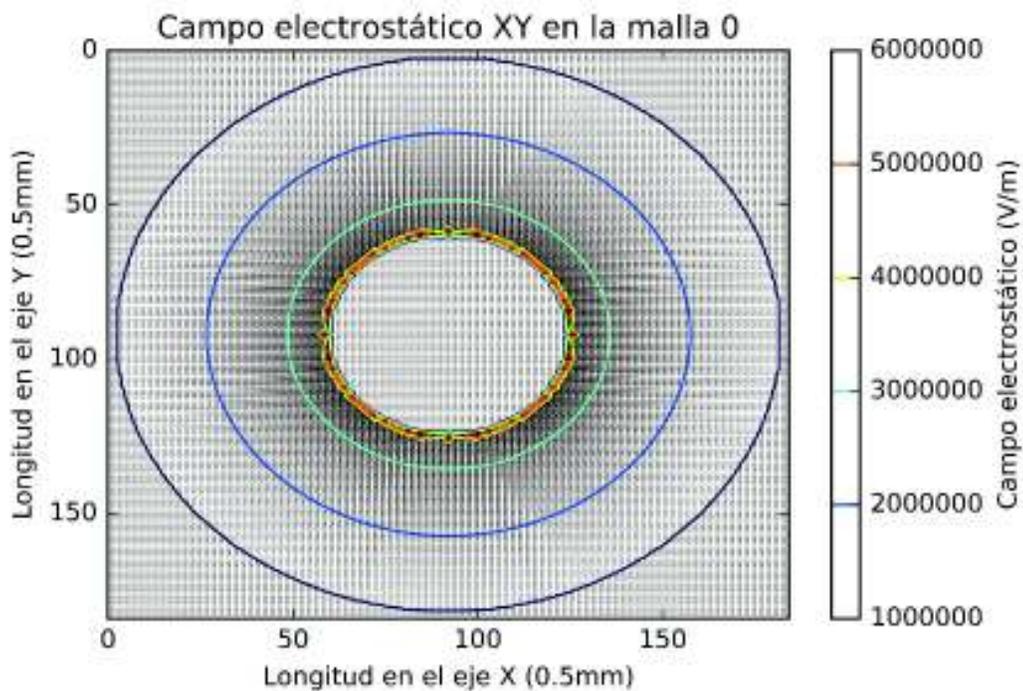


Figura 4.12. Campo electrostático en un cable monopolar con aislamiento XLPE
[Resultado obtenido por CEDIF].

Es importante que los valores del campo electrostático sean comparados con otros con el fin de evidenciar el funcionamiento de CEDIF. Los valores obtenidos de forma analítica, por el programa FEMM y por la nombrada herramienta se muestran en la Tabla 4.5 y en la Figura 4.13. Cabe recalcar que los valores correspondientes a la herramienta de software son obtenidos de la recta paralela al eje x del lado derecho del aislamiento.

Tabla 4.5. Resultados de campo eléctrico en el aislamiento [Elaboración propia].

Límite del aislamiento	Campo eléctrico (V/m)			Distancia del conductor a la parte exterior del aislamiento (cm)
	Análiticamente	Simulaciones del programa FEMM [14] (Figura 4.11)	CEDIF	
Parte más cercana al conductor	3.87×10^6	$> 3.776 \times 10^6$	6.562×10^6	0.05
Parte externa del aislamiento	1.495×10^6	entre 1.391 y 1.59×10^6	$< 1.5 \times 10^6$	2.75

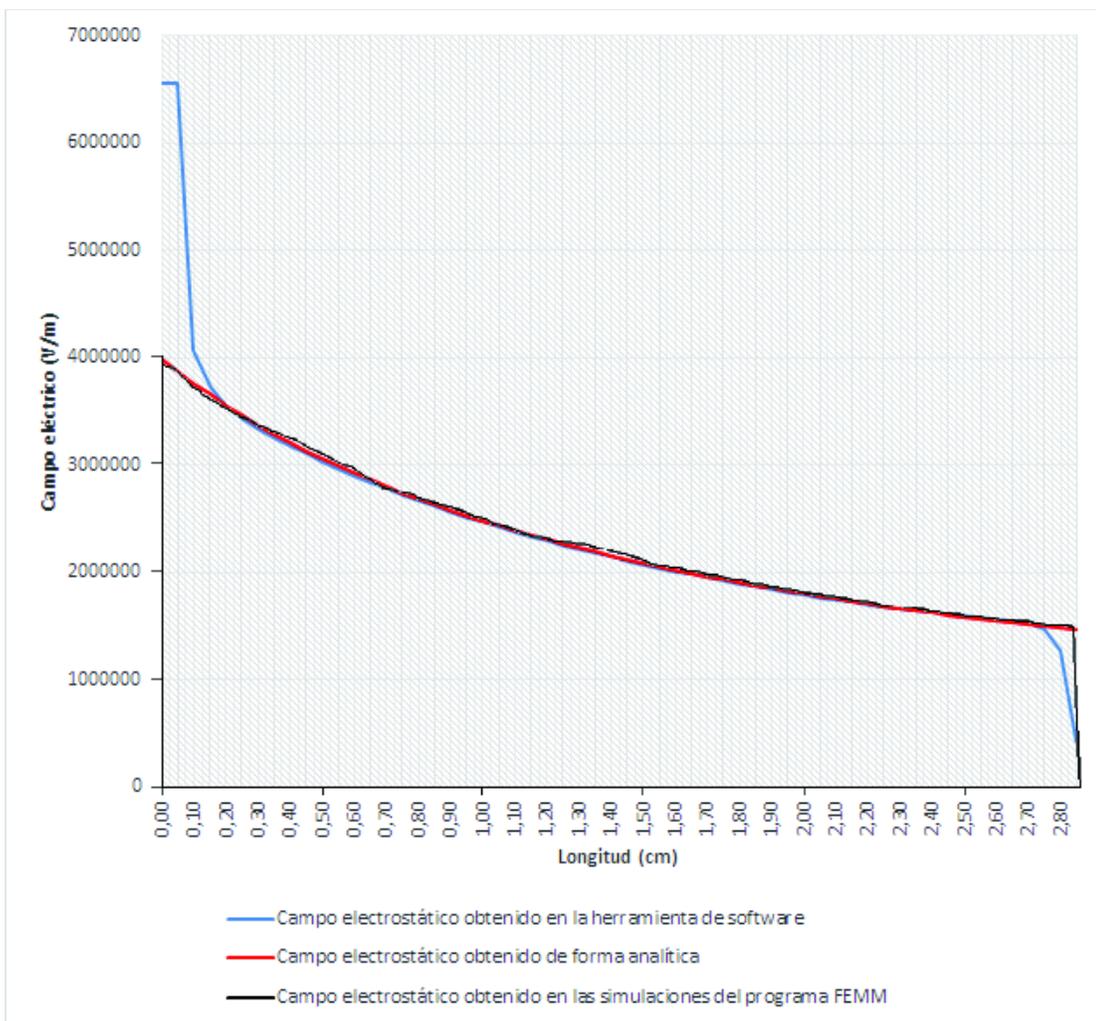


Figura 4.13. Comparación de resultados del campo eléctrico para un cable monopolar con una capa de aislamiento [Elaboración propia].

Los valores del campo electrostático que se obtienen en los límites del aislamiento según CEDIF están comprendidos en los resultados del programa FEMM. Por esto a los resultados de la herramienta se aceptan como aproximaciones válidas. Estas afirmaciones se observan en la Tabla 4.5 y en la Figura 4.13.

En general, la respuesta obtenida al compararla con la solución analítica presenta similitud en las regiones internas del aislamiento. El problema radica en los límites del aislamiento. Los errores de la magnitud del campo eléctrico se deben a la capa de grosor que se debe especificar en el método de las diferencias finitas al formar una circunferencia (aterrización del aislamiento) y un círculo (conductor a 66 kV). De esta manera se cometen errores de modelamiento y se observan debido a un desfase en los límites del aislamiento.

CEDIF para este ejemplo en particular maneja aproximadamente 34 225 datos, los cuales son procesados en un tiempo estimado de 4 minutos con 32.97 segundos.

4.2.2.2 Cable con un núcleo conductor y dos capas de aislamiento

En esta sección, se presenta un ejemplo con el fin de comprobar y comparar los resultados obtenidos por CEDIF con los del programa FEMM.

El ejemplo se efectúa en un cable monopolar con dos capas diferentes de aislamiento con las siguientes características [14]:

- Voltaje del conductor de cobre ($\epsilon_r = 1$) 15 kV.
- Radio externo del cable de 1.99 cm.
- Espesor de cada aislante de 0.745 cm.
- Radio del conductor de 0.5 cm
- Las capas del aislamiento se las distribuirá colocando la de mayor permitividad en la cercanía del núcleo y la de menor permitividad en la parte exterior del aislamiento. Los materiales aislantes serán el EPR ($\epsilon_r = 3$) y el XLPE ($\epsilon_r = 2.3$).

El problema se describe en la Figura 4.14.

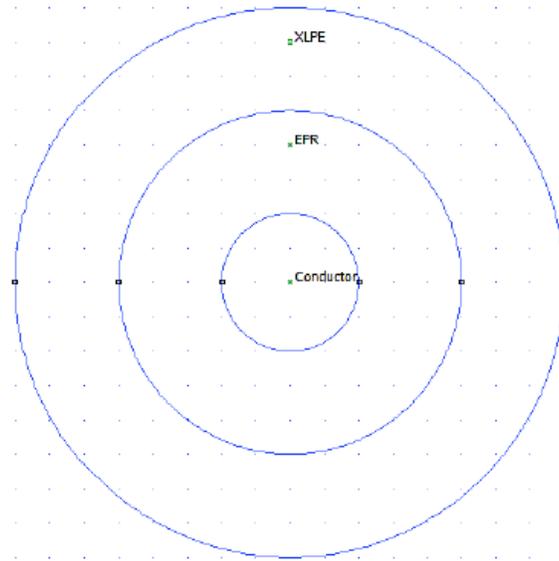


Figura 4.14. Cable monopolar con 2 capas de aislamiento (EPR y XLPE). [14].

CEDIF ha validado su proceso hasta la obtención de voltajes, por lo tanto únicamente se presentarán los resultados del campo eléctrico y no los de voltaje. Estos valores son los conseguidos de las simulaciones del programa FEMM y de la herramienta de software. Figuras 4.15 y 4.16.

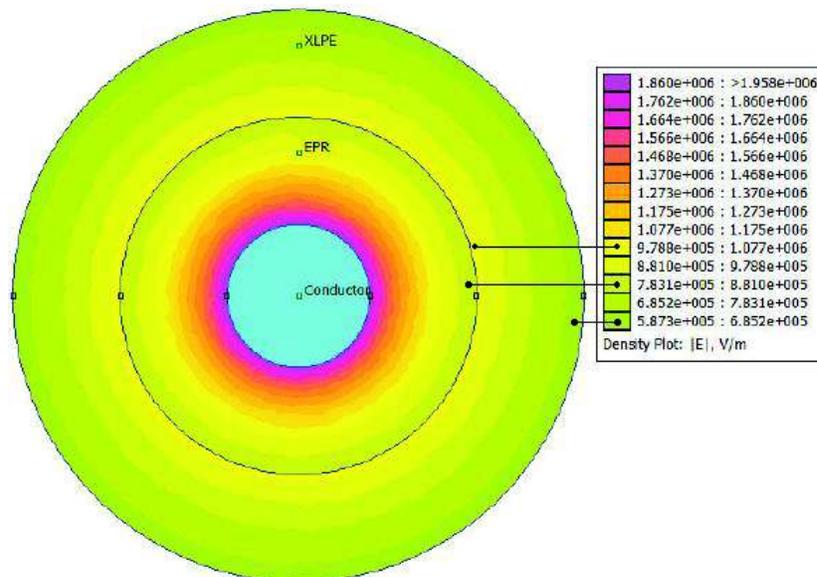


Figura 4.15. Comportamiento del campo eléctrico en un cable monopolar con dos capas de aislamiento (EPR y XLPE) (Simulaciones en el programa FEMM). [14]

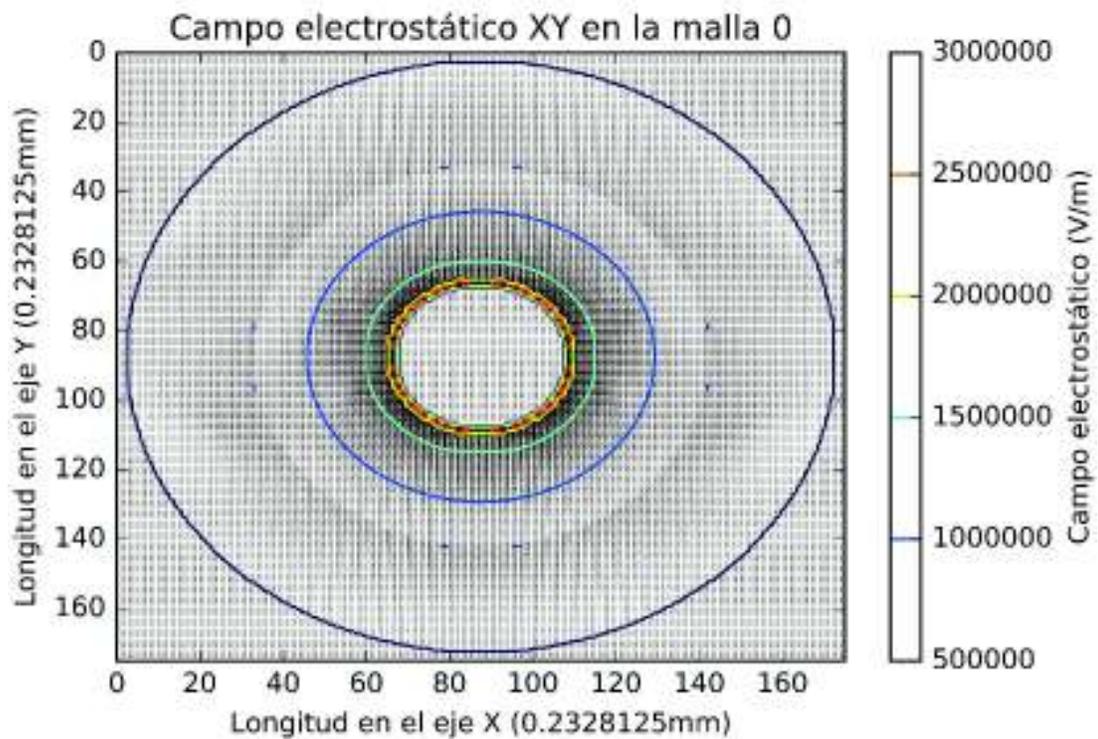


Figura 4.16. Comportamiento del campo eléctrico en un cable monopolar con dos capas de aislamiento (EPR y XLPE) [Resultado obtenido por CEDIF].

CEDIF para resolver este problema se demoró aproximadamente 5 minutos manejando cerca de 31 329 datos.

Los resultados obtenidos en CEDIF y el programa FEMM se comparan y se presentan en la Figura 4.17.

Vale señalar que los valores correspondientes a la herramienta de software son obtenidos de la recta perpendicular al eje y del lado derecho del aislamiento.

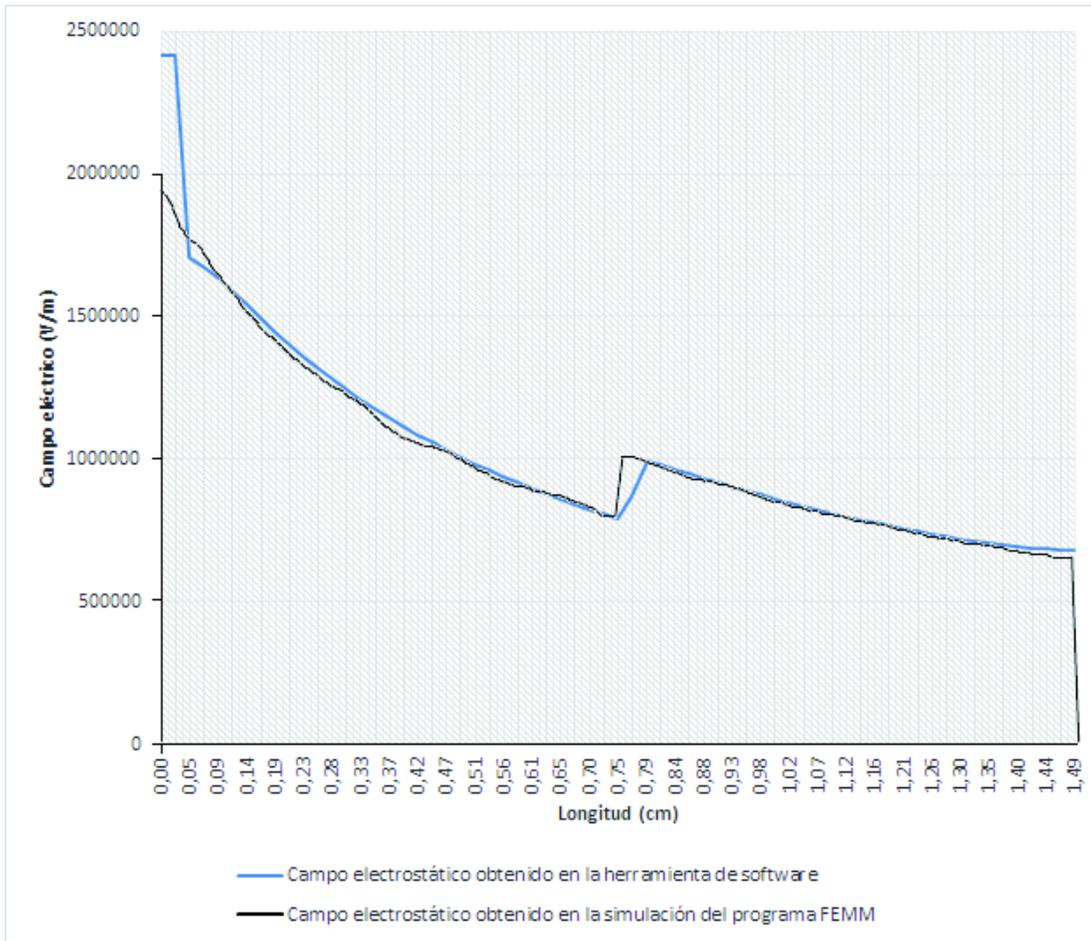


Figura 4.17. Comparación de resultados del campo eléctrico para un cable monopolar con dos capas de aislamiento [Elaboración propia].

En la Tabla 4.6 se muestra una comparativa entre los resultados obtenidos en el programa FEMM y los obtenidos por CEDIF en los límites de los materiales.

Tabla 4.6. Valores aproximados de campo electrostático en los límites de los materiales para un cable monopolar con dos capas de aislamiento según el programa FEMM y CEDIF [Elaboración propia].

Aislante	Límite	(V/m)	
		Programa FEMM	CEDIF
EPR	Conductor	$\geq 1.86 \times 10^6$	2.413×10^6
	XLPE	$7.831 - 8.81 \times 10^5$	7.896×10^5
XLPE	EPR	$9.788 - 10.77 \times 10^5$	8.697×10^5
	Vacío	$5.873 - 6.852 \times 10^5$	6.823×10^5

Los resultados obtenidos con el programa FEMM y los resultados de CEDIF son similares en las regiones centrales de los materiales aislantes. Existiendo un error en los límites donde el entorno cambia de material.

Se presenta un error en las fronteras del material aislante. Esta falla se presenta en tres partes para el cable monopolar con dos capas de aislamiento. El primero en la unión del conductor con el aislamiento EPR. El segundo entre los aislamientos EPR y XLPE. Y el último en el límite del aislante XLPE.

Los errores antes mencionados se deben a la utilización del método de las diferencias finitas. Ya que no se puede simular exactamente una circunferencia o un círculo debido a que se obtendrían errores de modelamiento (grosor en líneas que conforman una circunferencia). Este método utiliza longitudes infinitesimales lo que produce la presencia de errores por truncamiento y por redondeo.

4.2.3 Implementación de la herramienta de software a un problema asimétrico de teoría electromagnética

En esta sección se plantea un ejemplo tridimensional implementado en CEDIF para estudiar el funcionamiento del mismo.

4.2.3.1 Representación del problema

El problema tridimensional se describe geométrica como un cubo aterrizado con aristas X, Y, Z de 12, 10 y 15 cm de longitud cada una respectivamente. En el interior del cubo se encuentran dos esferas conductoras huecas no concéntricas una dentro de la otra. La esfera hueca externa con un diámetro exterior de 8 cm, diámetro interior de 7.125 cm, 69 kV y punto concéntrico en (5, 5.328, 9). La esfera hueca interna con un diámetro exterior de 4 cm, diámetro interior de 2.254 cm, 30 kV y punto concéntrico en (5.856, 4.92, 8.0335).

La Figura 4.18 muestra la descripción técnica del problema. También, los niveles de voltaje y materiales de los aislamientos se muestran en cada elemento.

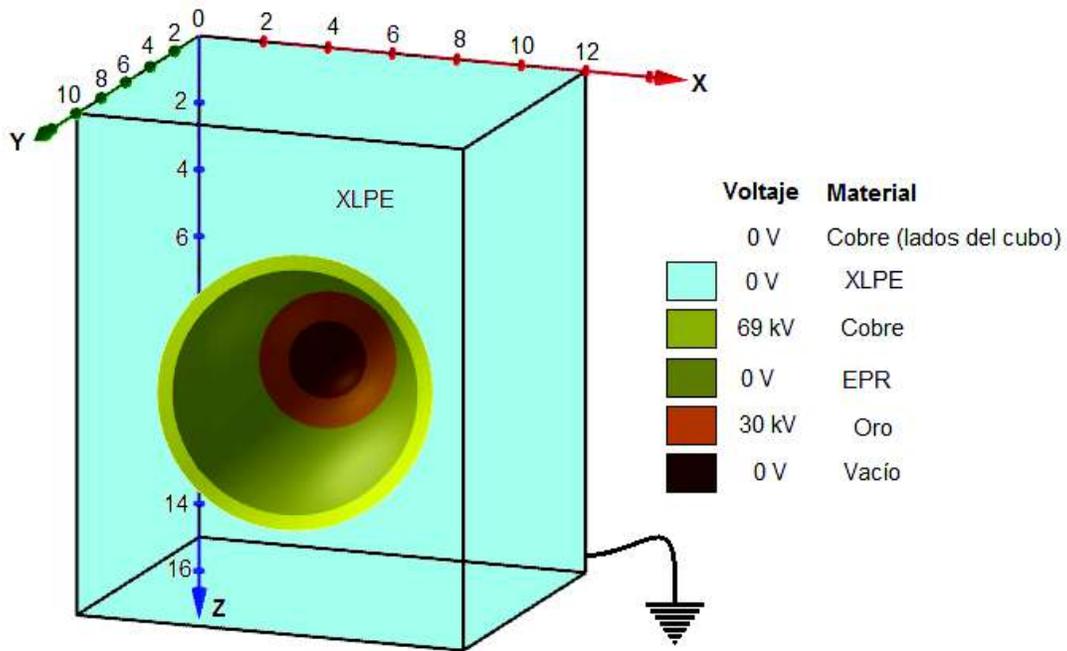


Figura 4.18. Problema asimétrico tridimensional [Elaboración propia].

4.2.3.2 Datos ingresados a la herramienta de software

Los datos ingresados a CEDIF para modelar el problema son:

Número de dimensiones = 3

Unidad de longitud = cm

Longitud en X = 12

Longitud en Y = 10

Longitud en Z = 15

Longitud infinitesimal = 0.2

Tolerancia = 0.001

Parámetro de relajación = 1.9.

4.2.3.3 Resultados

Este problema maneja un número considerable de datos para ilustrar el problema. Específicamente 236 436 valores en una matriz tridimensional de 51x61x76 elementos, los cuales se procesan en un tiempo estimado de 32.5 minutos obteniendo 76 planos (0-75) perpendiculares al eje Z.

Los resultados obtenidos mostrarán superficies equipotenciales y el campo electrostático en los planos de mayor relevancia. Para así evitar la aglomeración de gráficas.

Descripción de resultados:

Los planos analizados se observan en la Figura 4.19.

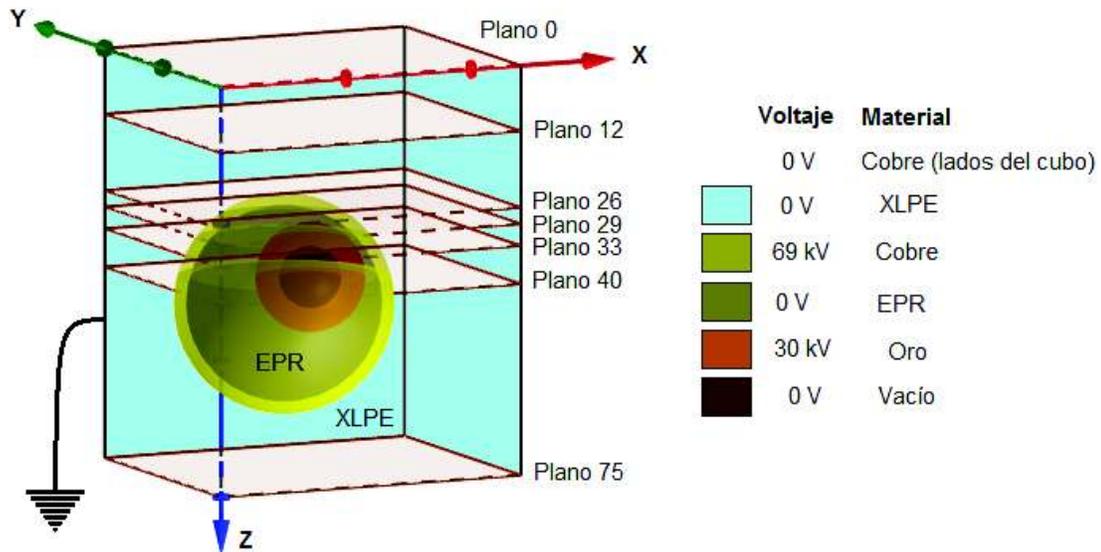


Figura 4.19. Planos que se analizarán en el problema asimétrico tridimensional [Elaboración propia].

- **Plano 0 y plano 75**

Tanto el primer plano como el último son considerados condiciones de frontera, por lo que se mantienen constantes en todo momento al utilizar el método de las diferencias finitas. Entonces estos son superficies equipotenciales de voltaje cero, debido a las condiciones iniciales del problema.

El campo electrostático en estos planos es cero para las componentes X, Y debido a que la variación del voltaje es cero. La componente Z del campo no se calcula porque estos planos son los límites en la dirección Z.

- **Plano 12**

El plano 12 se encuentra en la zona del aislante XLPE e inicialmente con 0 V.

Al utilizar CEDIF se obtienen los voltajes en cada punto de rejilla, así se forman superficies equipotenciales una dentro de la otra en un entorno parcialmente homogéneo. Figura 4.20.

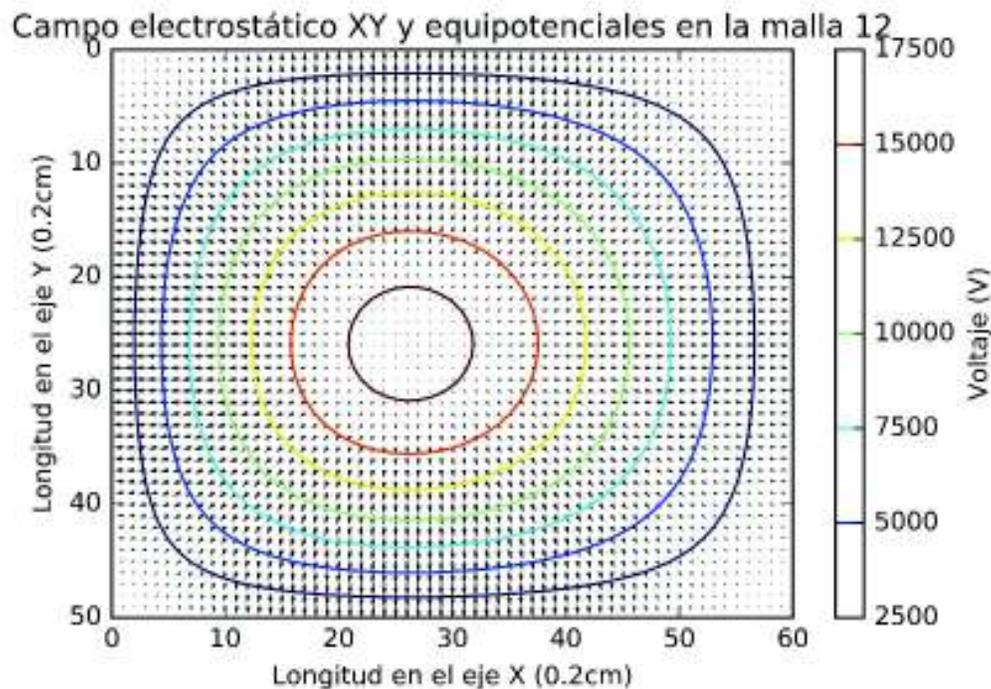


Figura 4.20. Campo electrostático y superficies equipotenciales en el plano 12 [Resultado obtenido por CEDIF].

El campo electrostático sale desde la periferia de la circunferencia equipotencial a 17 500 V formada en el centro del plano. El campo se dirige hacia los lados extremos del plano, de tal manera que parece cruzar de forma perpendicular a cada curva equipotencial. Además, en las esquinas del plano el campo eléctrico tiene los valores de magnitud más bajos. Estas afirmaciones se observan en la Figura 4.21.

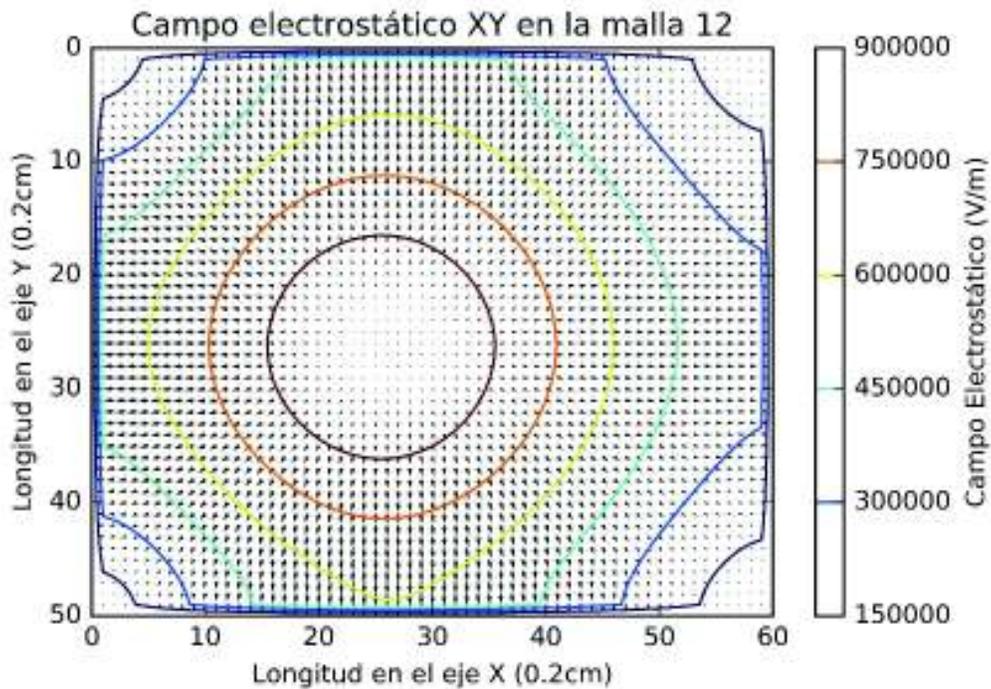


Figura 4.21. Campo electrostático en el plano 12 [Resultado obtenido por CEDIF].

- **Plano 26**

La malla 26 corta de forma transversal la zona superior de la esfera hueca de cobre. Este plano contiene dos materiales, uno el aislante XLPE que inicialmente está a 0 V y el segundo el cobre a 69 kV.

Los resultados obtenidos en cuanto al voltaje muestran superficies equipotenciales cerradas una dentro de la otra, la de mayor magnitud adentro (60 kV) y la de menor magnitud afuera (10 kV). Estos efectos se deben a la distribución de voltajes, ya que la esfera conductora externa posee mayor potencial que los lados del cubo aterrizados (esfera conductora externa se ubica en el interior del cubo). Figura 4.22.

Es importante mencionar que la esfera conductora hueca de cobre a 69 kV es referenciada a partir de la superficie equipotencial a 60 kV. Por tanto la esfera se ubica en el interior de estos 60 000 V. Ver Figura 4.22.

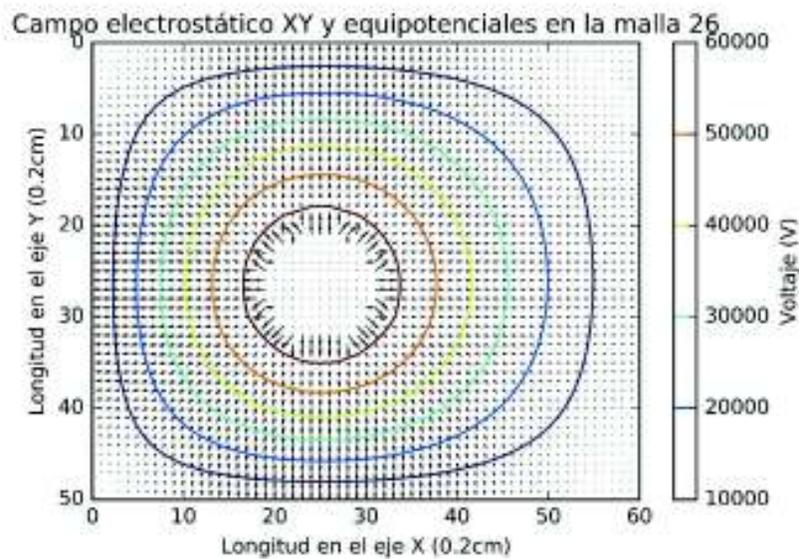


Figura 4.22. Campo electrostático y superficies equipotenciales en el plano 26 [Resultado obtenido por CEDIF].

El campo electrostático en este plano parece salir de la superficie de la esfera hueca perpendicularmente hacia los contornos. Asimismo este solo existe en el aislamiento XLPE, donde los valores máximos se encuentran en las cercanías de la esfera y van decreciendo hasta los extremos aterrizados. De igual forma el campo en el interior de esta esfera es cero, esto se evidencia al imaginar la superficie cerrada a 69 kV en el interior de la superficie equipotencial a 60 kV. Figura 4.23.

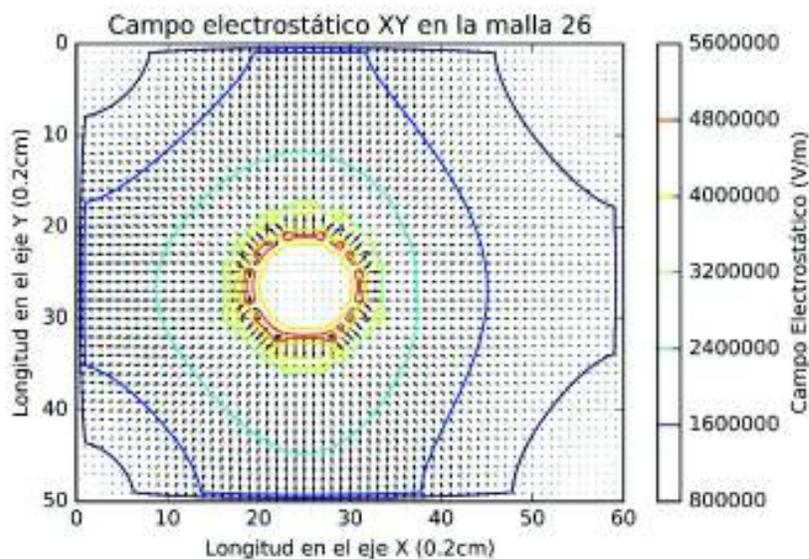


Figura 4.23. Campo electrostático en el plano 26 [Resultado obtenido por CEDIF].

- **Plano 29**

Este plano corta de forma transversal la parte superior del aislante EPR. De esta manera el plano consta de tres materiales uno al interior del otro. El EPR está ubicado en la parte interna, el cobre en el medio y el XLPE en el segmento exterior.

La distribución del voltaje y del campo se divide en dos análisis. El primero entre la esfera hueca de cobre y las paredes del plano. El voltaje en esta región se comporta similar a los casos anteriores, es decir, superficies equipotenciales cerradas una dentro de la otra donde el voltaje mayor se ubica lo más cercano al cobre y la de menor voltaje cerca a los lados del plano. De igual manera el campo parte perpendicularmente desde la periferia del cobre hacia los lados del plano, por ende se dirige desde las superficies de mayor potencial a las de menor potencial pasando siempre de forma perpendicular. Figura 4.24.

El segundo análisis se ubica dentro de la esfera hueca de cobre, en el aislamiento EPR. El campo electrostático en esta región cambia de sentido (de afuera hacia adentro). Este enunciado favorece el anteriormente dicho, ya que el campo va desde un punto de mayor potencial a un punto de menor potencial. Esto se evidencia porque en el exterior del aislamiento EPR se ubica cobre a 69 kV y en el interior oro a 30 kV. Ver Figura 4.25.

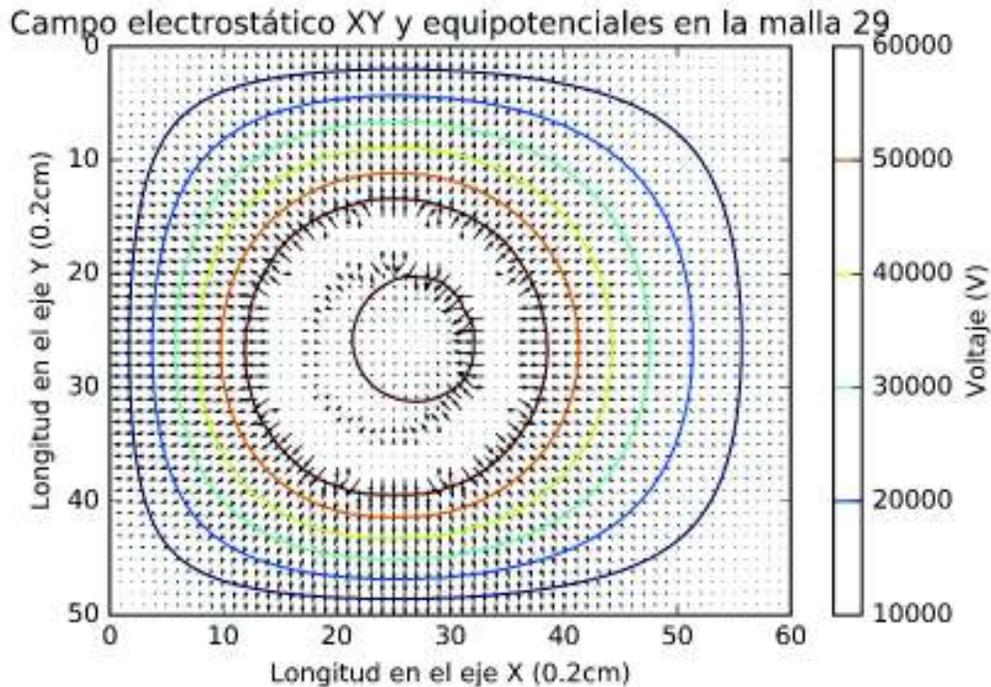


Figura 4.24 Campo electrostático y superficies equipotenciales en el plano 29
[Resultado obtenido por CEDIF].

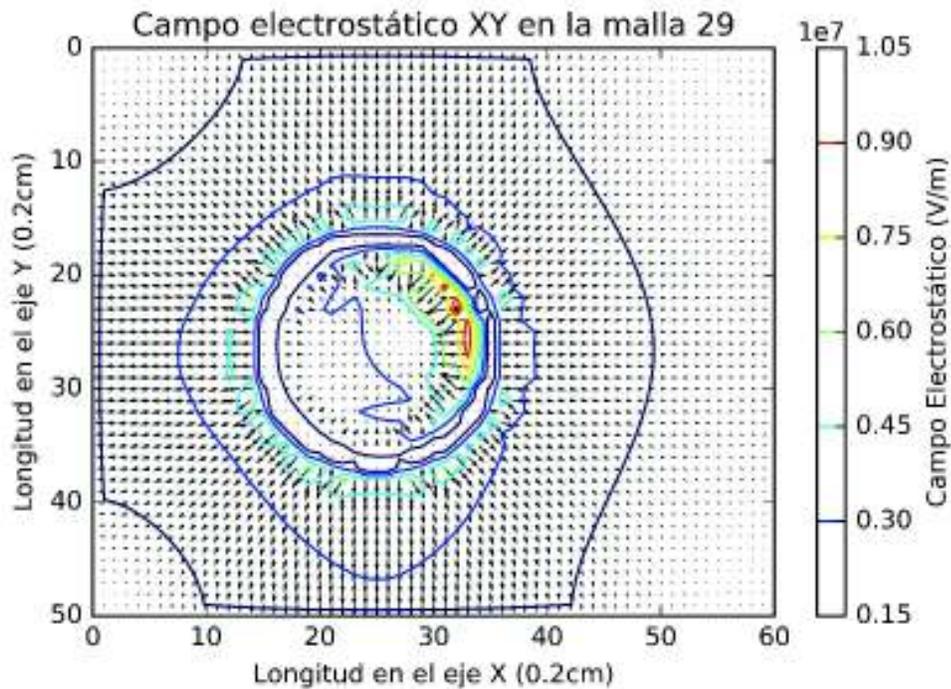


Figura 4.25. Campo electrostático en el plano 29 [Resultado obtenido por CEDIF].

- **Plano 33**

El plano 33 describe la parte superior de la esfera hueca de oro, por tal motivo se encuentra un entorno con 4 permitividades, la del XLPE, la del cobre, la del EPR y la del oro.

El voltaje se comporta de forma similar a los casos anteriores, por tal motivo no se abordará este análisis.

El campo electrostático y su magnitud están representados por flechas y sus tamaños respectivamente. La magnitud del campo es mayor en la región donde las esferas huecas están más cercanas, específicamente en el aislamiento EPR. Esto es por causa de la diferencia de potencial que existe en una distancia menor. Figuras 4.26 y 4.27.

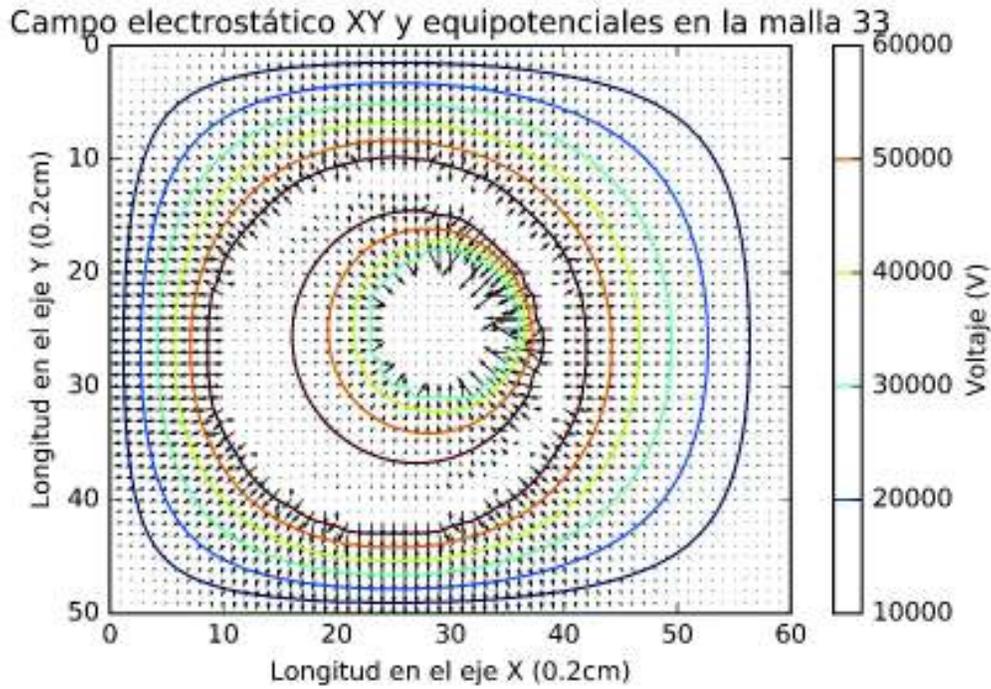


Figura 4.26. Campo electrostático y superficies equipotenciales en el plano 33 [Resultado obtenido por CEDIF].

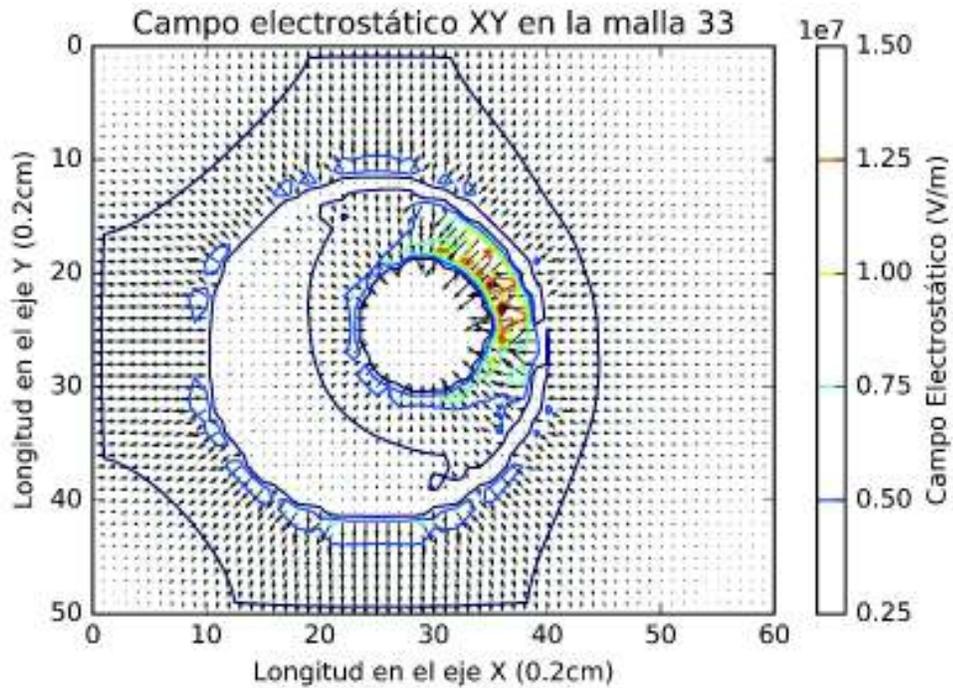


Figura 4.27. Campo electrostático en el plano 33 [Resultado obtenido por CEDIF].

- **Plano 40**

Los resultados del plano 40 se representan en las Figuras 4.28 y 4.29.

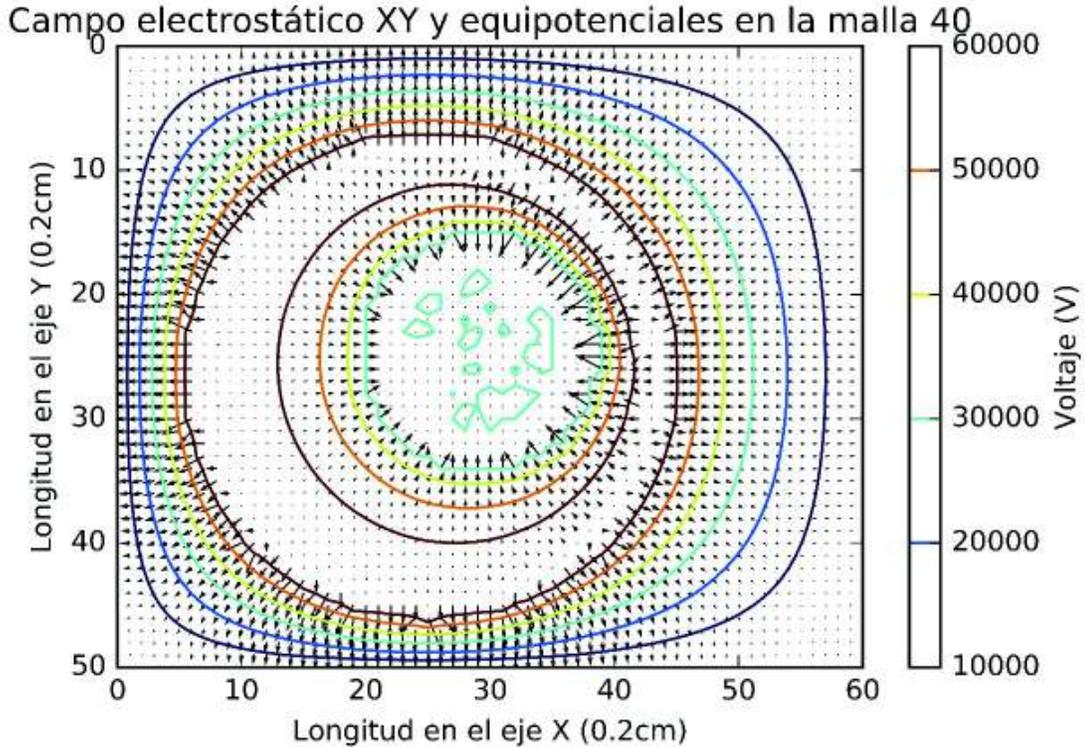


Figura 4.28. Campo electrostático y superficies equipotenciales en el plano 40
[Resultado obtenido por CEDIF].

El plano 40 es la parte transversal de todos los materiales (XLPE, cobre, EPR, oro y vacío).

El resultado más relevante en este plano en cuanto al voltaje es la uniformidad que se obtiene en el interior de la esfera interna. Ya que esta región adquiere un voltaje de 30 kV al estar totalmente cubierta por la esfera hueca de oro. Figura 4.28.

El campo electrostático en el interior de la esfera hueca interna es cero (se observa como puntos en la Figura 4.29), porque se formó una región equipotencial uniforme en el vacío. Puesto que la diferencia de potencial entre puntos vecinos es igualmente cero.

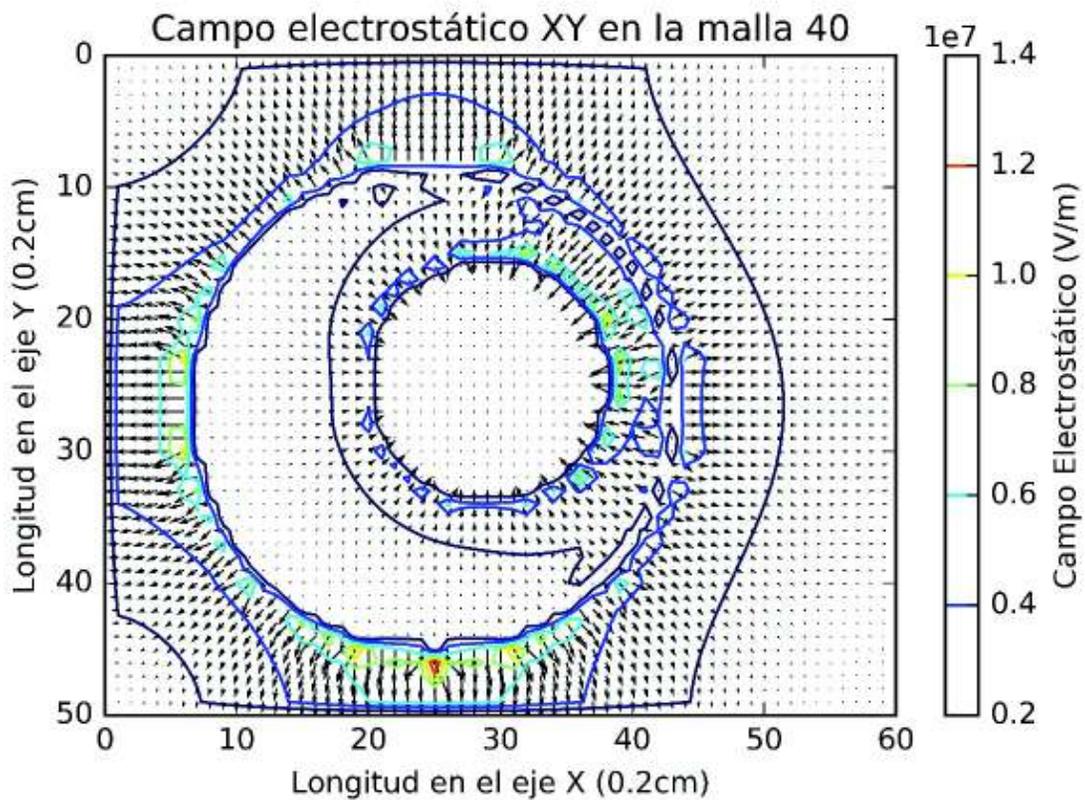


Figura 4.29. Campo electrostático en el plano 40 [Resultado obtenido por CEDIF].

4.2.4 Implementación de la herramienta de software a un transformador monofásico de potencia

CEDIF se aplica para describir el comportamiento del campo electrostático en el interior de un transformador monofásico de potencia tipo acorazado.

A continuación, se presentan los datos generales del transformador.

4.2.4.1 Datos Generales del Transformador

4.2.4.1.1 Valores nominales

Los datos de placa del transformador se destacan en la Tabla 4.7.

Tabla 4.7. Valores nominales del transformador de potencia [15]

Parámetro	Magnitud
Potencia aparente nominal	5 MVA
Voltaje primario	33 kV
Voltaje secundario	11 kV
Frecuencia	50 Hz
Densidad de flujo máxima	1.6 T
Densidad de corriente conductor BV	2.58 A/mm ²
Densidad de corriente conductor AV	2.58 A/mm ²
Reactancia porcentual	7.44 %
Resistencia porcentual	0.678 %
Impedancia porcentual	7.47 %
Máxima eficiencia del transformador	99.5 %
Caída de voltaje entre vacío y plena carga	5.154 %

4.2.4.1.2 Dimensiones de la geometría interna del transformador

La Tabla 4.8 detalla la geometría interna del transformador.

Tabla 4.8. Medidas internas del transformador [15]

Parámetro	Longitud (mm)
Diámetro de la columna	328
Altura de la ventana del núcleo	1080
Distancia entre centros de columna	620
Diámetro interno devanado BV	353
Diámetro externo devanado BV	437
Diámetro interno devanado AV	485
Diámetro externo devanado AV	597
Altura devanado BV	920
Altura devanado AV	921
Ancho devanado BV	42
Ancho devanado AV	56

Los datos faltantes en la Tabla 4.8 se detallan a continuación:

- La distancia entre el tanque y el núcleo es dos veces la distancia entre el núcleo y el devanado de bajo voltaje, 25 mm.

- La distancia entre el tanque y el devanado de alto voltaje es dos veces la distancia entre el núcleo y el devanado de alto voltaje, 157 mm.

4.2.4.2 Modelación del transformador

El transformador necesita ser modelado para que se ingresen los datos en la hoja “Datos iniciales”, seguido de esto se presentan planos que ilustran este equipo.

4.2.4.2.1 Forma gráfica del problema

En las Figuras 4.30, 4.31 y 4.32 se observan las vistas del transformador modelado.

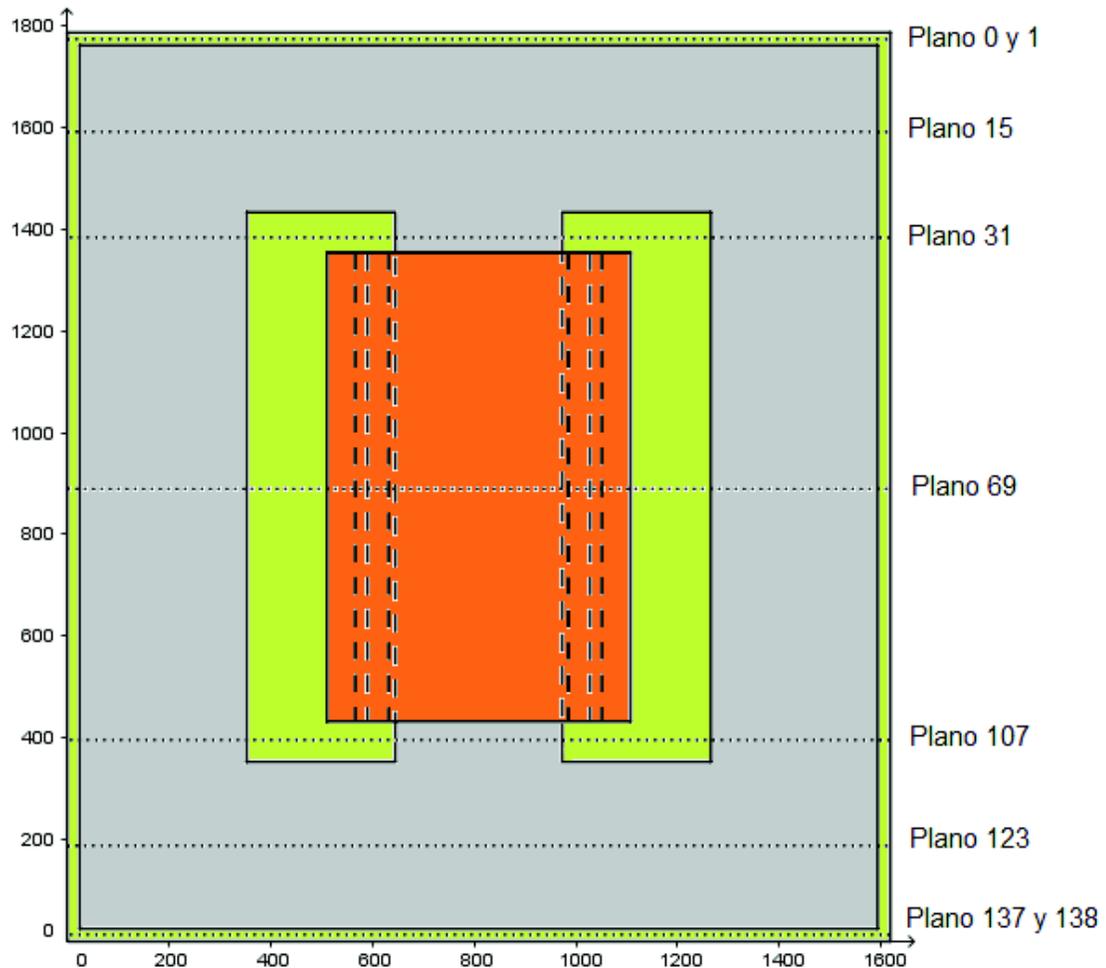


Figura 4.30. Vista frontal, vista posterior y planos que se analizarán [Elaboración propia].

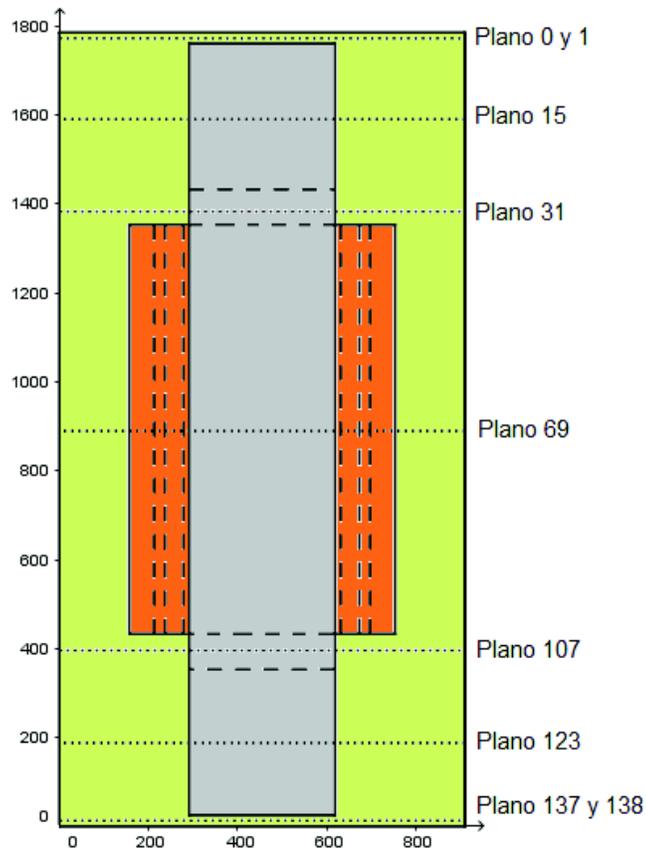


Figura 4.31. Vista lateral derecha, vista lateral izquierda y planos que se analizarán [Elaboración propia].

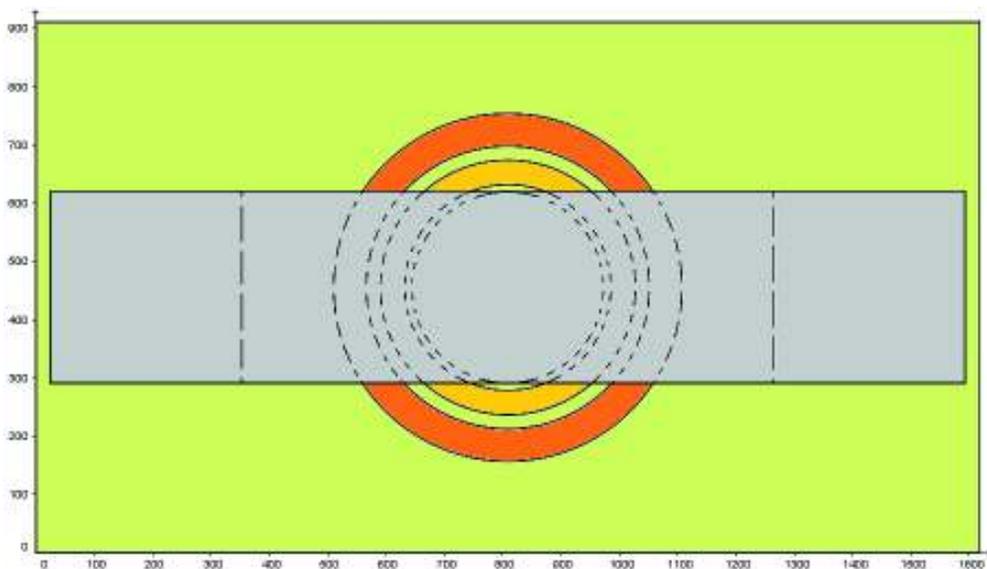


Figura 4.32. Vista superior e inferior [Elaboración propia].

4.2.4.2.2 Voltajes en los devanados

La distribución de voltajes en los devanados del transformador se realiza obteniendo una relación de cambio del voltaje entre cada plano. Los bobinados están divididos en 71 planos desde el 34 hasta el 104 (total de planos 0 al 138). El voltaje pico de cada devanado es dividido en 70, porque el plano de la parte inferior debe tener 0 V. Como resultado se obtiene la razón de cambio en el devanado de alto voltaje 666.701 V y en el bajo voltaje de 222.234 V. Para una mejor comprensión se presenta la Figura 4.33.

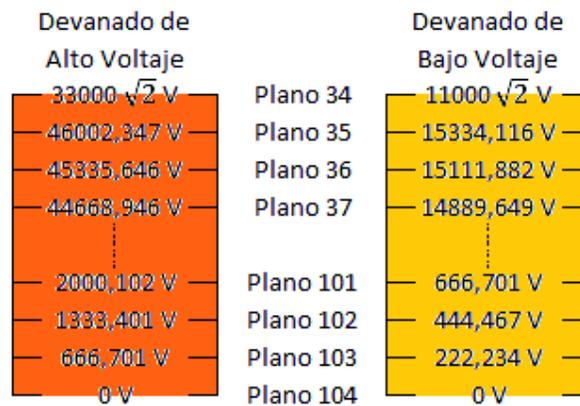


Figura 4.33. Distribución de voltajes en cada devanado (Vista frontal y posterior)
[Elaboración propia].

4.2.4.3 Datos ingresados a la herramienta de software

Los datos ingresados en CEDIF para modelar el transformador monofásico son:

Número de dimensiones = 3

Unidad de longitud = mm

Longitud en X = 1618

Longitud en Y = 911

Longitud en Z = 1786

Longitud infinitesimal = 12.994

Tolerancia = 0.001

Parámetro de relajación = 1.9.

4.2.4.4 Resultados

CEDIF maneja un enorme número de datos, los cuales procesa para obtener resultados. Por ejemplo 1 261 008 datos son ubicados en 139 planos (0 - 138) de 72 x 126 elementos cada uno. Aproximadamente, la herramienta demoró 1 hora, 47.5 minutos en depurar este problema tridimensional.

A continuación se presentan los resultados de mayor importancia.

- **Plano 0 y plano 138**

El primer y último plano representan el tanque del transformador a 0 V. Estos planos forman parte del límite de la modelación, por esta razón se mantienen constantes en todo momento al utilizar el método de las diferencias finitas.

El campo electrostático en estos planos de igual manera que el voltaje es cero. Las componentes X, Y del campo son nulas debido a que no existe variación de voltaje. Además, la componente Z del campo no se calcula por ser planos límite en la dirección Z.

- **Plano 1**

El plano 1 representa el aceite del transformador que se encuentra entre el tanque y la parte superior del núcleo.

La dirección del campo electrostático en esta zona se distribuye hacia el tanque y hacia la cara superior del núcleo, pasando de forma perpendicular las superficies equipotenciales. Figuras 4.34 y 4.35.

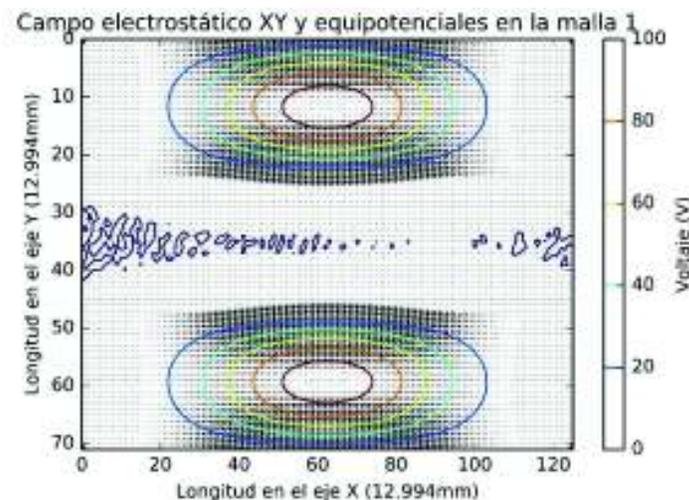


Figura 4.34. Campo electrostático y superficies equipotenciales en el plano 1 [Resultado obtenido por CEDIF].

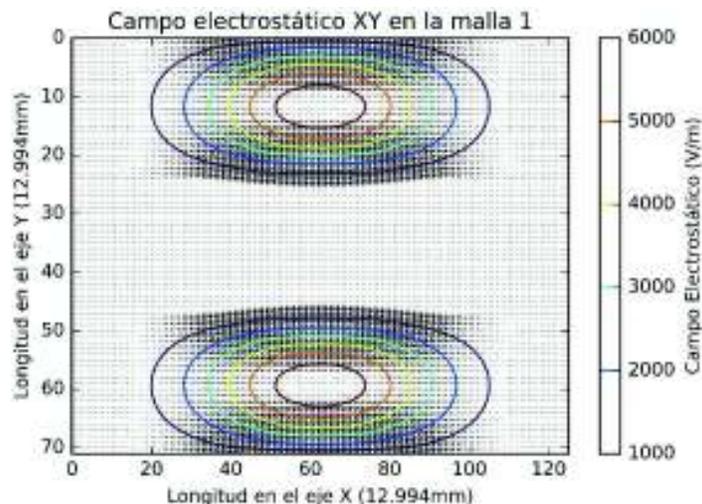


Figura 4.35. Campo electrostático en el plano 1 [Resultado obtenido por CEDIF].

La magnitud del campo electrostático en el plano 1 se ubica en dos zonas similares, las cuales forman circunferencias achatadas a los lados. Estas figuras aplastadas muestran su mayor magnitud en la zona central y la menor magnitud hacia los lados del tanque y hacia el lado más cercano al núcleo. Ver Figura 4.35.

- **Plano 15**

El plano 15 ejemplariza el corte transversal a la fila superior del núcleo. En la Figura 4.36 se modela la distribución del campo electrostático en esta zona juntamente con las superficies equipotenciales. Y de igual manera que el caso anterior el campo se dirige hacia el tanque y hacia la fila superior del núcleo.

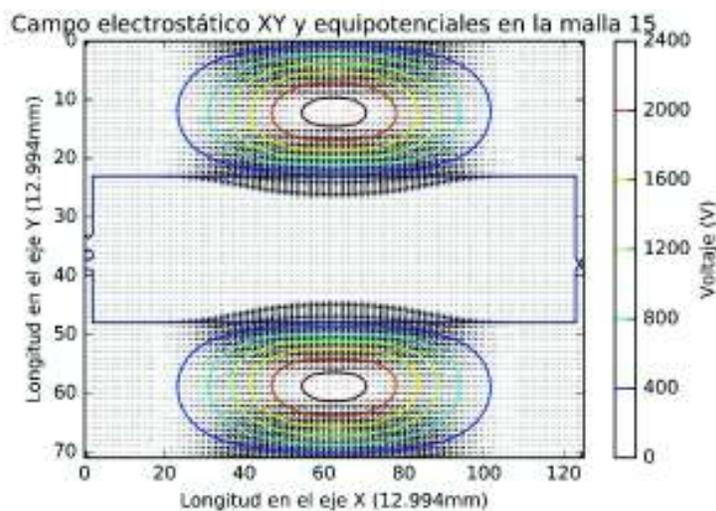


Figura 4.36. Campo electrostático y superficies equipotenciales en el plano 15 [Resultado obtenido por CEDIF].

La magnitud del campo electrostático se representa en la Figura 4.37, donde la mayor magnitud se da en la zona central junto al núcleo y va disminuyendo conforme se aleja de esta zona. Esto se debe a los devanados ya que están ubicados en la zona central inferior de la fila superior del núcleo.

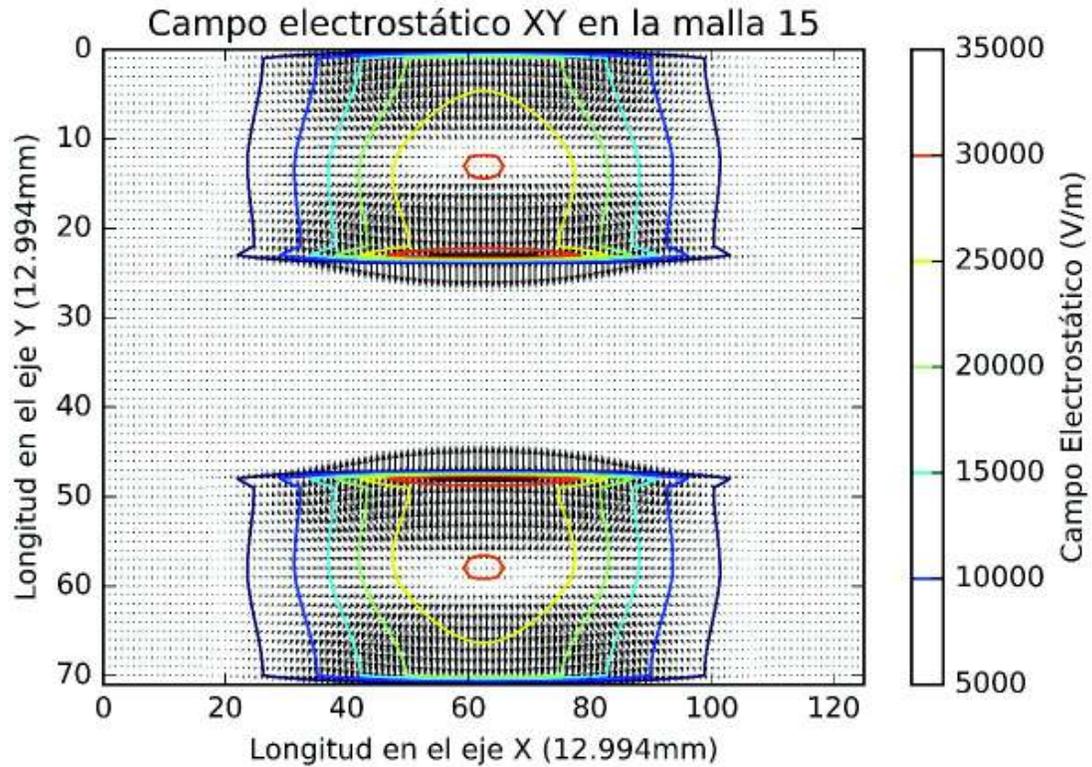


Figura 4.37. Campo electrostático en el plano 15 [Resultado obtenido por CEDIF].

- **Plano 31**

El plano 31 está ubicado entre la parte superior de los devanados y la parte inferior de la fila superior del núcleo.

Las líneas de campo se dirigen hacia los lados aterrizados del transformador, o sea hacia el tanque y hacia el núcleo. Ver Figura 4.38 y 4.39.

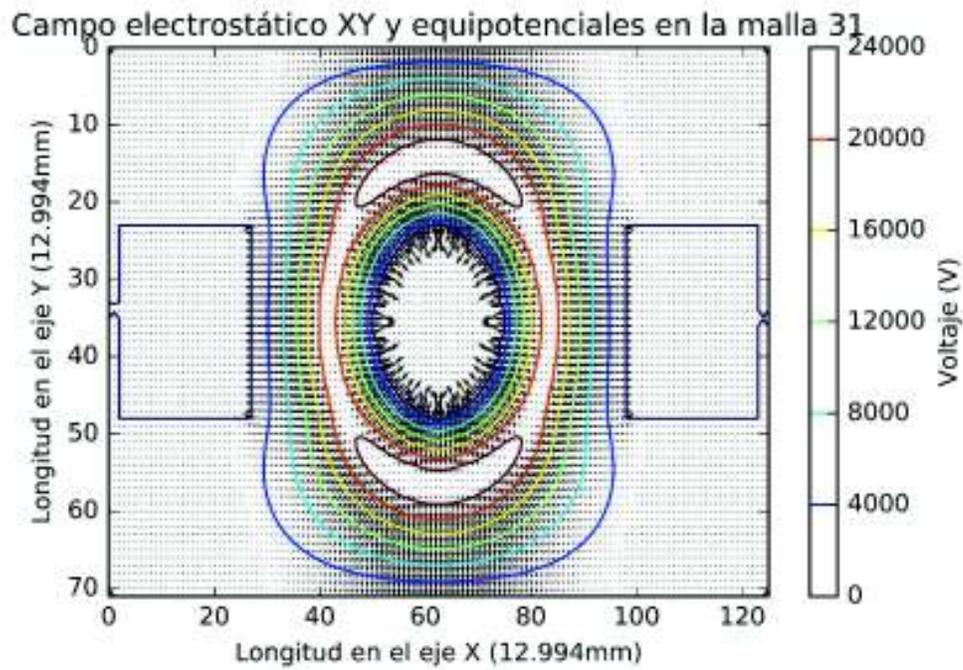


Figura 4.38. Campo electrostático y superficies equipotenciales en el plano 31 [Resultado obtenido por CEDIF].

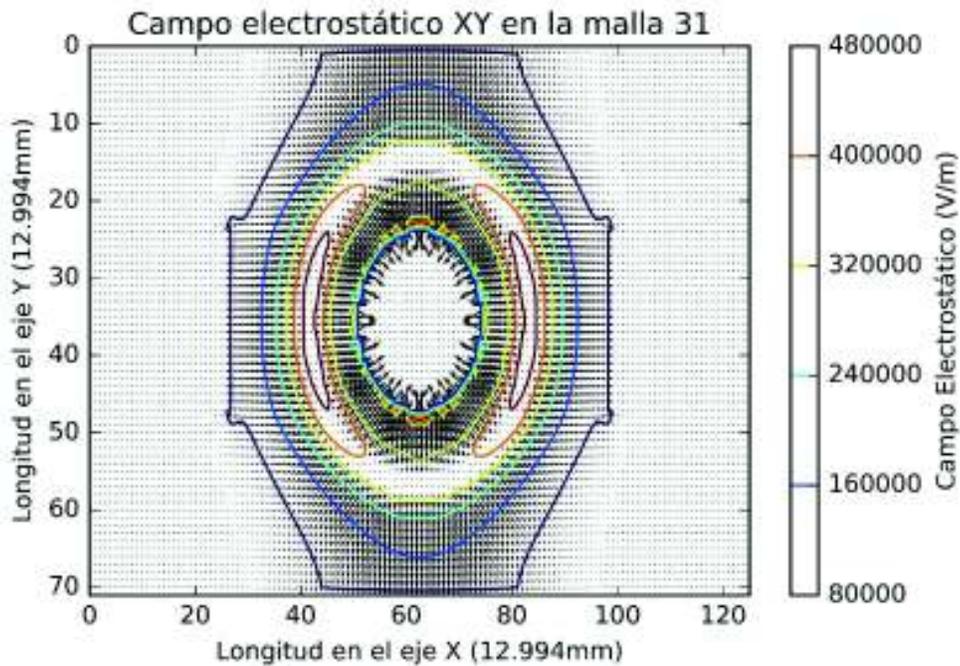


Figura 4.39. Campo electrostático en el plano 31 [Resultado obtenido por CEDIF].

La magnitud del campo electrostático aparece en mayor volumen por encima de los devanados, específicamente entre la fila superior del núcleo y el devanado de alto voltaje a 33 kVrms. Este comportamiento se debe a la diferencia de potencial entre el devanado de AV y el núcleo, además de la pequeña distancia que los separa. Figura 4.39.

- **Plano 69**

El plano 69 corta de forma transversal a los devanados del transformador, de esta manera se secciona el devanado de alto voltaje, de bajo voltaje y las tres columnas del núcleo.

La dirección del campo electrostático se analiza en dos regiones. La primera entre los devanados, donde el campo se dirige del devanado de AV hacia el de BV, haciendo referencia a que este se orienta desde los puntos de mayor potencial hacia los puntos de menor potencial. La segunda en la zona exterior del devanado de AV, ya que en esta región el campo se origina alrededor del devanado y se direcciona hacia las zonas aterrizadas. Figura 4.40.

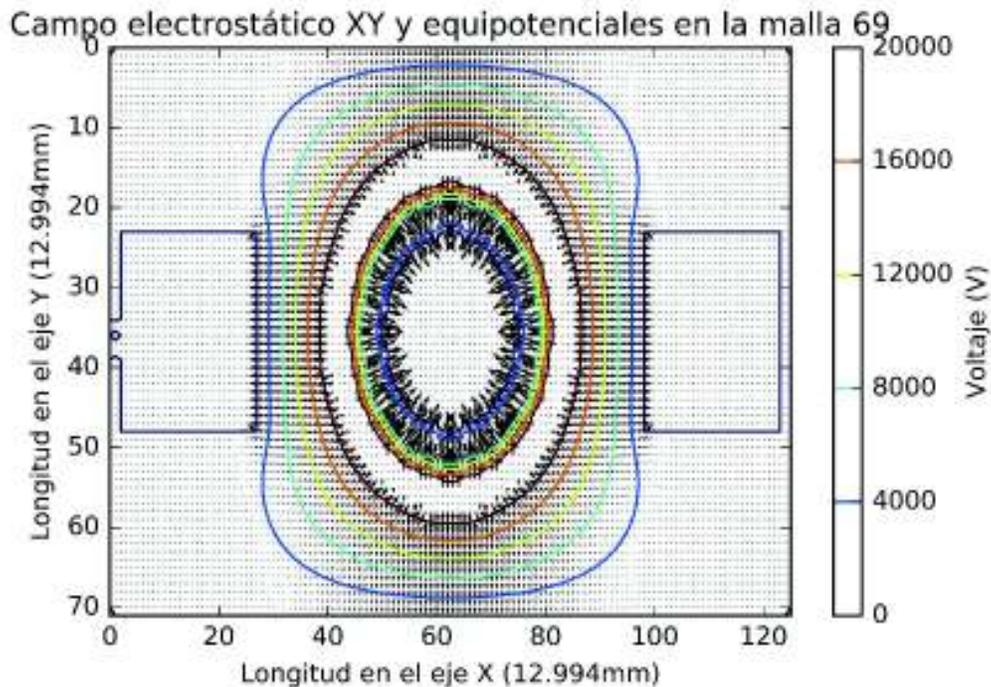


Figura 4.40. Campo electrostático y superficies equipotenciales en el plano 69 [Resultado obtenido por CEDIF].

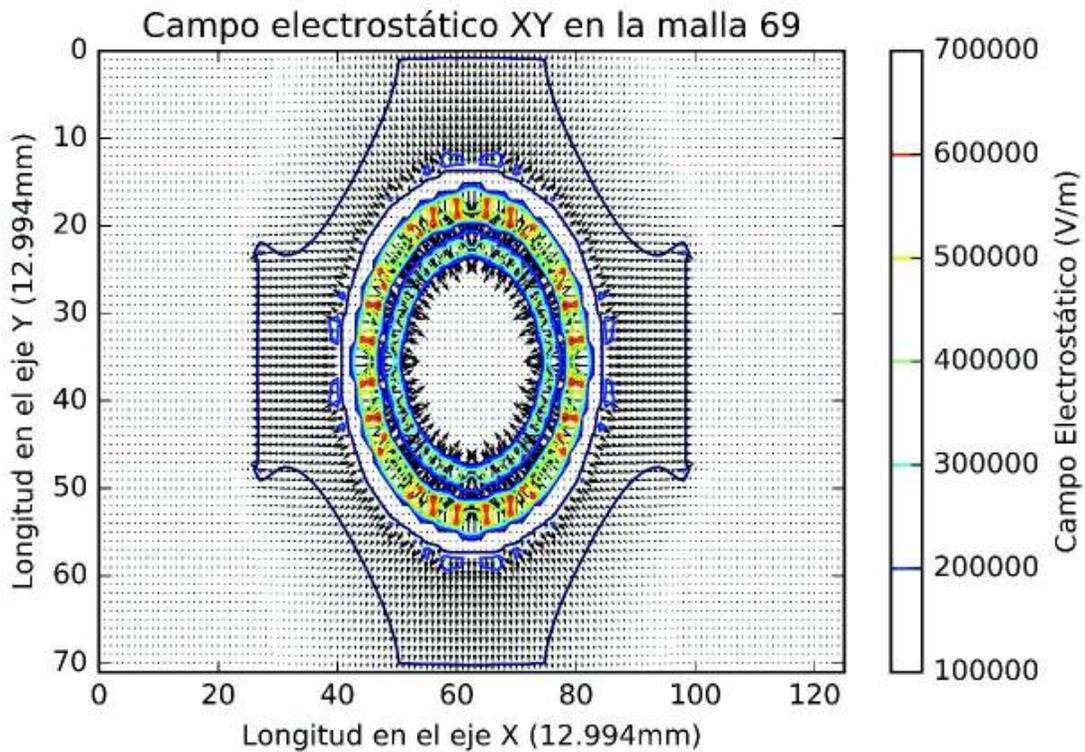


Figura 4.41. Campo electrostático en el plano 69 [Resultado obtenido por CEDIF].

El campo eléctrico tiene su máximo valor entre los devanados del transformador (Figura 4.41). Asimismo, el campo es cero en el interior de los devanados, del núcleo y en el borde del tanque. El tanque no recibe grandes cantidades de campo, debido a que estas magnitudes tienen sus valores más bajos en los límites con el tanque.

- **Plano 107**

El plano 107 es la transversal entre los devanados y la fila inferior del núcleo.

El campo se distribuye hacia el núcleo y hacia el tanque del transformador tal como se muestra en las Figuras 4.42 y 4.43.

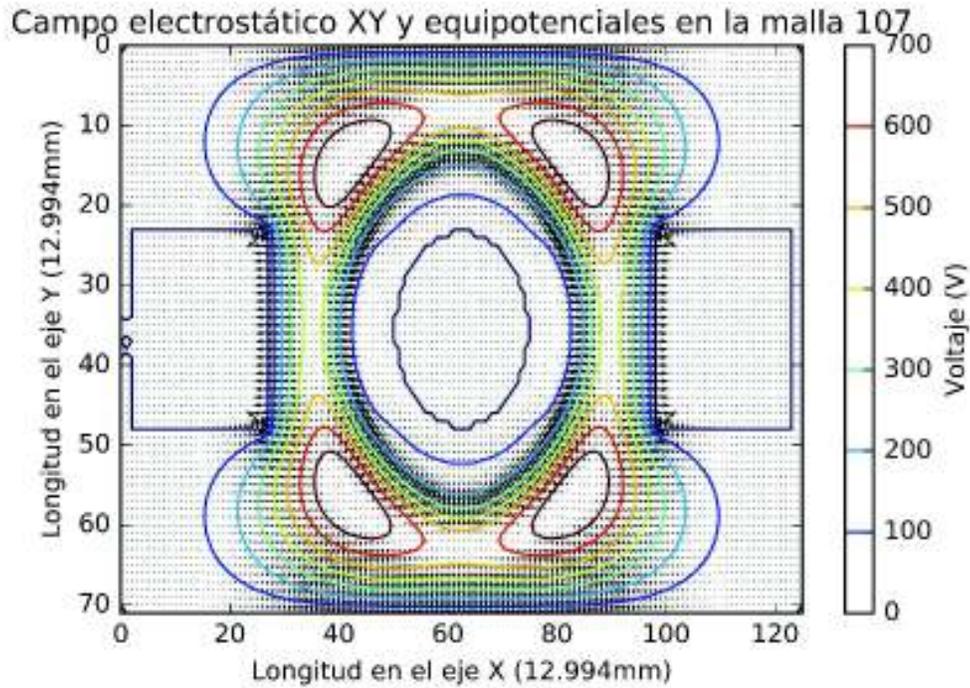


Figura 4.42. Campo electrostático y superficies equipotenciales en el plano 107 [Resultado obtenido por CEDIF].

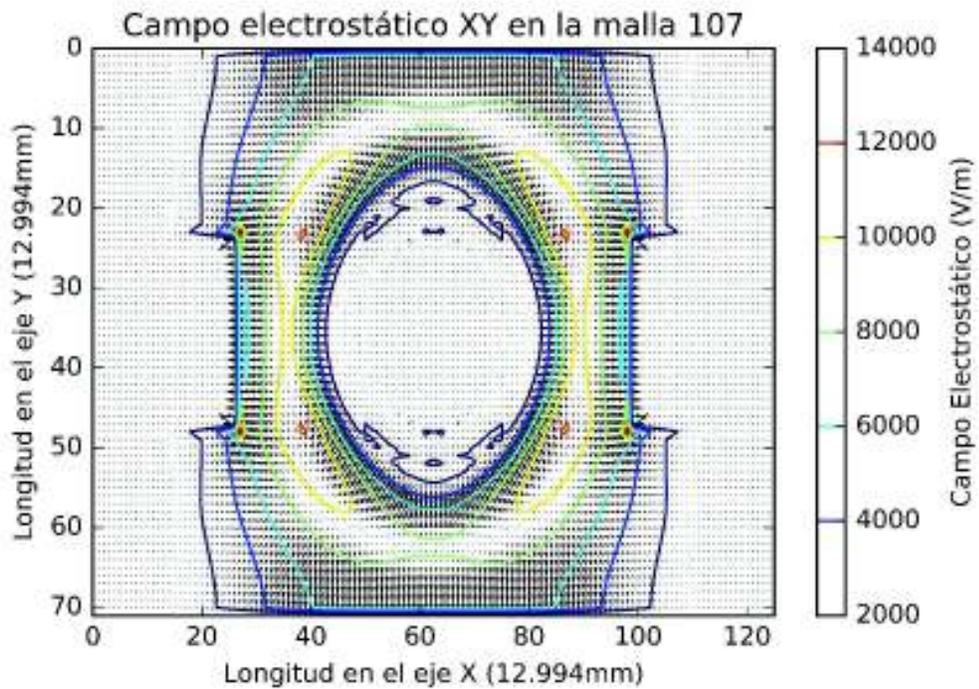


Figura 4.43. Campo electrostático en el plano 107 [Resultado obtenido por CEDIF].

La magnitud del campo en este plano cambia con respecto al plano 31 (transversal entre los devanados y la fila superior del núcleo). Esto se debe a que el devanado en la parte inferior posee valores bajos en comparación a los 33 kV en el devanado de AV y a los 11 kV en el devanado de BV. Por esta razón en campo parece ampliarse en el aceite. Ver Figura 4.43.

- **Plano 123**

Este plano está ubicado en la fila inferior del núcleo.

El campo de igual manera que en casos anteriores se orienta hacia las zonas aterrizadas. Ver Figuras 4.44 y 4.45.

El voltaje se sitúa de forma similar a ambos lados del núcleo formando superficies equipotenciales cerradas. Estas superficies se crean por la influencia de los voltajes de los devanados, específicamente de los bajos voltajes. Figuras 4.44.

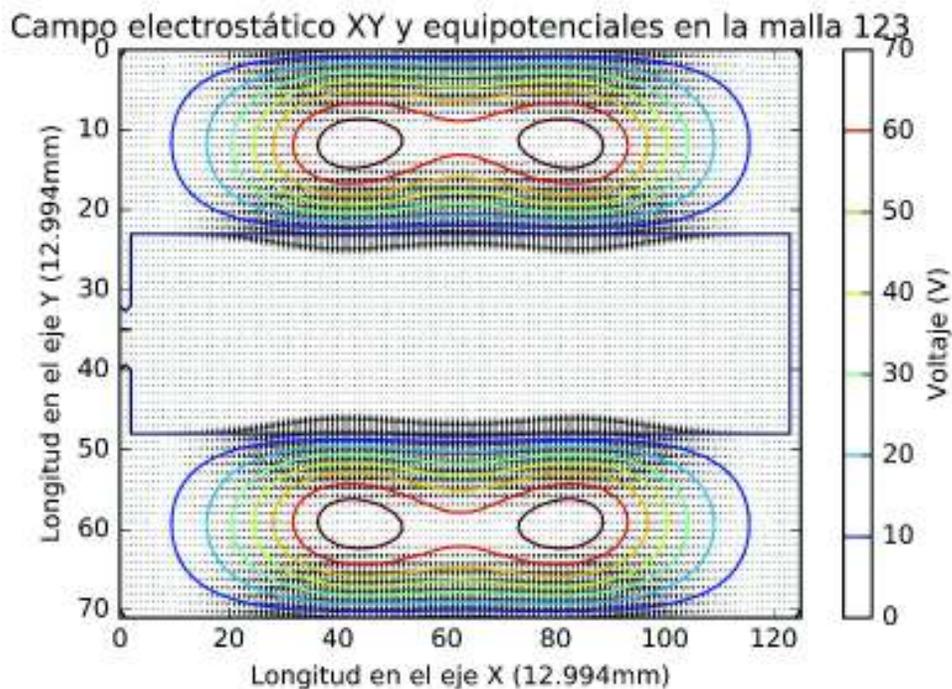


Figura 4.44. Campo electrostático y superficies equipotenciales en el plano 123
[Resultado obtenido por CEDIF].

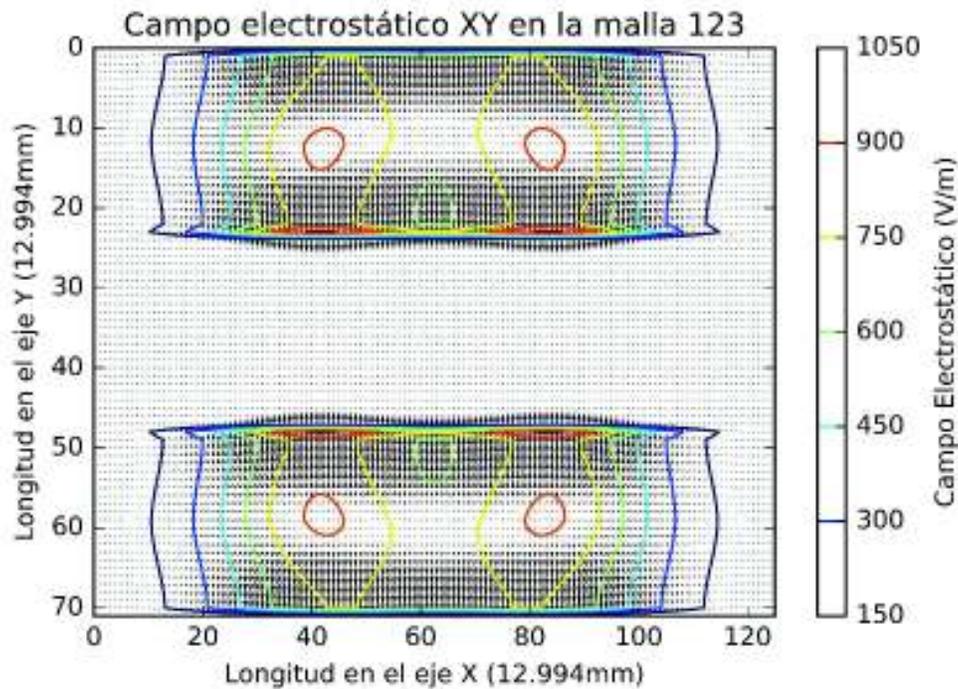


Figura 4.45. Campo electrostático en el plano 123 [Resultado obtenido por CEDIF].

La magnitud del campo en este plano tiene sus valores máximos cerca del núcleo debido a la cercanía que existe entre este punto y los devanados. De igual manera el campo se amplía en la mayoría del aceite con valores bajos en V/m.

- **Plano 137**

El plano 137 se encuentra entre la parte inferior de la fila inferior del núcleo y sobre el apoyo del tanque.

Este plano es similar al plano 1. Las diferencias radican en las magnitudes del voltaje y del campo electrostático. Además estos parámetros se alargan expandiéndose hacia los lados. Figuras 4.46 y 4.47.

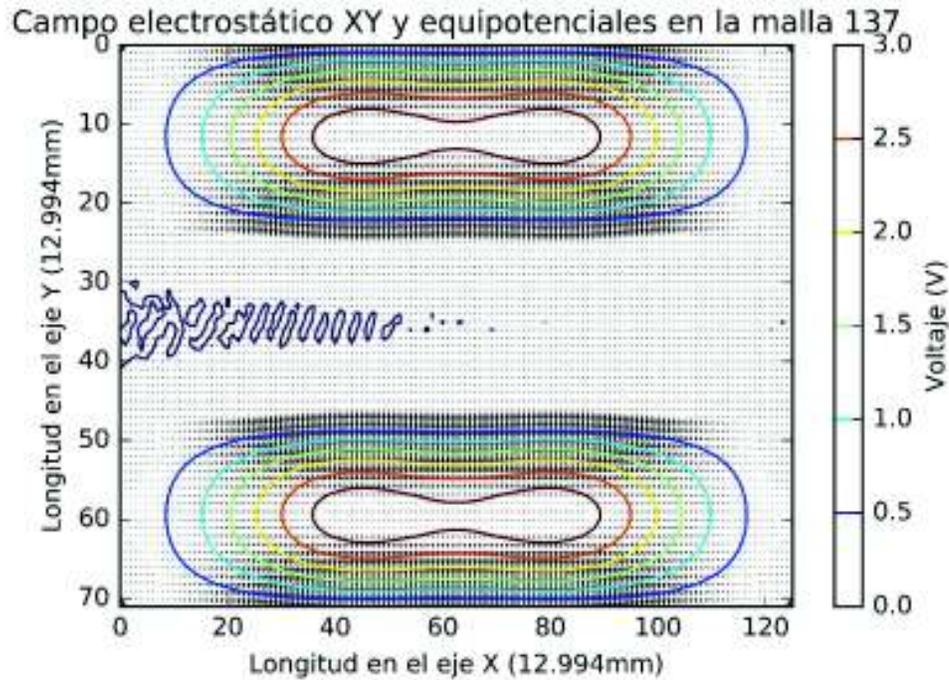


Figura 4.46. Campo electrostático y superficies equipotenciales en el plano 137 [Resultado obtenido por CEDIF].

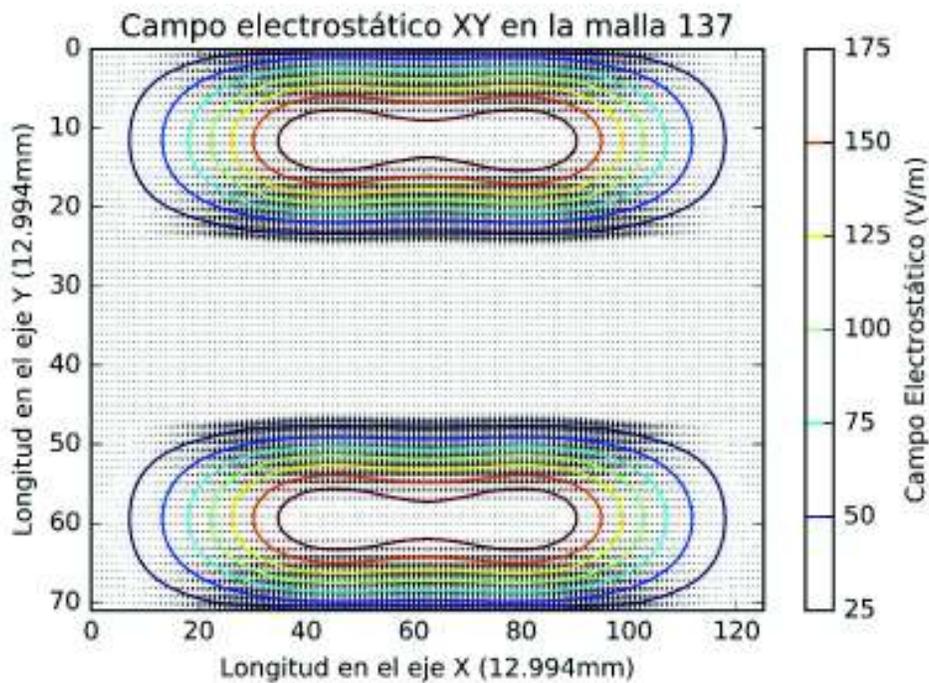


Figura 4.47. Campo electrostático en el plano 137 [Resultado obtenido por CEDIF].

5. CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

- Los sistemas de potencia presentan diferentes tipos de aislantes y conductores los cuales definen entornos no homogéneos. Por esta razón, es importante determinar una expresión parecida a la ecuación de Laplace que utilice las permitividades de cada material con el fin de establecer voltajes en entornos.
- El modelo matemático obtenido a partir de la ecuación de Laplace para campos electrostáticos responde de forma adecuada al obtener voltajes.
- CEDIF presenta ligeros sobresaltos al aplicar el gradiente de potencial para simular el comportamiento del campo electrostático, esto se debe a los errores por modelamiento y por discretización que se cometen al definir el problema.
- La facilidad de aplicar el método de las diferencias finitas en problemas de ingeniería favorece a su utilización computacional debido al ahorro de memoria causado por la sobre escritura de datos que se realiza en cada iteración.
- El método de las diferencias finitas resuelve sistemas de ecuaciones diferenciales en puntos de rejilla separados por una cierta distancia, si esta distancia es lo suficientemente pequeña la respuesta será más exacta, pero a su vez aumenta el número de datos, el número de cálculos y el tiempo de ejecución.
- Las rejillas y nodos que se utilizan en el método de las diferencias finitas crean errores por discretización, por redondeo y por modelamiento que a su vez dependen uno del otro. Por ejemplo, si aumenta el error por discretización disminuye el error por redondeo y es más probable que el modelamiento del entorno sea malo. El error por discretización es proporcional al error por modelamiento e inversamente proporcional al error de redondeo.
- El tiempo de ejecución del programa computacional depende directamente del número de datos y el valor de tolerancia que caracteriza el problema.
- El modelo utilizado cumple con las condiciones básicas de teoría electromagnética, es decir se verifica la dirección y la magnitud del campo electrostático.
- El campo electrostático posee mayor magnitud en la cercanía de los devanados dirigiéndose hacia las zonas aterrizadas, el núcleo y el tanque. El máximo valor de

campo que se obtiene está entre los devanados de alto y bajo voltaje. Además, el campo que llega al tanque es de baja magnitud, por tal razón este no se ve afectado en comparación a las partes del núcleo cercanas a los devanados.

- CEDIF que se desarrolla en lenguaje Python 3.5 obtiene una aproximación del comportamiento del campo electrostático para entornos homogéneos, no homogéneos, simétricos y asimétricos de dos y tres dimensiones mediante la utilización de diferencias finitas.

5.2 Recomendaciones

- El programa desarrollado en este trabajo podría ser extendido hacia su utilización en campos magnetostáticos.
- El algoritmo efectuado en este trabajo podría mejorarse utilizando programación orientada a objetos para la obtención de datos.
- Crear algoritmos en Python para modelar equipos eléctricos como transformadores, líneas de transmisión, cables, etc. De esta manera estos códigos de programa podrían conectarse directamente con CEDIF para obtener datos y no tener que importarlos.
- Como adición al método se podría incluir longitudes infinitesimales variables, lo que conllevaría a utilizar rejillas irregulares.
- Un futuro estudio podría llevarse a cabo aplicando el método de las diferencias finitas en el dominio del tiempo y así aplicarlo tanto en campos eléctricos como en campos electromagnéticos.

6. REFERENCIAS BIBLIOGRÁFICAS

- [1] E. Mendoza, "ANÁLISIS DEL SABER", Universidad Fermín Toro, Venezuela, Editorial: MENDOZA, URL: <https://es.slideshare.net/ErnestoMendoza10/anlisis-numrico-74245817>
- [2] "Finite Element Method Magnetics", 6 Abril 2014, URL: <http://www.femm.info/wiki/HomePage>
- [3] Direct INDUSTRY, "ANSYS", 2017, URL: <http://www.directindustry.es/prod/ansys/product-9123-504524.html>
- [4] A. Sánchez, R. Leoviglido, "Métodos Numéricos", Ingeniería en Software, 9 Diciembre 2016, URL: https://issuu.com/argenissan/docs/metodos_numericos._u_3_y_4.docx
- [5] J. Edminister, "Teoría y problemas de electromagnetismo", Serie de compendios Schaum, Bogotá: Editorial McGRAW-HILL Latinoamericana s.a., 1981.
- [6] W. Hayt y J. Buck, "Teoría electromagnética", Séptima ed., México, México: McGraw-Hill Interamericana, 2006, pp. 1-209.
- [7] A. Carrillo y O. Mendoza, "Introducción al Método de Diferencias Finitas y su Implementación Computacional", UNAM, Facultad de Ciencias, 2015.
- [8] G. Hernández, "El método de diferencias finitas", URL: http://mmc2.geofisica.unam.mx/cursos/hidrogeologia/NotasCurso/1-MDF1_1-10.pdf
- [9] Departamento de Matemáticas, "Métodos Iterativos para Resolver Sistemas Lineales", 17 Julio 2009, URL: <http://www.mty.itesm.mx/dmti/materias/ma2008/lecturas/ma2008-09a.pdf>
- [10] J. De la Fuente, "Sistemas de ecuaciones lineales - Métodos iterativos de resolución", Universidad Politécnica de Madrid, Escuela Técnica Superior de Ingenieros Industriales, 2016, URL: http://www.jldelafuenteoconnor.es/Clase_itera_2016.pdf
- [11] D. López, "MÉTODO SOR", Universidad de Antioquia, Aprende en línea - Plataforma académica para pregrado y posgrado, 30 Abril 2016. URL: <http://aprendeonline.udea.edu.co/lms/moodle/mod/page/view.php?id=24492>
- [12] A. Arenas, M. Murcia y O. Quijano, "Método Gauss-Seidel por relajación (SOR)", 2014, URL: <https://prezi.com/rcychknkgod4/metodo-gauss-seidel-por-relajacion-sor/>
- [13] Ángels, "Cuál es la diferencia entre círculo y circunferencia", UNCÓMO, URL: <https://educacion.uncomo.com/articulo/cual-es-la-diferencia-entre-circulo-y-circunferencia-23126.html>
- [14] R. Armas, "Estudio del comportamiento del campo eléctrico en cables subterráneos con aislamiento distribuido en capas de diferente permitividad", Escuela Politécnica Nacional, Quito, Pichincha, 2017.
- [15] I. Dasgupta, "Design of Transformers", New Delhi: McGraw-Hill, 2002.

7. ANEXOS

ANEXO I

USO DE CEDIF

Este anexo explicará cómo se realiza el ingreso de información y datos en la herramienta de software CEDIF.

A. VERIFICAR LA UBICACIÓN DE LA HERRAMIENTA DE SOFTWARE

CEDIF se compone de dos documentos. El primero es un documento en Excel donde se ingresa la información y los datos del problema electrostático. El segundo es un algoritmo en Python 3.5 que se encarga de la obtención de datos e información del Excel. De esta manera la herramienta realiza cálculos y devuelve respuestas del comportamiento del campo electrostático de forma gráfica.

Estos dos archivos deben ser ubicados en una misma carpeta del ordenador (Figura I.1) para garantizar el correcto funcionamiento de CEDIF. También esta carpeta almacenará las respuestas gráficas del problema analizado.

Nombre del documento Excel : Datos del campo electrostático

Nombre del algoritmo en Python 3.5: Campo_electrostático_2D_y_3D



Figura I.1 Documentos que conforma CEDIF [Elaboración propia].

B. INGRESO DE INFORMACIÓN EN EL DOCUMENTO EXCEL

El documento Excel presenta la siguiente plantilla. Figura I.2.

ESCUELA POLITÉCNICA NACIONAL

DETERMINACIÓN DEL COMPORTAMIENTO DEL CAMPO ELECTROSTÁTICO EN ELECTRODOS DE CONFIGURACIÓN COMPLEJA PARA AMBIENTES BIDIMENSIONALES Y TRIDIMENSIONALES

DATOS REALES:

Número de dimensiones:

Unidad de longitud:

Longitud en X:

Longitud en Y:

Longitud en Z:

Longitud infinitesimal:

Tolerancia:

Parámetro de relajación:

APROXIMACIÓN DE LAS LONGITUDES:

Longitud en X:

Longitud en Y:

Longitud en Z:

DIMENSIONES:

Malla individual:

Malla total:

LÍMITES PARA INGRESAR DATOS A LAS MALLAS:

Malla individual:

Malla total:

2 DIMENSIONES

3 DIMENSIONES

CONDICIONES:

- Los elementos de la "Matriz de voltajes" pueden tomar cualquier valor.
- Todos los elementos de la "Matriz de voltajes" que formen parte del contorno, son consideradas condiciones de frontera (siempre constantes en las iteraciones).
- Si un elemento de la "Matriz de voltajes" tiene un valor diferente de cero, entonces ese valor será siempre constante en las iteraciones.
- Si un elemento de la "Matriz de voltajes" es cero y se desea que ese elemento nunca cambie en las iteraciones, entonces el elemento de la "Matriz de condiciones frontera" con la misma ubicación, debe ser cero.

Figura I.2 Plantilla del documento Excel [Impresión de pantalla de Excel].

En esta plantilla se ingresa el número de dimensiones, la unidad de longitud, la longitud en X, Y, Z, la longitud infinitesimal, tolerancia y parámetro de relajación.

Como ejemplo se mostrará el problema simétrico bidimensional de teoría electromagnética para un enrejado de 16x16 (ver capítulo 3). En la Figura I.3 se muestra la plantilla de la hoja "Datos iniciales" con los datos correspondientes al ejemplo planteado

DATOS REALES:		DIMENSIONES:	
Número de dimensiones:	2	Malla individual:	17 x 17
Unidad de longitud:	cm	Malla total:	17 x 17
Longitud en X:	4		
Longitud en Y:	4		
NO INGRESAR			
Longitud infinitesimal:	0,25		
Tolerancia:	0,0001		
Parámetro de relajación:	1,9		
APROXIMACIÓN DE LAS LONGITUDES:		LÍMITES PARA INGRESAR DATOS A LAS MALLAS:	
Longitud en X:	4	Malla individual:	Desde A1 hasta Q17
Longitud en Y:	4	Malla total:	Desde A1 hasta Q17
Longitud en Z:	0		

Figura I.3 Información del problema electrostático [Impresión de pantalla de Excel].

C. INGRESO DE DATOS EN EL DOCUMENTO EXCEL

El documento Excel posee 4 hojas. En la primera se ingresa la información del problema electrostático, en la segunda los valores de voltaje, en la tercera las condiciones frontera y en la cuarta las permitividades.

Nombre de la primera hoja: Datos iniciales

Nombre de la segunda hoja: Matriz de voltajes

Nombre de la tercera hoja: Matriz de condiciones frontera

Nombre de la cuarta hoja: Matriz de permitividades

La sección “*Límites para ingresar datos a las mallas*” ubicada en la hoja “*Datos iniciales*” devuelve las celdas donde se debe ingresar los datos de voltaje, condiciones de frontera y permitividades en las 3 últimas hojas de Excel. Todo esto después de haber ingresado la información del problema.

Las hojas de Excel serían llanadas como se muestra en las Figuras I.4, I.5 e I.6.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figura I.4 Datos que se ingresan en la “*Matriz de voltajes*” para el problema simétrico bidimensional de teoría electromagnética [Impresión de pantalla de Excel].

Los datos ingresados en la “*Matriz de voltajes*” y en la “*Matriz de condiciones frontera*” deben cumplir con las condiciones especificadas en la hoja “*Datos iniciales*”.

Condiciones especificadas en la hoja “*Datos iniciales*”:

- Los elementos de la “*Matriz de voltajes*” pueden tomar cualquier valor.
- Todos los elementos de la “*Matriz de voltajes*” que formen parte del contorno, son considerados condiciones de frontera (siempre constantes en las iteraciones)
- Si un elemento de la “*Matriz de voltajes*” tiene un valor diferente de cero, entonces ese valor será siempre constante en las iteraciones.
- Si un elemento de la “*Matriz de voltajes*” es cero y se desea que ese elemento nunca cambie en las iteraciones, entonces el elemento de la “*Matriz de condiciones frontera*” con la misma ubicación debe ser cero.

La “*Matriz de condiciones frontera*” solo puede contener valores de cero o uno según se especifica a continuación:

- Cero: Asigna a un elemento de la matriz como condición de frontera, es decir nunca variará en las iteraciones.
- Uno: Asigna a un elemento de la matriz como punto de análisis, es decir variará en cada iteración hasta cumplir la tolerancia.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
3	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
4	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
5	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
6	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
7	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
8	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
9	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
10	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
11	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
12	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
13	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
14	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
15	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
16	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figura I.5 Datos que se ingresan en la “*Matriz de condiciones frontera*” para el problema simétrico bidimensional de teoría electromagnética [Impresión de pantalla de Excel].

La “*Matriz de permitividades*” puede contener cualquier valor a excepción del cero.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
5	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
11	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
12	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
13	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
14	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
15	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
16	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
17	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

Figura I.6 Datos que se ingresan en “*Matriz de permitividades*” para el problema simétrico bidimensional de teoría electromagnética [Impresión de pantalla de Excel].

Para finalizar el proceso de ingreso de datos, el documento Excel debe ser guardado.

D. EJECUCIÓN DEL ALGORITMO

El algoritmo en Python 3.5 debe ser ejecutado de la siguiente manera:

- Clic en la parte superior izquierda de Python donde se encuentra la opción abrir archivo (Figura I.7)



Figura I.7 Opción abrir archivo [Impresión de pantalla de Python].

- En la ventana que se desplegó buscar la dirección del documento Python llamado “*Campo_electrostático_2D_y_3D*” (misma carpeta donde se ubica en documento Excel). Seleccionar el documento y dar clic en abrir.

La pantalla de Python en la parte superior tiene la opción Ejecutar archivo (Figura I.8), la cual es la instrucción final para empezar el proceso de cálculos y obtención de resultados.



Figura I.8 Opción Ejecutar archivo [Impresión de pantalla de Python].

E. EJECUCIÓN DE LA HERRAMIENTA DE SOFTWARE

CEDIF una vez que empezó la ejecución devuelve las características principales del problema en forma de membrete tal como se muestra en la Figura I.9.

```
Datos iniciales      :  
Dimensiones         : 2  
Dimensión en X      : 4 cm  
Dimensión en Y      : 4 cm  
Dimensión en Z      : 0 cm  
Longitud infinitesimal : 0.25 cm  
Tolerancia          : 0.0001  
Parámetro de relajación: 1.9
```

Figura I.9 Membrete del problema [Impresión de pantalla de Python].

El algoritmo muestra un mensaje para observar de forma gráfica los valores de voltaje ingresados en Excel después de observar el membrete del problema. Todo esto con el fin de comprobar la modelación del entorno. Figura I.10.

```
Desea observar los gráficos del voltaje ingresados en Excel (1-SI):
```

Figura I.10 Mensaje mostrado por el algoritmo para observar los gráficos ingresados en Excel [Impresión de pantalla de Python].

Ingresar el número uno si se desea observar dichos gráficos, en caso contrario el cero. Enseguida se desplegarán el gráfico o gráficos correspondientes dependiendo si es caso bidimensional o tridimensional (Figura I.11).



Figura I.11 Gráfico de los valores de voltaje ingresados en Excel para el modelamiento del problema simétrico de teoría electromagnética [Resultado obtenido por CEDIF].

CEDIF vuelve a exponer tres instrucciones que son seleccionadas por el usuario y estas son:

- Desea observar los gráficos de voltajes equipotenciales (1-SI):
- Desea observar los gráficos del campo electrostático en las superficies equipotenciales (1-SI):
- Desea observar los gráficos del campo electrostático (1-SI):

Las siguientes estipulaciones deben tenerse en cuenta para las instrucciones anteriores:

- Cada instrucción presenta gráficos correspondientes a su orden específica.
- Si el usuario selecciona un valor distinto de uno en las instrucciones, estas devuelven un mensaje como: “No se mostrarán los gráficos”, saltándose de forma instantánea a la siguiente instrucción.

- Las instrucciones que se efectuaron guardarán los resultados gráficos en la misma carpeta donde se ubicaron el documento Excel y el documento de Python.
- El algoritmo mide el tiempo de ejecución de todo el problema.

El presente anexo mostró la manera de ingresar la información a la herramienta de software CEDIF para simular y obtener resultados del comportamiento del campo electrostático en entornos bidimensionales y tridimensionales.

ANEXO II

CÓDIGO DE PROGRAMA EN PYTHON 3.5

```
# Módulos y librerías
from time import time
import openpyxl
import numpy as np
import matplotlib.pyplot as plt
import sys

# Obtención de datos desde Excel
# El documento debe ser llamado "Hoja_de_datos_del_campo_electrostático"
# "Excel", debe contener 4 hojas: Datos iniciales, Matriz de voltajes,....
# Matriz de condiciones frontera, Matriz de permitividades
# Iniciación del cronómetro
tiempo_inicial = time()
doc = openpyxl.load_workbook('Datos del campo electrostático.xlsx')
hoja1 = doc.get_sheet_by_name('Datos iniciales')
hoja2 = doc.get_sheet_by_name('Matriz de voltajes')
hoja3 = doc.get_sheet_by_name('Matriz de condiciones frontera')
hoja4 = doc.get_sheet_by_name('Matriz de permitividades')

# Glosario
# Malla.- plano horizontal en 2D
# Cubo .- conjunto de mallas horizontales en 2D ubicadas una bajo la otra

# Constantes:
# dim = número de dimensiones
# lon = unidad de longitud de la longitud infinitesimal (mm, cm, dm, m)
# longX = longitud de X
# longY = longitud de Y
# longZ = longitud de Z
# li = longitud infinitesimal
# rel = parámetro de relajación
# tol = tolerancia

dim = hoja1['C7'].value
lon = hoja1['C8'].value
longX = hoja1['C9'].value
longY = hoja1['C10'].value

if(dim==2):
    longZ=0
else:
    longZ = hoja1['C11'].value

li = hoja1['C12'].value
tol = hoja1['C13'].value
rel = hoja1['C14'].value

#-----
```

```

# Función para insertar títulos, datos y generar matrices
#-----
def titulos_datos(dim, longX, longY, longZ, li, tol, lon, rel):
    # Retornos:
    # X = número de puntos en X
    # Y = número de puntos en Y
    # Z = número de puntos en Z
    # v = matriz de voltajes
    # maco = matriz de condiciones frontera
    # per = matriz de permitividades
    # vi = matriz de voltajes iniciales (ingresados por el usuario en Excel)

    # Membrete del Problema
    print("Datos iniciales      :")
    print("Dimensiones           :", dim)
    print("Dimensión en X         :", longX, lon)
    print("Dimensión en Y         :", longY, lon)
    print("Dimensión en Z         :", longZ, lon)
    print("Longitud infinitesimal :", li, lon)
    print("Tolerancia              :", tol)
    print("Parámetro de relajación:", rel)

    # Creación de rejillas
    X = int(np.ceil(longX/li) + 1)
    Y = int(np.ceil(longY/li) + 1)
    Z = int(np.ceil(longZ/li) + 1)

    # Inicialización de matrices
    v = np.zeros((Z, Y, X))
    maco = np.ones((Z, Y, X))
    per = np.ones((Z, Y, X))
    vi = np.zeros((Z, Y, X))
    E = np.zeros((Z, Y, X))
    return X, Y, Z, v, maco, per, vi, E

#-----
# Función para graficar
#-----
def graficar(v, Z, Y, X, li, lon, camp, Ex, Ey, inst, tit, exp, no_inst,
            med_esc):
    # Argumentos:
    # v = matriz de voltajes
    # X, Y, Z = dimensiones de la matriz de voltajes en cada eje
    # li = longitud infinitesimal
    # camp = condición para graficar el Campo Eléctrico (1-si 0-no)
    # Ex = componentes en X del campo eléctrico
    # Ey = componentes en Y del campo eléctrico
    # inst = mensaje de la instrucción a ejecutar
    # tit = título de los gráfico que se muestra en pantalla
    # exp = título de los gráficos que se exportan
    # no_inst = mensaje de la operación no ejecutada
    # med_esc = unidad de medida de la escala

    # Orden mostrada para realizar un gráfico

```

```

graf= input(inst)

if(graf == str(1)):
    # escala de longitud para los gráficos
    escala = str(li) + lon
    # Barrido de todos los planos (eje Z)
    for k in range(int(Z)):
        # Creación de gráficos
        plt.figure(k)

        if(camp == 1):
            # Creación de las líneas de campo
            plt.quiver(Ex[k, :, :], -Ey[k, :, :])
            plt.hold(True)
            # Apunta al primer elemento de la matriz
            ant = v[k,0,0]
            # Barrido de todas las filas y columnas de cada plano (X, Y)
            for j in range(int(Y)):
                for i in range(int(X)):
                    # Apunta al elemento actual de la matriz
                    act = v[k,j,i]

                    if(ant == act):
                        ant = act
                        # Ingresa si todos los elementos de un plano...
                        # son iguales
                        if((j == int(Y-1)) and (i == int(X-1))):
                            # Diferencia los niveles por colores
                            CS = plt.contourf(v[k, :, :])
                        else:
                            # Grafica los niveles del gráfico por contornos
                            CS = plt.contour(v[k, :, :])
                            # Paso al siguiente plano
                            break
                    if(ant != act):
                        break

            else:
                plt.grid(True)
                CS = plt.contourf(v[k, :, :])

        # Crea una barra vertical que describe los niveles de la escala
        c = plt.colorbar(CS)
        # Ubica el título de los niveles de la escala
        c.set_label('med_esc')
        # Ubica los títulos de ejes y del gráfico
        plt.ylabel('Longitud en el eje Y (%s)'%(escala))
        plt.xlabel('Longitud en el eje X (%s)'%(escala))
        plt.title('tit %'(k))
        # Invierte el eje y
        plt.gca().invert_yaxis()
        # Exporta los gráficos
        plt.savefig('exp %'(k), format='jpg', figsize = (20,20),dpi = 800)
        # Muestra los gráficos

```

```

        plt.show()
    else:
        print(no_inst)
    return

#-----
# Función Gauss – Seidel con parámetro de relajación
#-----
def gauss_seidel_rel(v, maco, per, tol, dim, X, Y, iniZ, finZ, cte, rel):
    # Argumentos:
    # v = matriz de voltajes antes de una iteración seidel
    # maco = matriz de condiciones frontera
    # per = matriz de permitividades
    # tol = tolerancia
    # dim = número de dimensiones
    # X, Y = dimensiones de la matriz de voltajes en cada eje
    # iniZ = valor inicial de k en el bucle "for"
    # finZ = valor final de k en el bucle "for"
    # cte = número de puntos vecinos al punto analizado
    # Variables internas:
    # v_ant = voltaje en un punto antes de la iteración Gauss-Seidel
    # v_des = voltaje en un punto después de la iteración Gauss - Seidel
    # Retornos:
    # v = matriz de voltajes después de una iteración seidel
    # no_tol = número de elementos de la matriz de voltaje que no cumplen....
    #     la tolerancia

    # Encero la variable para una nueva iteración seidel
    no_tol = 0

    # Barrido de todos los elementos de la matriz 3D a excepción de los bordes
    for k in range(iniZ, finZ + 1):
        for j in range(1, int(Y) - 1):
            for i in range(1, int(X) - 1):

                if(maco[k,j,i]==1):
                    # Análisis debido al número de dimensiones
                    if(dim == 2):
                        ter_adi = 0
                    elif(dim == 3):
                        ter_adi = ((v[k-1,j,i] + v[k+1,j,i])*4*per[k,j,i] +
                                    (v[k+1,j,i] - v[k-1,j,i])*(per[k+1,j,i] -
                                                            per[k-1,j,i]))

                v_ant = v[k, j, i]

                # Iteración Gauss - Seidel
                v[k,j,i] = ((rel / cte) * (v[k,j,i+1] + v[k,j,i-1] +
                                            v[k,j+1,i] + v[k,j-1,i] +
                                            ((per[k,j,i+1] - per[k,j,i-1])*
                                             (v[k,j,i+1] - v[k,j,i-1]) +
                                             (per[k,j+1,i] - per[k,j-1,i])*
                                             (v[k,j+1,i] - v[k,j-1,i]) +
                                             ter_adi) / (4*per[k,j,i])) +

```

```

        (1-rel)*v[k,j,i])

    v_des = v[k,j,i]

    # Análisis de tolerancia en cada punto
    if(abs(v_des - v_ant) >= tol):
        no_tol = no_tol + 1
    return v, no_tol

#-----
# Función del campo eléctrico
#-----
def campo(Z, Y, X, v, li, tol, lon):
    # Argumentos:
    # X, Y, Z = dimensiones de la matriz de voltajes en cada eje
    # v = matriz de voltaje
    # li = longitud infinitesimal
    # tol = tolerancia
    # lon = unidad de longitud de la longitud infinitesimal
    # Variables internas:
    # lmi = longitud en metros de la longitud infinitesimal
    # num1i = voltaje del punto a la izquierda del punto analizado
    # num2i = voltaje del punto a la derecha del punto analizado
    # num1j = voltaje del punto de al frente del punto analizado
    # num2j = voltaje del punto de atrás del punto analizado
    # num1k = voltaje del punto que se encuentra debajo del punto analizado
    # num2k = voltaje del punto que se encuentra encima del punto analizado
    # mayori = voltaje entre los puntos vecinos en el eje X
    # mayorj = voltaje entre los puntos vecinos en el eje Y
    # mayork = voltaje entre los puntos vecinos en el eje Z
    # sigi = dirección del campo en el eje X
    # sigj = dirección del campo en el eje Y
    # sigk = dirección del campo en el eje Z
    # Retornos:
    # Ex, Ey, Ez = matrices que contienen las componentes del campo eléctrico
    # E = campo electrostático en cada punto de la malla

    # Asignación de la unidad de longitud
    if(lon==str("mm")):
        lon=0.001
    elif(lon==str("cm")):
        lon=0.01
    elif(lon==str("dm")):
        lon=0.1
    elif(lon==str("m")):
        lon=1

    lmi = li * lon

    # Inicialización de las matrices del campo eléctrico
    Ex = np.zeros((Z, Y, X))
    Ey = np.zeros((Z, Y, X))
    Ez = np.zeros((Z, Y, X))

```

```

# Barrido de todos los elementos de la matriz de voltaje que puedan...
# contribuir a la componente X del campo electrostático
for k in range(int(Z)):
    for j in range(int(Y)):
        for i in range(1, int(X) - 1):

            # Designación de voltajes vecinos en el eje X
            num1i = v[k, j, i - 1]
            num2i = v[k, j, i + 1]
            # Ingresa si los voltajes vecinos en el eje X son iguales,...
            # es decir el punto analizado se encontraría en una región...
            # equipotencial
            if(abs(num1i - num2i) < tol):
                mayori = v[k, j, i]
                sigi = 1
            else:
                # Ingresa si el punto analizado se encuentra en el...
                # límite derecho de una región equipotencial (en la...
                # dirección de X)
                if(num1i == v[k, j, i]):
                    mayori = num2i
                    sigi = -1
                # Ingresa si el punto analizado se encuentra en el...
                # límite izquierdo de una región equipotencial (en la...
                # dirección de X)
                elif(num2i == v[k, j, i]):
                    mayori = num1i
                    sigi = 1
                # Ingresa si el punto analizado se encuentra en una...
                # región de voltaje variante (en la dirección de X)
                elif(num1i > num2i):
                    mayori = num1i
                    sigi = 1
                else:
                    mayori = num2i
                    sigi = -1

```

$$Ex[k, j, i] = sigi * (mayori - v[k, j, i]) / lmi$$

```

# Barrido de todos los elementos de la matriz de voltaje que puedan...
# contribuir a la componente Z del campo electrostático
for j in range(int(Y)):
    for i in range(int(X)):
        for k in range(1, int(Z) - 1):

            # Designación de voltajes vecinos en el eje Z
            num1k = v[k - 1, j, i]
            num2k = v[k + 1, j, i]
            # Ingresa si los voltajes vecinos en el eje Z son iguales,...
            # es decir el punto analizado se encontraría en una región...
            # equipotencial
            if(abs(num1k - num2k) < tol):
                mayork = v[k, j, i]
                sigk = 1

```

```

else:
    # Ingresa si el punto analizado se encuentra en el...
    # límite inferior de una región equipotencial (en la...
    # dirección de Z)
    if(num1k == v[k, j, i]):
        mayork = num2k
        sigk = -1
    # Ingresa si el punto analizado se encuentra en el...
    # límite superior de una región equipotencial (en la...
    # dirección de Z)
    elif(num2k == v[k, j, i]):
        mayork = num1k
        sigk = 1
    # Ingresa si el punto analizado se encuentra en una...
    # región de voltaje variante (En la dirección de Z)
    elif(num1k > num2k):
        mayork = num1k
        sigk = 1
    else:
        mayork = num2k
        sigk = -1

```

$$Ez[k, j, i] = sigk * (mayork - v[k, j, i]) / lmi$$

```

# Barrido de todos los elementos de la matriz de voltaje que puedan...
# contribuir a la componente Y del campo electrostático
for i in range(int(X)):
    for k in range(int(Z)):
        for j in range(1, int(Y) - 1):

            # Designación de voltajes vecinos en el eje Z
            num1j = v[k, j - 1, i]
            num2j = v[k, j + 1, i]
            # Ingresa si los voltajes vecinos en el eje Y son iguales,...
            # es decir el punto analizado se encontraría en una región...
            # equipotencial
            if(abs(num1j - num2j) < tol):
                mayorj = v[k, j, i]
                sigj = 1
            else:
                # Ingresa si el punto analizado se encuentra en el...
                # límite posterior de una región equipotencial (en la...
                # dirección de Y)
                if(num1j == v[k, j, i]):
                    mayorj = num2j
                    sigj = -1
                # Ingresa si el punto analizado se encuentra en el...
                # límite frontal de una región equipotencial (en la...
                # dirección de Y)
                elif(num2j == v[k, j, i]):
                    mayorj = num1j
                    sigj = 1
                # Ingresa si el punto analizado se encuentra en una...
                # región de voltaje variante (En la dirección de Y)

```

```

        elif(num1j > num2j):
            mayorj = num1j
            sigj = 1
        else:
            mayorj = num2j
            sigj = -1

    Ey[k, j, i] = sigj * (mayorj - v[k, j, i]) / lml_i

E[:, :, :] = np.sqrt((Ex[:, :, :])**2 + (Ey[:, :, :])**2 + (Ez[:, :, :])**2)

return Ex, Ey, Ez, E

#-----
# Algoritmo para un problema bidimensional
#-----
# Variables internas:
# iter = número de iteraciones
# no_tol = número de elementos de la matriz de voltaje que no cumplen la...
#     tolerancia
if(dim == 2):
    # LLamado a la función "titulos_datos"
    X, Y, Z, v, maco, per, vi, E = titulos_datos(dim, longX, longY, 0, li,
                                                tol, lon, rel)
    # Barrido de todos los elementos de la malla
    for k in range(int(Z)):
        for j in range(int(Y)):
            for i in range(int(X)):

                # Los datos de "Excel" son almacenados en matrices
                v[k,j,i] = hoja2.cell(row = Y*k + j + 1, column = i+1).value
                p = hoja4.cell(row = Y*k + j + 1, column = i+1).value
                # La condición no permite permitividades nulas
                if(p == 0):
                    print("Matriz de permitividades tiene valor de cero,"
                          "cambiar matriz (no puede contener ceros)")
                    # Fin de la depuración
                    sys.exit()
                else:
                    per[k, j, i] = p

                # Condiciones de frontera
                if(j == 0 or i == 0 or j == int(Y-1) or i == int(X-1) or
                   v[k,j,i] != 0):
                    maco[k,j,i] = 0
                else:
                    maco[k,j,i] = hoja3.cell(row = Y*k + j + 1,
                                             column = i+1).value

vi[:, :, :] = v[:, :, :]

# LLamado a la función "graficar" para obtener los voltajes...
# ingresados en Excel
graficar(v, Z, Y, X, li, lon, 0, 0, 0, "Desea observar los gráficos del "

```

```

    "voltaje ingresados en Excel (1-SI): ",
    'Voltajes ingresados en Excel para la malla %s',
    'Voltajes ingresados en Excel para la malla %s.jpg',
    'No se mostrarán las matrices de voltajes ingresadas en Excel',
    'Voltaje (V)')

# Inicialización de las variables
iter = 0
no_tol = 1

# Iteraciones Gauss - Seidel hasta cumplir la tolerancia
while(no_tol != 0):

    # Llamado a la función "gauss_seidel_rel"
    v, no_tol = gauss_seidel_rel(v, maco, per, tol, dim, X, Y, 0, 0, 4,
                                rel)

    iter = iter + 1

# Llamado a la función "graficar" para obtener los voltajes equipotenciales
graficar(v, Z, Y, X, li, lon, 0, 0, 0, "Desea observar los gráficos de "
        "voltajes equipotenciales (1-SI): ",
        'Superficies equipotenciales en la malla %s',
        'Superficies equipotenciales en la malla %s.jpg',
        'No se mostrarán los gráficos de voltajes equipotenciales',
        'Voltaje (V)')

# Llamado a la función "campo"
Ex, Ey, Ez, E = campo(Z, Y, X, v, li, tol, lon)

# Llamado a la función "graficar" para obtener el campo electrostático...
# y superficies equipotenciales
graficar(v, Z, Y, X, li, lon, 1, Ex, Ey, "Desea observar los gráficos "
        "del campo electrostático en las superficies equipotenciales "
        "(1-SI): ",
        "Campo electrostático XY y equipotenciales en la malla %s",
        "Campo electrostático XY y equipotenciales en la malla %s.jpg",
        "No se mostrarán los gráficos del campo electrostático en las "
        "superficies equipotenciales", "Voltaje (V)")

# Llamado a la función "graficar" para obtener el campo electrostático
graficar(E, Z, Y, X, li, lon, 1, Ex, Ey, "Desea observar los gráficos "
        "del campo electrostático (1-SI): ",
        "Campo electrostático XY en la malla %s",
        "Campo electrostático XY en la malla %s.jpg",
        "No se mostrarán los gráficos del campo electrostático",
        "Campo electrostático (V/m)")

#-----
# Algoritmo para un problema tridimensional
#-----
# Variables internas:
# iter = número de iteraciones
# no_tol = número de elementos de la matriz de voltaje que no cumplen la...
#     tolerancia

```

```

elif(dim == 3):
    # Llamado a la función "titulos_datos"
    X, Y, Z, v, maco, per, vi, E = titulos_datos(dim, longX, longY, longZ,
                                                li, tol, lon, rel)
    # Barrido de todos los elementos del cubo
    for k in range(int(Z)):
        for j in range(int(Y)):
            for i in range(int(X)):

                # Los datos de "Excel" son almacenados en matrices
                v[k,j,i] = hoja2.cell(row = Y*k + j + 1, column = i+1).value
                p = hoja4.cell(row = Y*k + j + 1, column = i+1).value
                # la condición no permite permitividades nulas
                if(p == 0):
                    print("Matriz de permitividades tiene valor de cero, "
                          "cambiar matriz (no puede contener ceros)")
                    # Fin de la depuración
                    sys.exit()
                else:
                    per[k, j, i] = p

                # Condiciones de frontera
                if(k == 0 or j == 0 or i == 0 or k == int(Z-1) or
                   j == int(Y-1) or i == int(X-1) or v[k, j, i] != 0):
                    maco[k, j, i] = 0
                else:
                    maco[k, j, i] = hoja3.cell(row = Y*k + j + 1,
                                                column = i+1).value

vi[:, :, :] = v[:, :, :]

# Llamado a la función "graficar" para obtener los voltajes...
# ingresados en Excel
graficar(v, Z, Y, X, li, lon, 0, 0, 0, "Desea observar los gráficos del "
        "voltaje ingresados en Excel (1-SI): ",
        "Voltajes ingresados en Excel para la malla %s",
        "Voltajes ingresados en Excel para la malla %s.jpg",
        "No se mostrarán las matrices de voltaje ingresadas en Excel",
        "Voltaje (V)")

# Inicialización de las variables
iter = 0
no_tol = 1

# Iteraciones Gauss - Seidel hasta cumplir la tolerancia
while(no_tol != 0):

    # Llamado a la función "gauss_seidel_rel"
    v, no_tol = gauss_seidel_rel(v, maco, per, tol, dim, X, Y, 1,
                                 int(Z)-2, 6, rel)

    iter = iter + 1

```

```

# Llamado a la función "graficar" para obtener los voltajes equipotenciales
graficar(v, Z, Y, X, li, lon, 0, 0, 0, "Desea observar los gráficos de "
        "voltajes equipotenciales (1-SI): ",
        "Superficies equipotenciales en la malla %s",
        "Superficies equipotenciales en la malla %s.jpg",
        "No se mostrarán los gráficos de voltajes equipotenciales",
        "Voltaje (V)")

# Llamado a la función "campo"
Ex, Ey, Ez, E = campo(Z, Y, X, v, li, tol, lon)

# Llamado a la función "graficar" para obtener el campo electrostático...
# y superficies equipotenciales
graficar(v, Z, Y, X, li, lon, 1, Ex, Ey, "Desea observar los gráficos "
        "del campo electrostático en las superficies equipotenciales "
        "(1-SI): ",
        "Campo electrostático XY y equipotenciales en la malla %s",
        "Campo electrostático XY y equipotenciales en la malla %s.jpg",
        "No se mostrarán los gráficos del campo electrostático en las "
        "superficies equipotenciales", "Voltaje (V)")

# Llamado a la función "graficar" para obtener el campo electrostático
graficar(E, Z, Y, X, li, lon, 1, Ex, Ey, "Desea observar los gráficos "
        "del campo electrostático (1-SI): ",
        "Campo electrostático XY en la malla %s",
        "Campo electrostático XY en la malla %s.jpg",
        "No se mostrarán los gráficos del campo electrostático",
        "Campo Electrostático (V/m)")

else:
    # Mensaje mostrado cuando no se selecciona ninguna de las dos...
    # dimensiones posibles
    print("Dimensión no válida")

# Finalización del cronómetro
tiempo_final = time()
tiempo_ejecucion = tiempo_final - tiempo_inicial
ho = round(np.floor(tiempo_ejecucion / 3600),0)
mi = round(np.floor((tiempo_ejecucion - ho * 3600) / 60),0)
se = round(tiempo_ejecucion - ho * 3600 - mi * 60,2)

print ('El tiempo de ejecución fue:',ho,"horas,",mi,"minutos,",se,"segundos.")

```

ORDEN DE EMPASTADO