



Trabajo Fin de Grado

Desarrollo de un modelo electrostático por el método de elementos finitos de una bobina para aplicaciones de alta frecuencia

Autor

Ernesto Alloza Maicas

Director/es

Antonio Usón Sardaña
Jesús Letosa Fleta

Escuela de Ingeniería y Arquitectura de Zaragoza
2019

Agradecimientos

El trabajo aquí presente no habría sido posible sin la ayuda de Antonio Usón Sardaña, el que, para mí, junto a Jesús Letosa Fleta, han sido los profesores de los que más he aprendido durante mis estudios de Grado en Ingeniería Eléctrica en la Escuela de Ingeniería y Arquitectura de la Universidad de Zaragoza. A ellos les agradezco su innovador método de docencia, sus ganas de enseñar y la dirección y codirección del presente trabajo.

A mi tío Rafael, antiguo alumno de Ingeniería Industrial del antiguo CPS, el cual desde la lejanía del Reino Unido siempre me ha dado su apoyo. Siempre ha sido un referente y posiblemente debido a él elegí esta profesión. A él le debo haberme apoyado en los comienzos con esas clases de matemáticas que tanto dieron sus frutos.

A las personas más importantes para mí, mis padres Ángel y Pilar y a mi hermana Sara, los cuales han tenido una paciencia infinita durante todos estos años. Unos creyendo en mí más que otros, pero sobre todo ayudándome y dándome todo lo que siempre ha estado en sus manos.

A la mejor persona que he podido conocer, Giulia Giampaolo, la cual, siempre ha sido el más grande de mis apoyos y con la que a partir de ahora voy a compartir todo un proyecto de vida.

A Adrián, por siempre ser un punto de apoyo y referencia a pesar de apenas vernos.

A Eduardo, por todas las horas juntos en la biblioteca, por las largas llamadas entre otras aventuras.

A Tomás, por todos los viajes que hemos hecho y los que nos quedan por hacer, así como del apoyo mutuo que existe.

A Cestero, por todos los momentos que recordaremos y repetiremos siempre.

A Jaime, por su espíritu luchador y por ser una persona de referencia por sus méritos logrados.

A Mery, por su apoyo incondicional y su locura que tanto le caracteriza.

A Nieves por ser la mejor vecina y amiga que alguien puede tener.

A JR por ser como un hermano para mí.

Por último, a todas esas personas que han creído en mí y que han estado ahí siempre que las he necesitado.

*“Our virtues and our failings are inseparable, like force and matter.
When they separate, man is no more”*

Nikola Tesla

RESUMEN

Los transformadores, independientemente de su tamaño, no presentan solo fenómenos de inducción magnética y fenómenos disipativos, sino que también presentan fenómenos capacitivos debido a los efectos de las cargas eléctricas que aparecen entre los devanados, así como entre las espiras. Este efecto es importante a altas frecuencias o en el instante inicial de funcionamiento, cuando el transformador se conecta. Esto sucede independientemente de que sea un transformador de tamaño pequeño para aplicaciones de electrónica de potencia, o un transformador trifásico de gran tamaño para para aplicaciones eléctricas de potencia.

En este trabajo nos vamos a centrar en el cálculo de las capacidades existentes en un transformador, así como en la estimación del cálculo de la distribución inicial de tensiones en el mismo y de la capacidad equivalente, que es el valor de capacidad que presenta el transformador en una red de energía eléctrica, para altas frecuencias.

Se ha realizado un modelo en 2D de una bobina arrollada en un núcleo de un transformador de pequeño tamaño en dos programas de cálculo mediante elementos finitos, los cuales nos permiten obtener las capacidades existentes en él. Así mismo, se han comparado los resultados obtenidos en ambos softwares.

Posteriormente se ha desarrollado un programa computacional mediante Matlab que nos permite obtener la matriz de coeficientes de capacidad e inducción y generar un circuito equivalente en el software de análisis de circuitos LTSpice, el cual nos permite calcular la distribución inicial de tensiones y la capacidad equivalente.

Los scripts de parametrización implementados en Matlab permiten generalizar el modelo para otras dimensiones y aplicaciones.

ABSTRACT

Regardless of their size, the transformers not only present magnetic induction and dissipative phenomena, they also have capacitive phenomena due to the effects that appear between the windings, as well as between the spires. This effect is important at high frequencies or at the initial instant of operation when the transformer is connected. This happens regardless of whether it is a small size transformer for power electronics applications, or a large three phase transformer for power electrical applications.

In this degree thesis we are going to focus on the calculation of the existing capacities in a transformer, as well as on the estimation of the calculation of the initial voltage distribution in the transformer and on the calculation of the equivalent capacity, which would be equivalent to the capacity value of the transformer in an electrical power grid.

A 2D model of a coil rolled in a small transformer core has been made in two calculation programs, which use the finite elements method. They allow us to obtain the existing capacities in it. Also, the results obtained in both softwares have been compared.

Subsequently a computational program has been developed using Matlab, that allows us to obtain the coefficient matrix and generate an equivalent circuit in LTSpice, a circuit analysis software which allows us to analyze the initial voltage distribution and the equivalent capacity.

The parameterization scripts implemented in Matlab allow generalizing the model for other dimensions and applications.

Índice de contenidos

CAPÍTULO 1

1	Introducción.....	1
1.1	Antecedentes	1
1.2	Objetivos	2

CAPÍTULO 2

2	Comportamiento capacitivo del transformador.....	4
2.1	Ley de Gauss para el campo eléctrico	4
2.2	Diferencia de potencial.....	4
2.3	Capacidad	5
2.4	Coefficientes de capacidad y de inducción	5
2.4.1	Origen y procedencia.....	5
2.4.2	Definición.....	6
2.4.3	Matriz de Coeficientes	6
2.4.4	Interpretación y propiedades y unidades	7
2.5	Capacidad de un sistema de varios conductores.....	8

CAPÍTULO 3

3	Procedimiento para el cálculo del circuito equivalente a partir de los coeficientes de capacidad y de inducción	10
3.1	Cálculo de los condensadores equivalentes a partir de la Matriz Circuital 10	
3.2	Relación entre la Matriz Circuital y la Matriz de Coeficientes.....	12

CAPÍTULO 4

4	Cálculo de los coeficientes de capacidad mediante el MEF.....	14
4.1	Softwares de simulación mediante MEF utilizados.	15
4.1.1	Breve descripción de los softwares utilizados	15
4.2	Modelo realizado con GetDP	17
4.2.1	Geometría.....	17

4.2.2	Mallado	19
4.2.3	Procedimiento de cálculo y condiciones de contorno	19
4.3	Modelo realizado con FEMM.....	22
4.3.1	Geometría.....	22
4.3.2	Mallado	23
4.3.3	Procedimiento de cálculo y condiciones de contorno	24
4.4	Obtención de la Matriz de Coeficientes mediante MATLAB.....	27
4.4.1	Modelo realizado con GetDP	27
4.4.2	Modelo realizado con FEMM	29
 CAPÍTULO 5		
5	Aplicaciones de la Matriz de coeficientes de capacidad e inducción	31
5.1	Circuito capacitivo equivalente.....	31
5.2	Capacidad equivalente.....	34
5.3	Determinación de la distribución inicial de tensiones	36
 CAPÍTULO 6		
6	Conclusiones.....	40
6.1	Vías futuras	41
BIBLIOGRAFÍA		42
ANEXO I.....		44
Códigos implementados por el autor		44
Código archivo “Transformer_613783664.pro”		44
Código archivo “Fuente.m”		47
Código archivo “Cable.m”		47
Código archivo “Condensador.m”		47
Código archivo “Computecapacitances_function.m”		48
Código archivo “Computecapacitances_LTSPICE.m”		49
Código archivo “Modelo_Trafo_Automatizado.m”		51

Código archivo “Computecapacitances_function_FEMM.m”	53
Código archivo “Computecapacitances_LTSPICE_FEMM.m”	54

Introducción

1.1 Antecedentes

Desde los primeros cursos del Grado en Ingeniería Eléctrica en la Universidad de Zaragoza, participé en cursos voluntarios donde se introducía al alumnado en el conocimiento de fenómenos electromagnéticos y en el cálculo de campos mediante métodos de elementos finitos (MEF), lo que hizo que mostrase gran interés en este sector de la ingeniería eléctrica.

Años después, no habiendo visto nada más sobre MEF, surgió la oportunidad de realizar un intercambio en la Universidad de Lieja (Bélgica) en donde trabajan personal docente e investigador que habían realizado su propio software de resolución de problemas electromagnéticos mediante MEF (GetDP), el cual fue utilizado en el presente trabajo.

Este proyecto fue propuesto y semidesarrollado durante mi estancia en la Universidad de Lieja como parte del programa Erasmus+. Por diversos motivos, el proyecto no pudo finalizarse en la universidad de origen, motivo que llevó consigo la decisión de trasladar el trabajo realizado e intentar aprovecharlo en la Universidad de Zaragoza.

Gracias al director y codirector del presente trabajo se decidió dar un enfoque distinto al proyecto, pudiendo finalizarse tal y como se muestra en la presente memoria.

Originalmente los intereses residían en implementar un modelo completo (parte inductiva, resistiva y capacitiva) en 2D para un transformador de aplicaciones de electrónica de potencia. En la Universidad de Zaragoza se reorientó para ceñirse solamente a la parte electrostática.

Se ha realizado un modelo de un pequeño transformador, pero este puede extrapolarse a grandes transformadores de distribución.

Hacer notar el significado de energizar, es decir, el primer efecto que sucede en un transformador de gran tamaño en el instante inicial a su conexión. Este se carga electrostáticamente en los primeros microsegundos, en un transitorio muy pequeño, pero muy complejo porque la distribución de cargas no es igual en todas las partes del transformador y esto crea problemas en la distribución del potencial que hacen que la geometría y la fabricación de estos deba realizarse con especial cuidado.

El fenómeno es similar al que sucede en una línea eléctrica, en la cual se carga todo electrostáticamente. Es por eso, por lo que cuando se desconecta una línea eléctrica, los ingenieros encargados de realizarlo denominan a este proceso descargo de la línea.

Además, en el nuevo enfoque del proyecto se propuso realizar modelos con el conocido software libre FEMM, comparando y verificando los resultados obtenidos con los calculados previamente con GetDP.

1.2 Objetivos

Como hemos comentado, el presente trabajo va a centrarse en el cálculo de las capacidades entre espiras de un transformador y al posterior análisis de dos fenómenos relevantes que suceden en grandes transformadores trifásicos de potencia. Estos son la distribución inicial de tensiones y la capacidad equivalente. Por tanto, los objetivos del presente proyecto son:

- Generar modelos electrostáticos de MEF en GetDP y FEMM para calcular los coeficientes de inducción y de capacidad de una bobina en la que se puede elegir el número de espiras entre 1 y 32.
- Calcular automáticamente con Matlab, la Matriz de Capacidades de una bobina en la que se puede elegir el número de espiras entre 1 y 32, a partir de los coeficientes obtenidos en los programas de MEF.
- Crear automáticamente circuitos equivalentes en LTSpice a partir de las Matrices de Coeficientes y de Condensadores.
- Utilizar los circuitos equivalentes obtenidos para calcular la capacidad equivalente y distribución de tensiones con programas de circuitos.

Para facilitar la parametrización y análisis de datos los programas de elementos finitos, estos se manejan desde Matlab. También se comparan los resultados obtenidos con GetDP y FEMM y se calculan las capacidades equivalentes del circuito equivalente obtenido con GetDP y FEMM comparando los resultados.

Comportamiento capacitivo del transformador

En los transformadores se presentan tres fenómenos electromagnéticos [1].

- Inducción magnética: modelados por inductancias, así como acopladores inductivos.
- Disipativos: la disipación de energía proviene de diversos fenómenos, como efecto Joule agravadas por el efecto “skin” y los efectos de proximidad, así como las pérdidas magnéticas por histéresis. Se modelan con resistencias.
- Capacitivos: vinculados a capacitancias parásitas. Se desarrollan entre los distintos conductores del transformador. Son modelados solo por una red de condensadores.

Como explico en el apartado uno, en el presente proyecto nos centramos en los fenómenos capacitivos. Estos son debidos a cargas eléctricas y son descritos por la 1ª Ley de Maxwell, también llamada Ley de Gauss para el campo eléctrico.

2.1 Ley de Gauss para el campo eléctrico

La Ley de Gauss permite calcular campos eléctricos en una región del espacio.

Su forma diferencial afirma que la divergencia del campo eléctrico es proporcional a la densidad de carga eléctrica, es decir:

$$\vec{\nabla} \cdot \vec{E} = \frac{\rho}{\epsilon} \quad [2.1]$$

2.2 Diferencia de potencial

Se define la diferencia de potencial (ddp) entre los puntos A y B ($V_B - V_A$) como

$$\Delta V = V_B - V_A = \frac{\Delta U}{q_0} = - \int_A^B \vec{E} \cdot d\vec{l} \quad [2.2]$$

En forma diferencial permite obtener la relación local entre el campo eléctrico el potencial:

$$\vec{E} = -\nabla V \quad [2.3]$$

Que sustituyendo en 2.2 permite obtener la relación local entre el potencial y la carga:

$$\nabla^2 V = -\frac{\rho}{\epsilon} \quad [2.4]$$

Ecuación conocida como “de Poisson”.

2.3 Capacidad

El potencial de un único conductor aislado, que contiene una carga q , depende del tamaño, de la forma del conductor y del dieléctrico que lo rodea. En general, cuanto mayor es la superficie del conductor, mayor es la cantidad de carga que puede almacenar para un determinado potencial [8]. Por lo tanto, la capacidad es la habilidad de un sistema para almacenar y mantener una carga eléctrica para un determinado potencial eléctrico existente, y se define como:

$$C = \frac{q}{V} \quad [2.5]$$

Como el potencial es siempre proporcional a la carga esta relación no depende ni de Q ni de V , sino sólo del tamaño, forma del conductor y del dieléctrico que lo rodea, y es siempre positivo.

2.4 Coeficientes de capacidad y de inducción

2.4.1 Origen y procedencia

En electrostática la magnitud fundamental que necesitamos conocer es el potencial en todo punto del espacio.

La resolución del problema consiste en resolver la ecuación en derivadas parciales de Poisson [2.4] con las condiciones de contorno adecuadas.

La solución de este problema permite hallar el campo eléctrico todos los puntos de interés, la carga de los conductores y la energía almacenada. Esta solución se obtiene en este proyecto mediante resolución numérica por el método de Elementos Finitos en dos dimensiones, utilizando los programas de software libre GetDP y FEMM.

2.4.2 Definición

Partimos de un conjunto de N conductores situados en una región libre de carga, donde cada conductor recibe una carga q_i , debido a la cual adquieren un potencial V_i . Se demuestra en [6], [9] que existe una solución para la ecuación [2.6] :

$$Q = C \cdot \Delta V \quad [2.6]$$

que nos da los potenciales de los conductores en función de sus cargas. Si suponemos el sistema lineal (no hay dieléctricos no lineales presentes) la carga de cada conductor puede ponerse como suma proporcional de los potenciales de cada conductor, lo que permite plantear un sistema de N ecuaciones.

$$q_i = \sum_{j=1}^N c_{ij} \cdot V_j \quad [2.7]$$

Donde c_{ii} se denomina coeficiente de capacidad y c_{ij} ($i \neq j$) coeficiente de inducción [9].

2.4.3 Matriz de Coeficientes

Simplemente es la notación matricial de las ecuaciones lineales descritas en el apartado anterior. Para un caso de N conductores queda:

$$Q = C \cdot V \quad [2.8]$$

Donde \mathbf{Q} y \mathbf{V} son vectores columna de N elementos, con las cargas y tensiones respectivamente y \mathbf{C} es una matriz $N \times N$ formada por los coeficientes de capacidad e inducción [4]:

$$\begin{bmatrix} q_1 \\ q_2 \\ \dots \\ q_n \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & \dots & C_{1n} \\ C_{21} & C_{22} & \dots & C_{2n} \\ \dots & \dots & \dots & \dots \\ C_{n1} & C_{n2} & \dots & C_{nn} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ \dots \\ V_n \end{bmatrix} \quad [2.9]$$

2.4.4 Interpretación y propiedades y unidades

- **Los coeficientes de capacidad, c_{ii} ,** son los elementos de la diagonal principal de la matriz \mathbf{C} . Corresponden a la carga por unidad de potencial que aparece en el conductor i cuando dicho conductor está a potencial unidad y el resto de los conductores están a tierra.

Además, estos coeficientes son siempre positivos: $c_{ii} > 0$.

- **Los coeficientes de inducción, c_{ij} ,** son el resto de los elementos de la matriz \mathbf{C} . Cada coeficiente es la carga por unidad de potencial que se induce en el conductor i cuando el conductor j está a potencial unidad y el resto de los conductores (incluyendo el i) están a tierra.

Los coeficientes de inducción son simétricos: $c_{ij} = c_{ji}$. Lo que significa que la carga que se induce en el conductor i cuando el j está a potencial unidad y el resto a tierra, es la misma que se induce en el conductor j cuando es el i el que está a potencial unidad y el resto a tierra. Esto implica que la matriz \mathbf{C} es simétrica.

Además, los coeficientes de inducción son siempre negativos o nulos: $c_{ij} < 0$.

Debido a que relacionan una carga con un potencial, ambos coeficientes se miden en Faradios:

$$1F = \frac{C}{V} \quad [2.10]$$

Sin embargo, a pesar de esta interpretación, debe quedarnos claro que se trata de magnitudes puramente geométricas, puesto que su valor queda totalmente determinado por la forma y disposición espacial de los N conductores y dieléctricos presentes [4].

Para ilustrar lo anteriormente explicado, mostramos un ejemplo para cuando tenemos un solo conductor, y dos conductores. En el primer caso la matriz se reduce a un solo elemento:

$$Q = C \cdot V \quad [2.11]$$

y cuando hay dos conductores en el sistema:

$$\begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} \\ C_{12} & C_{22} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \end{bmatrix}$$

ó

[2.12]

$$q_1 = C_{11} \cdot V_1 + C_{12} \cdot V_2$$

$$q_2 = C_{12} \cdot V_1 + C_{22} \cdot V_2$$

2.5 Capacidad de un sistema de varios conductores

Podemos definir la capacidad mutua como la capacidad entre dos conductores cercanos; en nuestro caso esto sucede entre las espiras de una bobina. Esta capacidad aparece tanto entre las espiras que están contiguas una a otra, como entre las que están a mayor distancia.

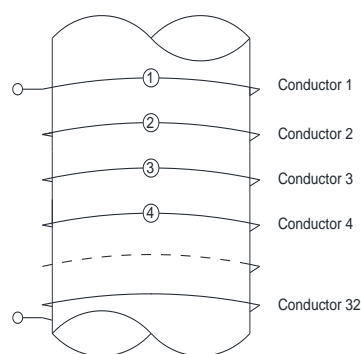


Fig.2.1. Ilustración de una bobina que ejemplifica nuestro problema a tratar.

Si cargamos la bobina, cada conductor recibirá una carga q , adquiriendo un potencial V , lo que dará lugar a una diferencia de potencial entre los distintos conductores, y a su vez estos conductores separados por aire pueden almacenar cargas iguales y opuestas, por lo que se estará formando un condensador entre una espira y otra. Esto puede verse ilustrado en la Fig.2.1.

El número de condensadores que se crean entre N conductores viene dado por la ecuación [2.13]

$$N^{\circ} \text{ condensadores entre } N \text{ conductores} = \frac{N \cdot (N - 1)}{2} \quad [2.13]$$

Además, existen N condensadores que representan las líneas de campo que van de cada conductor al infinito. Este valor corresponde con la autocapacidad del propio conductor [4].

Si simplificamos el caso de capacidades mutuas para N conductores, y lo realizamos para cuatro conductores situados en una región libre de carga, según la ecuación [2.13] se crean 6 condensadores entre los conductores y 4 conductores debido a la autocapacidad de cada conductor, como puede verse en la Fig.2.2.

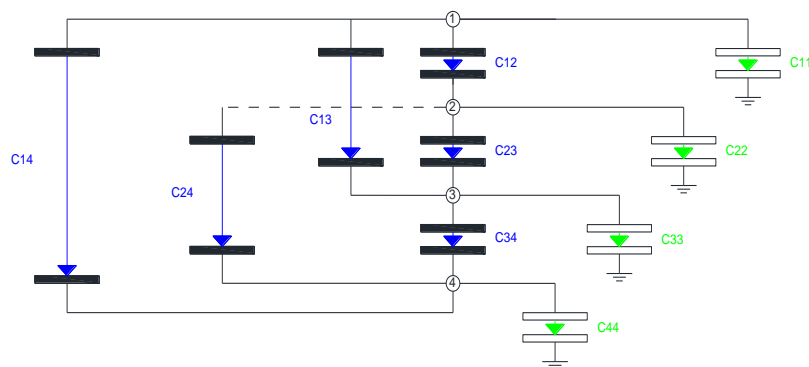


Fig.2.2. Ilustración de cómo se forman condensadores entre los distintos conductores de una bobina de cuatro espiras y su equivalente circuital.

Procedimiento para el cálculo del circuito equivalente a partir de los coeficientes de capacidad y de inducción

3.1 Cálculo de los condensadores equivalentes a partir de la Matriz Circuital

Continuando con el ejemplo del apartado 2.5. en el que hemos simplificado nuestro transformador en una bobina de cuatro espiras, se explica cómo se ha construido un circuito equivalente con condensadores y como se ha obtenido el valor de cada uno de ellos [4].

Si representamos cada conductor como un nodo, podemos modelar nuestro sistema de cuatro conductores como un circuito en el que entre cada nodo se crea un condensador. Además, se ponen otro condensador entre cada nodo y tierra.

Como hemos visto en el apartado 2.5, tenemos 6 condensadores entre conductores y 4 de autocapacidad entre cada conductor y tierra.

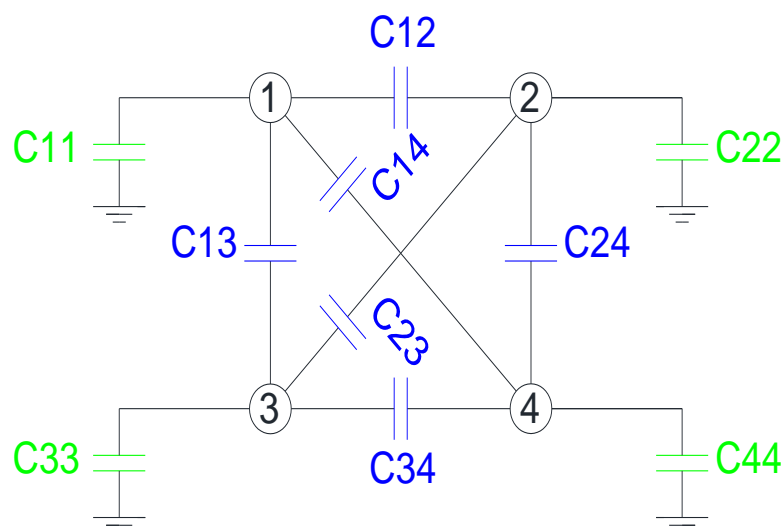


Fig.3.1. Circuito capacitivo equivalente de una bobina de cuatro espiras.

Para obtener el valor de las capacidades de cada condensador se hace uso de la definición de coeficientes de capacidad e inducción vista en el apartado 2.4. Por lo tanto, para ello se ha puesto un conductor a 1 Voltio y el resto de los conductores a tierra.

Se han redistribuido los conductores y condensadores tal y como puede verse en la Fig.3.2, podemos observar de una forma más sencilla el circuito equivalente:

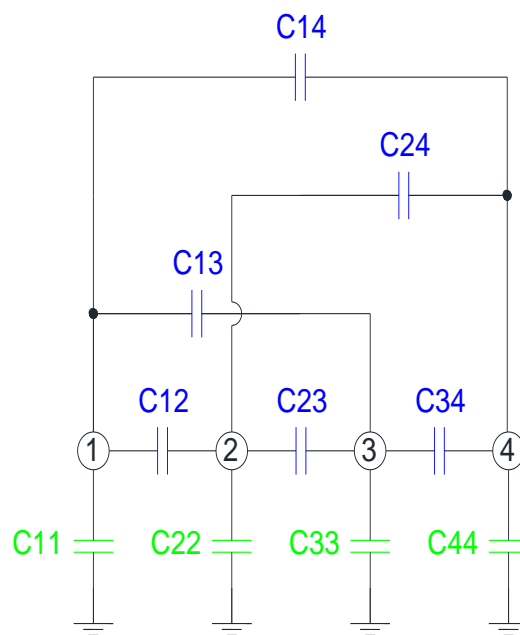


Fig.3.2. Circuito equivalente redistribuido de un sistema de cuatro conductores.

A la vista del circuito, tenemos que la carga Q_1 en el conductor 1, dados los voltajes V_1 , V_2 , V_3 y V_4 en los 4 conductores es:

$$q_1 = (C_{11} + C_{12} + C_{13} + C_{14}) \cdot V_1 - C_{12} \cdot V_2 - C_{13} \cdot V_3 - C_{14} \cdot V_4 \quad [3.1]$$

y para el resto de las cargas tenemos:

$$q_2 = -C_{21} \cdot V_1 + (C_{22} + C_{21} + C_{23} + C_{24}) \cdot V_2 - C_{23} \cdot V_3 - C_{24} \cdot V_4 \quad [3.2]$$

$$q_3 = -C_{31} \cdot V_1 - C_{32} \cdot V_2 - (C_{33} + C_{31} + C_{32} + C_{34}) \cdot V_3 - C_{34} \cdot V_4 \quad [3.3]$$

$$q_4 = -C_{41} \cdot V_1 - C_{42} \cdot V_2 - C_{43} \cdot V_3 - (C_{44} + C_{41} + C_{42} + C_{43}) \cdot V_4 \quad [3.4]$$

Estas cuatro ecuaciones pueden escribirse de forma matricial:

$$\begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} = \begin{bmatrix} (C_{11} + C_{12} + C_{13} + C_{14}) & -C_{12} & -C_{13} & -C_{14} \\ -C_{21} & (C_{22} + C_{21} + C_{23} + C_{24}) & -C_{23} & -C_{24} \\ -C_{31} & -C_{32} & (C_{33} + C_{31} + C_{32} + C_{34}) & -C_{34} \\ -C_{41} & -C_{42} & -C_{43} & (C_{44} + C_{41} + C_{42} + C_{43}) \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} \quad [3.5]$$

3.2 Relación entre la Matriz Circuital y la Matriz de Coeficientes

Como se puede ver en [4], [15], los programas de MEF nos dan como resultado los coeficientes de capacidad y de inducción, los cuales podemos poner de forma matricial en la Matriz de Coeficientes, tal y como aparecen en la ecuación [2.7]. Esto es lo que se ha realizado en el capítulo 4.

Como nuestro propósito ha sido crear el modelado circuital del fenómeno físico, hemos necesitado encontrar la forma de hallar el valor de los condensadores que aparecen en la ecuación [3.5] a partir de los coeficientes de capacidad e inducción. Para ello se han tenido que comparar la Matriz de Coeficientes con la Matriz de Circuital y mediante su comparación se ha observado que:

Los condensadores de la Matriz de la ecuación [3.5], C_{ij} (o C_{ji} porque son simétricos) son iguales a los coeficientes de capacidad C_{ij} (o C_{ji} porque también son simétricos) de la ecuación [2.7], pero de signo contrario. Esto es:

$$C_{ij} = -C_{ij} \quad [3.6]$$

Los valores de la diagonal principal de la Matriz Circuital se nombran como autocapacidades, y son los elementos del circuito de la Fig.3.2 que unen los conductores con tierra. Su valor es:

$$\begin{aligned}
 C_{ii} &= C_{ii} - \sum_{j=2}^N C_{ij} = \\
 &= C_{ii} - \sum_{j=2}^N (-C_{ij}) = \\
 &= C_{ii} + \sum_{j=2}^N C_{ij} = \sum_{j=1}^N C_{ij}
 \end{aligned}
 \tag{3.7}$$

De esta forma se observa que la Matriz de Circuital tiene todos los coeficientes positivos y además como ya hemos dicho, también es simétrica.

Los valores de los condensadores obtenidos a partir de la Matriz de Coeficientes según este método se pueden ver en el capítulo 5, y son los que se han utilizado para el modelo de LTSpice, que también se explica en dicho capítulo.

Cálculo de los coeficientes de capacidad mediante el MEF

Tradicionalmente la solución de las ecuaciones de Maxwell y en particular para el caso electrostático, la ecuación [2.4], se realizaba analíticamente mediante diversos métodos de cálculo, tomando consideraciones como una correcta elección de sistemas de coordenadas que se adaptan bien a las condiciones de frontera. Pero esto no siempre es posible para casos reales o con simetrías complejas. La aplicación de estas ecuaciones a casos particulares con una geometría general da lugar a ecuaciones diferenciales en derivadas parciales de difícil resolución analítica [2].

Gracias a la capacidad computacional actual se han podido implementar técnicas numéricas de aproximación, como es el caso del MEF, en el cual se divide la región en la que se desea encontrar la solución en subelementos adaptados a las fronteras. En cada uno de estos elementos finitos se construye una aproximación analítica sencilla de la solución [2].

En otras palabras, el MEF ofrece un procedimiento alternativo al analítico para resolver estas ecuaciones diferenciales mediante su resolución numérica. La viabilidad de este método ha dependido de la capacidad de cálculo de las máquinas que el ingeniero ha tenido a su alcance.

Desde el punto de vista matemático, los aspectos generales del MEF son:

- El MEF consiste en transformar la ecuación diferencial que define el problema electrostático [2.4] en un sistema de ecuaciones algebraicas. En este proceso matemático se manejan los conceptos de solución aproximada y funciones de aproximación.
- Definidas las funciones de aproximación, aparecen los conceptos de nudo y elemento, así como el de aproximación a nivel elemental, todo ello en el dominio matemático de validez de la ecuación diferencial.
- Obtenido el sistema de ecuaciones, se debe realizar la imposición de las condiciones de contorno.

Habitualmente los programas de elementos finitos están divididos en tres partes básicas: preprocesador, módulo de resolución y postprocesador. El preprocesador es un programa de CAD que permite introducir el modelo geométrico del sistema a analizar, las condiciones de contorno del problema y las características de los materiales.

Una vez creado el modelo geométrico se ejecuta el correspondiente módulo que realiza los cálculos de interés devolviendo los resultados numéricos.

Para interpretar los resultados, el software hace uso de funciones realizadas por el bloque post-procesador [2].

En este capítulo mostramos la forma en la que se ha implementado el MEF para obtener los coeficientes de capacidad y de inducción mediante dos softwares distintos. Se explican los programas utilizados, la modelización realizada y como se han obtenido los datos que posteriormente han sido objeto de estudio.

4.1 Softwares de simulación mediante MEF utilizados.

Para la realización y simulación de la bobina objeto de estudio, han sido utilizados dos softwares de MEF, los cuales son GetDP y FEMM. Ambos disponen de sus propios preprocesadores, módulos de cálculo y postprocesadores.

4.1.1 Breve descripción de los softwares utilizados

- **GetDP [10]**

GetDP es un software que lleva desarrollándose en la Universidad de Lieja desde 1997. Es distribuido bajo los términos de licencia GNU, licencia general pública (GLP), lo que significa que es un software libre es abierto a todo el mundo para su utilización.

GetDP significa entorno general para el tratamiento de problemas discretos, por sus siglas en inglés (*A General Environment for the Treatment of Discrete Problems*). Es un entorno de software científico para la solución numérica de ecuaciones integro-diferenciales, abierto al acoplamiento de problemas físicos, así como de MEF. Puede tratar problemas de hasta tres dimensiones.

La característica principal de GetDP es la cercanía entre su estructura interna (desarrollada en lenguaje C), la organización de datos que definen problemas discretos (introducidos por el usuario en archivos de datos ASCII) y las expresiones matemáticas simbólicas de estos problemas. Aunque según mi experiencia, no ha sido tan sencillo comprenderlo y trabajar con él.

Su objetivo es ser acogedor y de fácil uso para los niveles de desarrollo y aplicación: consiste en un entorno de trabajo en el que la definición de cualquier problema utiliza un número limitado de objetos, lo que hace que el entorno sea estructurado y conciso. Por lo tanto, ofrece a los investigadores herramientas de desarrollo avanzadas y una gran libertad para agregar nuevas funcionalidades.

- **FEMM [11]**

FEMM es también un software de MEF de libre licencia, por lo que permite su libre utilización con fines científicos. Es un programa desarrollado por el investigador David Meeker.

Es un entorno de software científico el cual nos permite resolver problemas de electromagnetismo entre otros, pero con la principal diferencia respecto al anterior en que solo es posible solucionar problemas en dos dimensiones.

Otra de las principales diferencias, es que dispone de una interfaz de usuario muy sencilla tanto para su preprocesador, como para su postprocesador. Su preprocesador dispone de entorno visual del estilo de dibujo asistido por ordenador (CAD) (no como en GetDP donde se debe introducir la geometría en coordenadas mediante un documento .txt escrito en caracteres ANSI). Esto nos facilita la creación de la geometría, así como la definición del problema y de los parámetros del mismo, al disponer de un entorno visual.

4.2 Modelo realizado con GetDP

4.2.1 Geometría

La modelización del transformador en geometría 2D se basa en una sección del transformador: los devanados se representan con su sección y forman un conjunto de conductores conectados por ecuaciones de acoplamiento eléctrico. La geometría en el caso de GetDP ha sido proporcionada por los profesores de la universidad de Lieja, esto significa que el dibujo completo (las coordenadas de todos los puntos del modelo que posteriormente se unen mediante líneas y círculos para formar la geometría completa), las características del aislante que forma el soporte de los devanados, así como la parte del núcleo ferromagnético provienen de los archivos proporcionados con formato .txt denominados "Transformer_613783664.geo" y "TransformerParam_613783664.geo", los cuales contienen toda la parametrización de la geometría en código ASCII. Mi trabajo en esta parte ha consistido en estudiar y comprender este código, por lo que podemos decir que este ha sido nuestro punto de partida.

El número de devanados que se ha utilizado es de dos, con un número de 16 vueltas para ambos devanados. Los conductores consisten en cobre sólido sin aislamiento basado en cables de 2,3 mm.

Para completar la geometría, se ha creado una región compuesta únicamente de aire que rodea el sistema a calcular, con su frontera suficientemente alejada para que el campo eléctrico en ella pueda considerarse aproximadamente cero. Esto permite modelar el transformador, como si estuviera en un espacio 2D infinito [1].

```

1 // Include parameters
2 Include "TransformerParam_613783664.geo";
3 // Insulator
4 Point(1000000) = {0.011200, 0.008841, 0.000000, lcinsulator_lay1_sect1};
5 Point(1000001) = {0.051200, 0.008841, 0.000000, lcinsulator_lay1_sect1};
6 Point(1000002) = {0.051200, 0.009328, 0.000000, lcinsulator_lay1_sect1};
7 Point(1000003) = {0.011200, 0.009328, 0.000000, lcinsulator_lay1_sect1};
8 Line(1000004) = {1000000, 1000001};
9 Line(1000005) = {1000001, 1000002};
10 Line(1000006) = {1000002, 1000003};
11 Line(1000007) = {1000003, 1000000};
12 Line Loop(1000008) = {1000004, 1000005, 1000006, 1000007};
13 Plane Surface(1000009) = {1000008};
14 Physical Surface(9000000) = {1000009};
15 // Round wire centered in x=0.013350, y=0.010641, diameter=0.002300
16 Point(1000010) = {0.013350, 0.010641, 0.000000, lcwisp_lay2_sect1};
17 Point(1000011) = {0.014500, 0.010641, 0.000000, lcwisp_lay2_sect1};
18 Point(1000012) = {0.013350, 0.011791, 0.000000, lcwisp_lay2_sect1};
19 Point(1000013) = {0.012200, 0.010641, 0.000000, lcwisp_lay2_sect1};
20 Point(1000014) = {0.013350, 0.009491, 0.000000, lcwisp_lay2_sect1};
21 Circle(1000015) = {1000011, 1000010, 1000012};
22 Circle(1000016) = {1000012, 1000010, 1000013};
23 Circle(1000017) = {1000013, 1000010, 1000014};
24 Circle(1000018) = {1000014, 1000010, 1000011};
25 Line Loop(1000019) = {1000015, 1000016, 1000017, 1000018};
26 Physical Line(1000020) = {1000015, 1000016, 1000017, 1000018};
27 Plane Surface(1000021) = {1000019};
28 Physical Surface(1000022) = {1000021};
29 // Round wire centered in x=0.015750, y=0.010641, diameter=0.002300
30 Point(1000023) = {0.015750, 0.010641, 0.000000, lcwisp_lay2_sect1};
31 Point(1000024) = {0.016900, 0.010641, 0.000000, lcwisp_lay2_sect1};
32 Point(1000025) = {0.015750, 0.011791, 0.000000, lcwisp_lay2_sect1};

```

Fig.4.1. Detalle del archivo "Transformer_613783664.geo",
 la cual contiene de la geometría del modelo de transformador en 2D.

Se ha tomado una simetría de tipo planar en la base, lo que significa que solo vamos a ver la mitad de los bobinados y del núcleo magnético y la línea de simetría actúa como un espejo que complementa la otra mitad del modelo. Además, los conductores se han considerado como cables rectos de un metro de longitud en profundidad, con dirección adentrándose en la pantalla.

Esta geometría es legible por GetDP y puede verse como se muestra en la Fig.4.2.

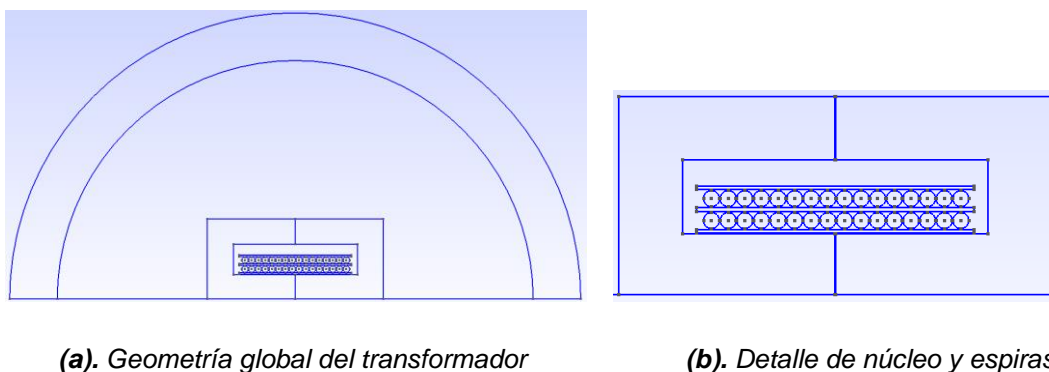


Fig.4.2. Geometría del transformador en 2D en GetDP.

Las diferentes regiones de componentes del transformador relevantes al cálculo electroestático tienen las siguientes características:

- Aire: Con permitividad eléctrica relativa de: $\epsilon = 1$
- Aislante: Con permitividad eléctrica relativa: $\epsilon = 4$

4.2.2 Mallado

Para realizar el mallado, GetDP se apoya en el software Gmsh [12], desarrollado por los mismos autores.

Gmsh es un generador de malla de MEF en 2D y/o 3D de código abierto con un motor CAD y postprocesador incorporado. Su objetivo de diseño es proporcionar una herramienta de mallado rápida, ligera y fácil de usar. Cuando le introducimos el archivo de geometría, el programa trabaja de forma iterativa para generar automáticamente el mallado.

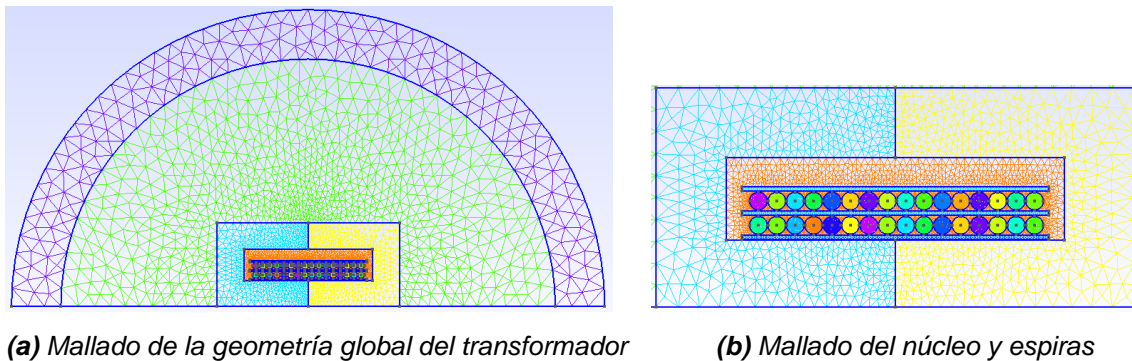


Fig.4.3. Mallado de la geometría del transformador en 2D en GetDP.

4.2.3 Procedimiento de cálculo y condiciones de contorno

Una vez se ha realizado la geometría se especifican las condiciones que modelan el modelo y el procedimiento de obtención que se ha utilizado para la estimación de los elementos de la Matriz de Coeficientes de capacidad e inducción.

Como se ha comentado anteriormente, hemos partido de una serie de archivos proporcionados que contenían la geometría, los parámetros del transformador y la formulación electrostática, esta última es el motor de cálculo que hay debajo de GetDP con toda la formulación diferencial para resolver problemas electrostáticos.

Para la resolución de nuestro problema, se ha creado un archivo denominado “Transformer_ 613783664.pro” (véase Anexo I. Código archivo “Transformer_

613783664.pro”) en el cual se han establecido los procedimientos de cálculo y condiciones necesarias para que nuestro modelo funcionara y pudiera proporcionarnos los coeficientes de capacidad. Esto significa, que en este script se ha implementado lo visto en el capítulo 3.

Para ello, se han incluido las regiones que posteriormente van a mallarse, estas son:

- Aislante
- Aire
- Núcleo ferromagnético
- Infinito

Posteriormente se han especificado las fronteras que conforman los 32 conductores, además de la línea que indica la simetría del problema.

Se ha decidido crear un caso variable en el que nosotros podemos especificar el número de conductores que queremos que se consideren en el modelo. Para ello se ha creado una variable que nos permite modificar manualmente dicho parámetro. De esta forma dentro de nuestro modelo de 32 conductores, podemos realizar problemas en los que se analicen los fenómenos electrostáticos entre el número de conductores que nosotros especifiquemos, siempre con la limitación geométrica que ofrece nuestro modelo, esto es, como mínimo la capacidad entre dos conductores, y como máximo, la capacidad existente entre los 32 conductores que conforman la bobina de nuestro transformador. Según la ecuación [2.13] corresponde a un total de 496 condensadores más 32 condensadores adicionales conectados a tierra.

Además, se ha implementado otra variable que nos permite indicar qué conductor va a ponerse a potencial de un voltio, mientras que el resto permanecen conectados a tierra.

Por último, se han implementado las librerías proporcionadas que contienen las funciones necesarias para que GetDP realice los cálculos.

Cada vez que ejecutemos GetDP, este nos va a devolver los resultados en dos ficheros .txt que contienen un número de valores correspondiente al número de conductores considerados en el modelo. Estos ficheros son:

- Fichero “voltages.txt”: contiene el potencial que se ha aplicado, en este caso solo observaremos dígitos correspondientes al número 0 entre los que habrá un 1 correspondiente al conductor que nosotros le hayamos indicado.
- Fichero “charges.txt”: Nos dará las cargas obtenidas previa simulación y cálculo de GetDP

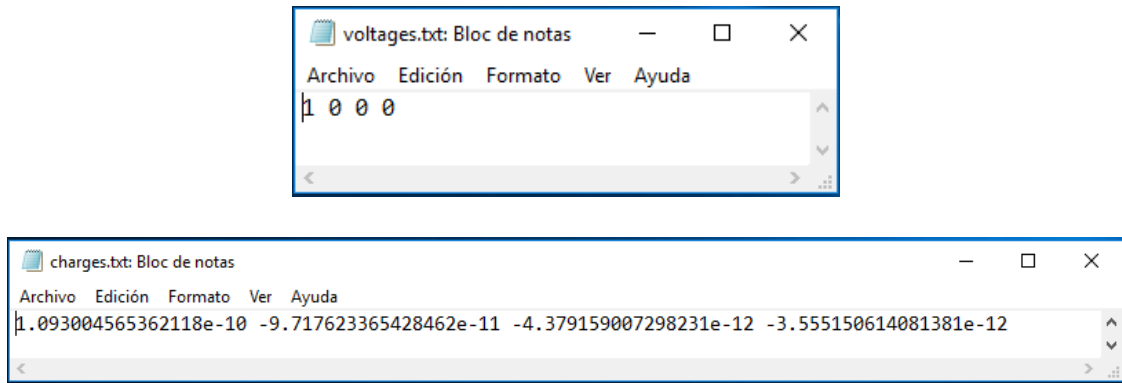
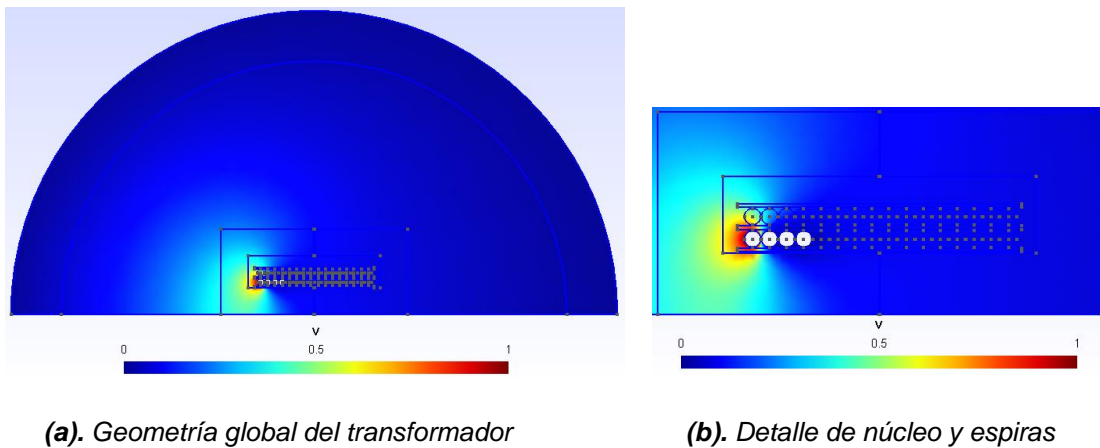


Fig.4.4. Visualización de los resultados en GetDP para un caso de 4 conductores con el 1º a 1V y el resto a tierra.

GetDP también nos permite ver visualmente la distribución de potencial en nuestro modelo.



(a). Geometría global del transformador

(b). Detalle de núcleo y espiras

Fig.4.5. Visualización de la distribución de potencial en GetDP para un caso de 4 conductores con el 1º a 1V y el resto a tierra.

Para construir la Matriz de Coeficientes, tal y como dice la interpretación de los mismos vista en el apartado 2.4.4, es necesario repetir el proceso de poner un

conductor a un voltio y el resto a tierra como se ha hecho en el Capítulo 3, posteriormente es necesario volver repetir esto mismo iterativamente con el resto de los conductores. Esto es precisamente lo que se ha implementado con la ayuda de Matlab, y se ilustra en el apartado 4.4.

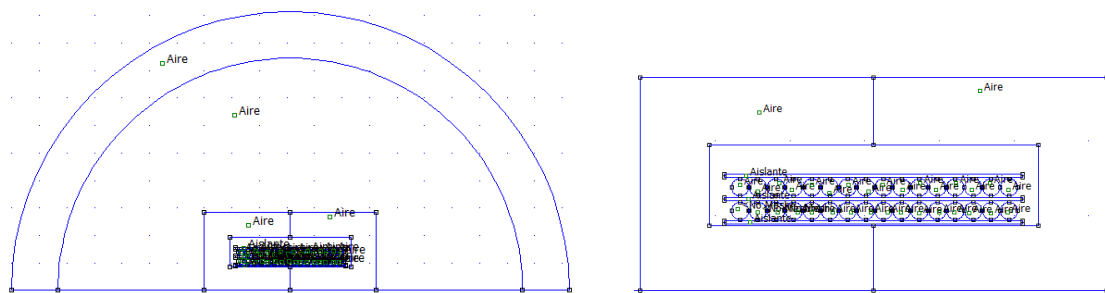
4.3 Modelo realizado con FEMM

Como se ha explicado en el apartado 1, he tenido que repetir con FEMM la parte del trabajo realizada con GetDP, para comparar los resultados y comprobar la validez de los dos modelos.

Hacer notar que esto ha sido un acierto, debido a que FEMM es mucho más intuitivo y sencillo de utilizar, ya que tiene la capacidad de poder observarse más fácilmente lo que está sucediendo, por lo que de manera didáctica es mucho más útil y enriquecedor. Esto va unido a que no es necesaria una cantidad enorme de código para hacerlo funcionar debido a su interfaz de usuario, por lo que es posible centrarse en lo importante del problema, sin perderse entre el código.

4.3.1 Geometría

En este caso se ha realizado la geometría partiendo de la proporcionada para el caso con GetDP. Para ello se han estudiado los puntos y coordenadas definidos en el fichero “Transformer_613783664.geo” y se han trasladado manualmente a FEMM. Mediante su entorno gráfico se han unido los puntos con líneas y con círculos replicando el modelo del que partimos.



(a). Geometría global del transformador

(b). Detalle de núcleo y espiras

Fig.4.6. Geometría del transformador en 2D en FEMM.

4.3.2 Mallado

FEMM no dispone de mallado adaptativo, por lo que se han refinado las mallas de forma manual. Para ello se han definido los materiales que van a conformar las diferentes regiones del modelo (aire y aislante) y posteriormente en las propiedades de dichos materiales se ha elegido el tamaño del mallado. Este puede realizarse más fino cerca de los lugares de interés para que la precisión de los resultados sea mayor, (ver Fig.4.7).

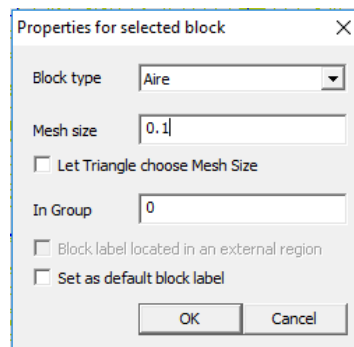
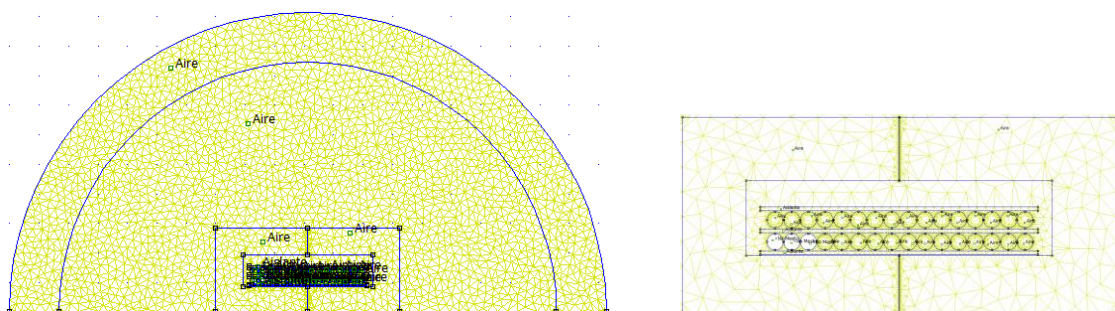


Fig.4.7. Ventana de propiedades de los materiales, donde se permite elegir el tamaño del mallado en FEMM.

En nuestro caso se han mallado las diferentes regiones, con distintos tamaños, aumentando la precisión de mallado en la zona de los conductores, que es donde más nos ha interesado, debido a que es donde existen los fenómenos electromagnéticos que hemos estudiado (Fig.4.8). Además, hemos buscado que los resultados sean los más similares posible al modelo realizado con GetDP.



(a) Mallado de la geometría global del transformador

(b) Mallado del núcleo y espiras

Fig.4.8. Mallado de la geometría del transformador en 2D en FEMM.

4.3.3 Procedimiento de cálculo y condiciones de contorno

Previamente a crear la geometría, FEMM nos solicita que especifiquemos el tipo de problema que va a tratarse. En nuestro caso, este ha sido un problema electrostático con simetría planar (ver Fig.4.9).

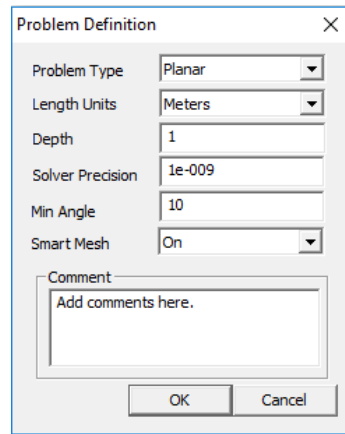


Fig.4.9. Ventana de definición de problema en FEMM.

Posteriormente a la creación de la geometría, se han definido los materiales que van a componer las diferentes regiones del modelo, estos son:

- Aire: Con permitividad eléctrica relativa de: $\epsilon = 1$
- Aislante: Con permitividad eléctrica relativa: $\epsilon = 4$

FEMM requiere que se establezcan condiciones de contorno explícitamente a la hora de crear el problema. Para problemas electrostáticos, tal y como explica el manual del programa [5], FEMM resuelve la ecuación [2.4] correspondiente a la ecuación de Poisson que se presentó en el apartado 2.4.1.

Por lo tanto, como usuarios se han introducido las siguientes condiciones de frontera:

- Condición de un voltaje fijo de 0 voltios en la línea que simula el infinito, como puede verse en la Fig.4.10:

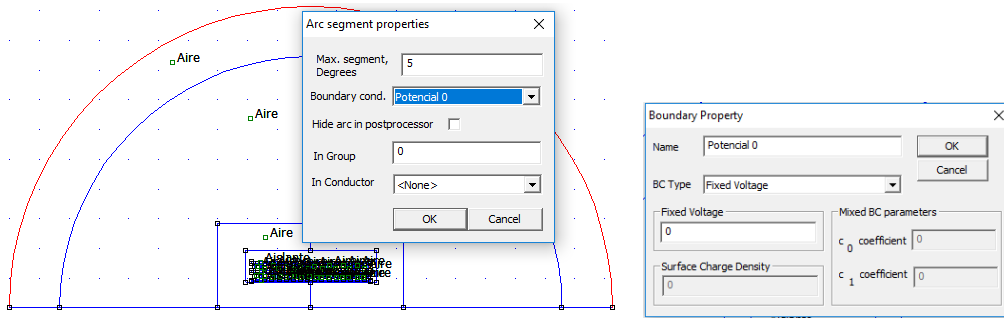


Fig.4.10. Condiciones de contorno para la frontera de la línea de infinito.

- Condición de frontera de voltaje mixto con coeficientes $c_0 = 0$ y $c_1 = 0$ para la línea de simetría, como puede verse en la Fig.4.11

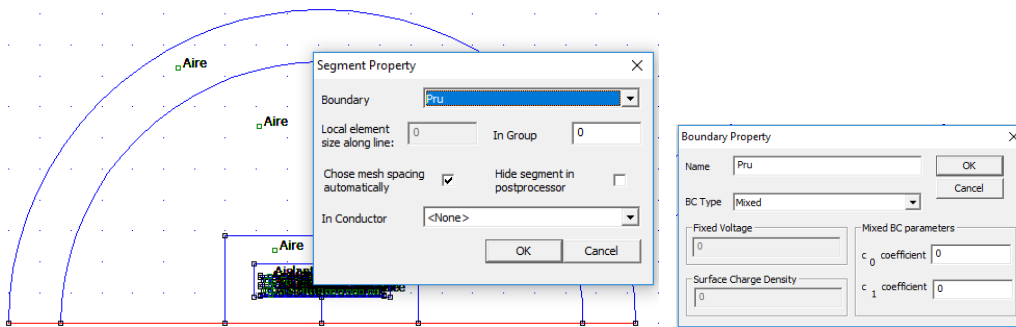
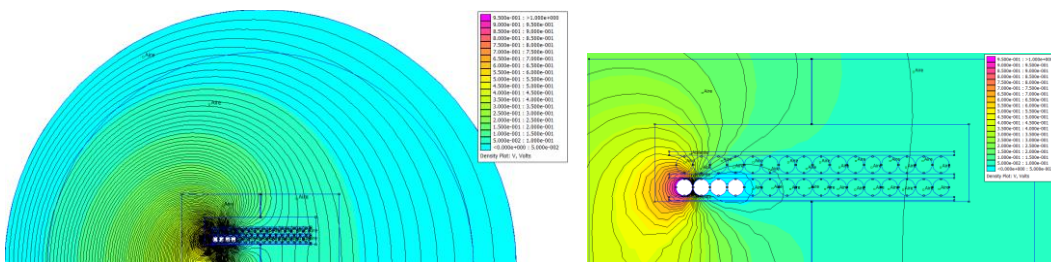


Fig.4.11. Condiciones de contorno para la línea de simetría.

De esta forma las líneas equipotenciales en mi modelo son paralelas a la línea de infinito y perpendiculares a la línea de simetría, de forma que es equivalente a tener en cuenta la otra parte simétrica no visible (ver Fig.4.12).



(a). Geometría global del transformador

(b). Detalle de núcleo y espiras

Fig.4.12. Visualización de la distribución de potencial en GetDP y de las líneas equipotenciales.

Por último, se ha definido como conductor a cada una de las regiones circulares que modelan un cable de la bobina, como puede verse en la Fig.4.13.

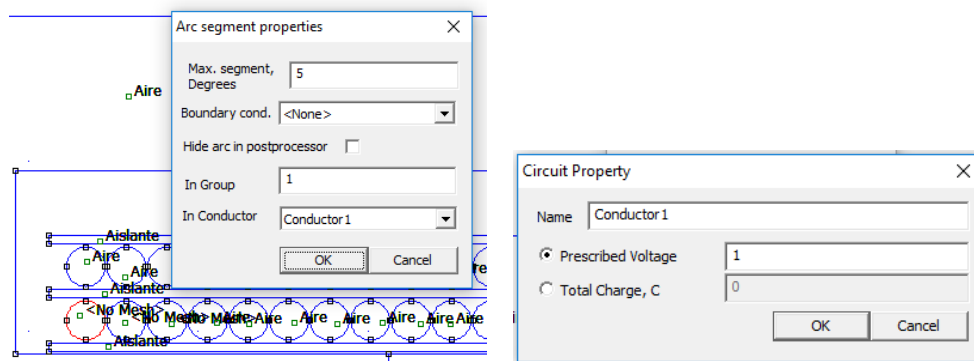


Fig.4.13. Ventana de definición de conductores

De esta forma podemos poner de forma manual el conductor que queramos a un voltio y el resto a potencial 0, para calcular los coeficientes de capacidad y de inducción.

Notar que, si queremos considerar menos de los 32 conductores del modelo, debemos modificar la región manualmente para que FEMM no la considere.

Para visualizar los resultados deberemos mallar el modelo con las condiciones consideradas tal y como se ha explicado en el apartado 4.3.2 y posteriormente pasar al postprocesador donde podremos ver los potenciales y cargas calculadas para cada conductor tal y como se muestra en la Fig.4.14:

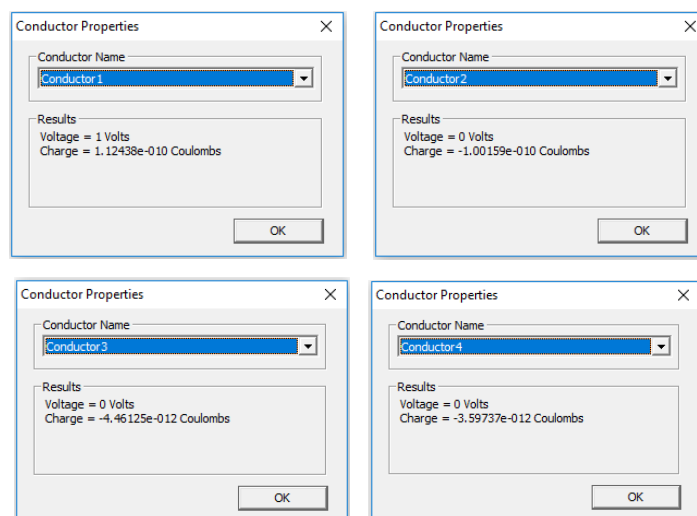
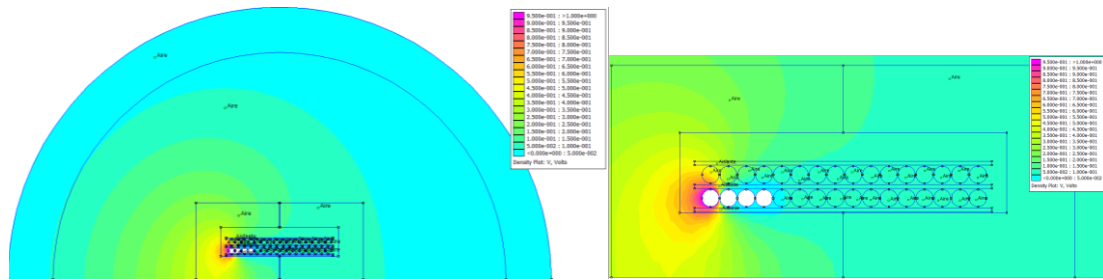


Fig.4.14. Visualización de los resultados en FEMM para un caso de 4 conductores con el 1º a 1V y el resto a tierra.

Entre otros muchos resultados más, FEMM nos permite también ver de forma visual la distribución de potencial de nuestro modelo (Fig.4.15):



(a). Geometría global del transformador

(b). Detalle de núcleo y espiras

Fig.4.15. Visualización de la distribución de potencial en FEMM para un caso de 4 conductores con el 1º a 1V y el resto a tierra.

4.4 Obtención de la Matriz de Coeficientes mediante MATLAB

En este apartado se explica cómo mediante la ayuda de Matlab, se han implementado varios códigos que nos permiten de forma automática ejecutar en bucle los programas de MEF utilizados. De esta forma, en cada iteración se modifica el conductor que va a estar a un voltio para recoger y almacenar los valores de carga y potencial para procesarlos y calcular la Matriz de Coeficientes de forma automática y, además, de forma variable dependiendo del número de conductores que queremos considerar.

Matlab es un software computacional no libre (para la realización de este proyecto se ha utilizado la versión estudiantil) el cual contiene un entorno de desarrollo con un lenguaje de programación propio. Este nos permite realizar cálculos, manipular matrices, representar datos mediante la implementación de algoritmos entre otras muchas funcionalidades. Otra funcionalidad de Matlab que nos ha sido muy útil ha sido la posibilidad de comunicarse con otros programas desarrollados en otros lenguajes distintos. Esto nos ha permitido manejar GetDP y FEMM a treves de él [13].

4.4.1 Modelo realizado con GetDP

Se ha creado una función denominada “Computecapacitances_function.m” en un script de Matlab (véase Anexo I. Código archivo “Computecapacitances_function.m”).

Este script ejecuta iterativamente GetDP tantas veces como conductores estemos considerando en el modelo. En cada iteración se ejecuta GetDp y se le indica que debe modificar el parámetro del fichero “Transformer_613783664.pro” que pone un conductor a un voltio y el resto a tierra, de forma que todos los conductores habrán estado a un voltio y el resto a tierra al final de la ejecución.

A su vez en cada iteración, el script accede a los ficheros resultado que presenta GetDP que se han presentado en el apartado 4.2.3: “voltages.txt” y “charges.txt”, de forma que va realizando el cociente entre las cargas y potenciales que aparecen en ambos y almacena el resultado en una matriz que al finalizar el proceso va a corresponder a la Matriz de Coeficientes con sus correspondientes coeficientes de capacidad e inducción.

Para un ejemplo de cuatro conductores Matlab nos muestra la matriz que se observa en la Fig.4.12:

```

C =

    1.0e-09 *

    0.1093   -0.0972   -0.0044   -0.0036
   -0.0972    0.1974   -0.0942   -0.0044
   -0.0044   -0.0942    0.1973   -0.0972
   -0.0036   -0.0044   -0.0972    0.1096
  
```

Fig.4.16. Matriz de Coeficientes obtenida con Matlab mediante los coeficientes da capacidad e inducción obtenidos con GetDP.

De esta forma hemos obtenido los coeficientes de capacidad e inducción vistos en la ecuación [2.7], pero para un caso en el que tenemos 4 conductores. Se muestran en la ecuación [4.1] modificado los coeficientes para que el orden de magnitud sea de 10^{-12} . (De esta forma podemos hablar de que los coeficientes son del orden de pico faradios).

$$\begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{bmatrix} = \begin{bmatrix} 109.3 & -97.2 & -4.4 & -3.6 \\ -97.2 & 197.4 & -94.2 & -4.4 \\ -4.4 & -94.2 & 197.3 & -97.2 \\ -3.6 & -4.4 & -97.2 & 109.6 \end{bmatrix} \cdot 10^{-12} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} \quad [4.1]$$

Según lo visto en el apartado 2.4, la matriz está compuesta por los coeficientes de capacidad y de inducción y cumple todas las propiedades expuestas, simetría y signos de los coeficientes.

4.4.2 Modelo realizado con FEMM

Como hemos visto en el apartado 4.3.3, el entorno gráfico de FEMM muestra los resultados de forma visual en ventana debido a su entorno de usuario.

Gracias a la incorporación de las secuencias de comandos en lenguaje Lua [5], ha dado a FEMM la posibilidad de agregar funciones de procesamiento de secuencia de comandos dando la posibilidad de utilizar FEMM junto con MatLab. Basándonos en esto, se ha automatizado FEMM de forma que puede verse el resultado en Anexo I. “Modelo_Trafo_Automatizado.m”.

“Modelo_Trafo_Automatizado.m” es un script de Matlab que contiene una función que permite automatizar nuestro modelo de FEMM de forma que pueda realizar lo que se ha hecho anteriormente con GetDP, es decir, en este script se ha implementado la posibilidad de seleccionar el número de conductores que se quiere considerar en el modelo.

Además, ejecuta FEMM de forma iterativa tantas veces como conductores se hayan considerado. En cada iteración, fija un conductor a un voltio y el resto a tierra, guardando los valores de potencial y de cargas que se obtienen en dos ficheros que se crean desde el propio script, denominados igual que para GetDP: “voltaje.txt” y “charges.txt”. Esto se ha realizado a propósito de esta forma debido a que realizando muy pocas modificaciones, haciendo posible aprovechar el código que se ha implementado con anterioridad para genera la Matriz de Coeficientes con GetDP. Este puede verse en Anexo I. “Computecapacitances_function_FEMM.m”, el cual realiza lo mismo que se ha explicado en el apartado 4.4.1. De forma que al final proporciona la Matriz de Coeficientes, la cual para un ejemplo de cuatro conductores puede verse en la Fig.4.13.

$$C =$$

$$1.0e-09 *$$

0.1124	-0.1002	-0.0045	-0.0036
-0.1002	0.2030	-0.0968	-0.0045
-0.0045	-0.0968	0.2029	-0.1000
-0.0036	-0.0045	-0.1000	0.1125

Fig.4.17. Matriz de Coeficientes obtenida con Matlab mediante los coeficientes de capacidad e inducción obtenidos con FEMM.

De esta forma también hemos obtenido los valores de la ecuación [2.7] y también los hemos modificado en la ecuación [4.2] para que aparezcan en el orden de magnitud de 10^{-12} :

$$\begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \\ Q_4 \end{bmatrix} = \begin{bmatrix} 112.4 & -100.2 & -4.5 & -3.6 \\ -100.2 & 203.0 & -96.8 & -4.5 \\ -4.5 & -96.8 & 202.9 & -100.0 \\ -3.6 & -4.5 & -100.0 & 112.5 \end{bmatrix} \cdot 10^{-12} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} \quad [4.2]$$

Puede observarse que las matrices de coeficientes que se incluyen en las ecuaciones [4.1] y [4.2] resultan muy similares.

Hacer notar que el preprocesador de FEMM también puede programarse, de forma que podríamos generar la geometría de nuestro modelo de transformador o cualquier otro mediante implementación de código. Esto podría hacer que pudiéramos generar automáticamente el modelo con más rapidez y sencillez solamente introduciendo los parámetros del modelo. En nuestro caso se ha realizado un primer caso de forma manual, centrándonos en una geometría concreta.

Aplicaciones de la Matriz de coeficientes de capacidad e inducción

5.1 Circuito capacitivo equivalente

Una vez se han obtenido los coeficientes de capacidad con los programas de MEF y se han construido las matrices de coeficientes mediante Matlab, se ha utilizado este último de nuevo para realizar los cálculos vistos en el apartado 3.2 para así obtener los valores de las capacidades que se incluirán en el circuito equivalente. Estos valores se han implementado en forma matricial y pueden verse en la Fig.5.1 obtenidos con GetDP y con FEMM (Para comparar los valores, nótese en la Fig.5.1 que Matlab nos da los valores de la matriz con órdenes de magnitud distintos).

CapacitanceMatrix =		CapacitanceMatrix =
1.0e-10 *		1.0e-09 *
0.0419	0.9718	0.0438
0.9718	0.0161	0.9415
0.0438	0.9415	0.0162
0.0356	0.0444	0.9717
		0.0356
		0.1002
		0.0045
		0.0036
		0.0016
		0.0968
		0.0016
		0.1000
		0.0045

(a). Obtenido con GetDP

(b). Obtenido con FEMM

Fig.5.1. Valor de los condensadores del circuito equivalente calculados a partir de los coeficientes obtenidos con los programas de MEF.

Una vez se ha obtenido el valor de los condensadores, el problema ha residido en como generar un circuito equivalente de forma automática a partir de Matlab, que sea capaz de colocar correctamente los condensadores con sus correspondientes valores y todo ello de forma variable dependiendo del número de conductores (entre 1 y 32) que queremos que formen parte en nuestro modelo.

Para ello se ha utilizado LTSpice [14], él cual es un software de simulación SPICE (Determinación de la Capacidad de Mejora del Proceso de Software) de alto rendimiento, que permite la captura de un circuito esquemático, visor de

formas de onda de señales entre otros para facilitar la simulación de circuitos analógicos.

Para darle solución al problema que se presenta, se ha creado un circuito aleatoriamente con los elementos que se han necesitado: condensadores, cables y fuente de tensión. Posteriormente se ha estudiado el código del archivo .asc de LTSpice que contiene la información de dicho circuito (véase Fig.5.2)

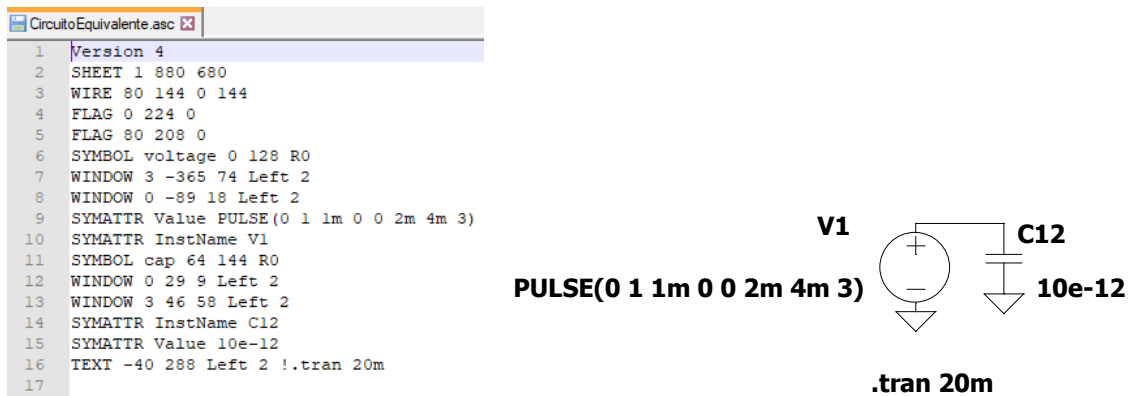


Fig.5.2. Ejemplo de código de un archivo .asc que contiene la información del circuito en LTSPICE .

Se han observado patrones y la lógica de funcionamiento de este código, así como la sintaxis que utiliza para crear cada uno de los elementos del circuito. Además, se ha interpretado la forma que utiliza para colocar los elementos creando conexiones mediante su sistema de coordenadas.

Se han desarrollado funciones en Matlab que contienen cada uno de los elementos que hemos necesitado. Estas pueden observarse en el Anexo I:

- Código archivo “Fuente.m”
- Código archivo “Cable.m”
- Código archivo “Condensador.m”

Posteriormente se ha hecho uso de estas funciones en el archivo “Computecapacitances_LTSPICE.m” para el modelo de GetDP y “Computecapacitances_ LTSPICE_FEMM.m” para el modelo de FEMM, junto a bucles “for” entre otros, para generar el circuito que nos interese en cada caso en un archivo .asc creado desde Matlab.

Estos códigos también pueden verse en Anexo I:

- Código archivo “Computecapacitances_LTSPICE.m”
- Código archivo “Computecapacitances_LTSPICE_FEMM.m”

El circuito que se ha generado para un caso de 4 conductores con GetDP puede verse en la Fig.5.3:

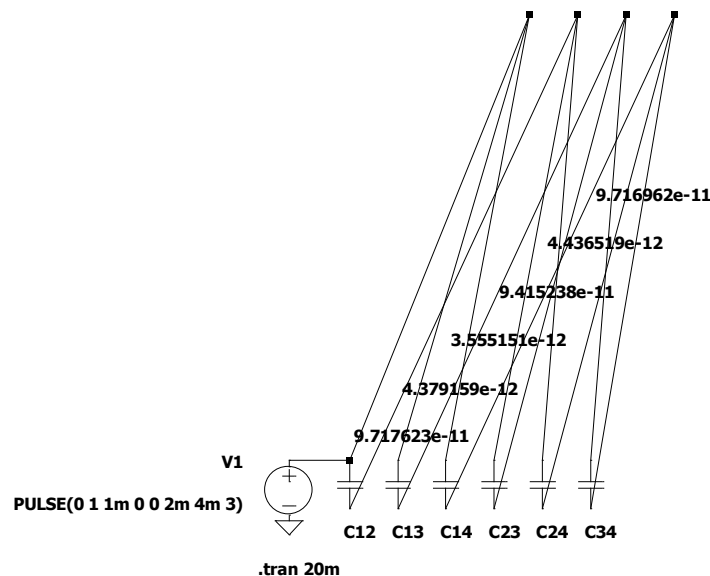


Fig.5.3. Circuito equivalente generado automáticamente para 4 conductores mediante Matlab, obtenidas las capacidades con GetDP.

Podemos observar que:

- La capacidad de los condensadores corresponde a los valores fuera de la diagonal principal de la matriz de la Fig.5.1 (a), los cuales han sido calculados según el procedimiento visto en el apartado 3.2.
- El número de condensadores cumple la ecuación [2.12] y es así para cualquier número de conductores que queramos considerar en el modelo, siempre dentro de la posibilidad de nuestro modelo físico.
- No aparecen los condensadores correspondientes a la diagonal principal debido a que en nuestros modelos hemos considerado que la tierra está a una distancia muy lejana, por lo tanto, la autocapacidad en los conductores es pequeña (diagonal principal de Fig.5.1).
- El modelo equivalente calculados los coeficientes con FEMM es igual que el de la Fig.5.3 pero con los valores de la matriz de la Fig.5.1 (b).

5.2 Capacidad equivalente

El modelo realizado en LTSpice nos ha permitido calcular la capacidad total del transformador. Para ello se ha introducido una fuente senoidal de voltaje 1V de pico a 1MHz.

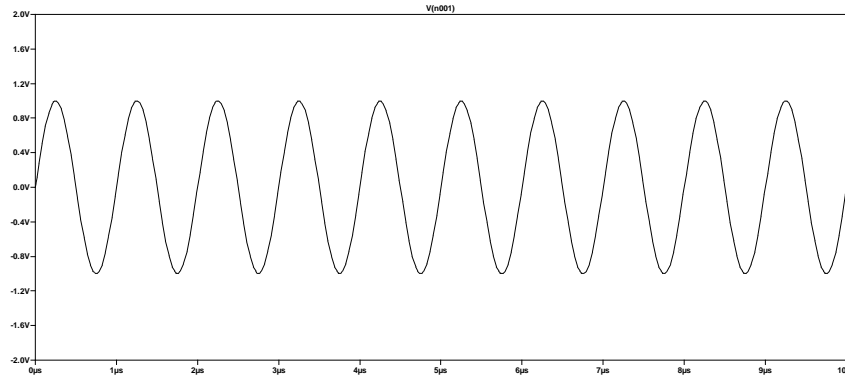


Fig.5.4. Señal de 1V de pico a 1MHz de la fuente utilizada para calcular la capacidad equivalente.

Se ha hecho uso de la sentencia .MEAS de LTSpice, la cual permite obtener las medidas indicadas en el esquemático. Esta ha sido utilizada para obtener la intensidad del circuito y posteriormente la capacidad total equivalente a partir de la intensidad obtenida y de la frecuencia igualando y sustituyendo las ecuaciones [5.1] y [5.2].

$$Z = \frac{V}{I} \quad [5.1]$$

$$Z = \frac{1}{2 \cdot \pi \cdot f \cdot C} \quad [5.2]$$

- La capacidad total para el circuito equivalente del modelo de 32 conductores calculado con GetDP es de: **14.40 pF**

```
itotal: i(v1)=9.04388e-005 at 5e-007  
ctotal: itotal/6.28meg=1.44011e-011
```

- La capacidad total para el circuito equivalente del modelo de 32 conductores calculado con FEMM es de **14.73 pF**

```
itotal: i(v1)=9.25072e-005 at 5e-007
ctotal: itotal/6.28meg=1.47304e-011
```

Para comprobar la veracidad de los resultados, se ha calculado analíticamente el caso para 4 conductores. Para ello se ha reorganizado el circuito manualmente como puede verse en la Fig.5.7. Además, se ha realizado una simulación para el mismo caso con el modelo de GetDP y de FEMM. Los resultados se muestran a continuación:

- Analíticamente:

$$C_{equivalente} = 3.85 \cdot 10^{-11} F$$

- GetDP:

```
itotal: i(v1)=0.000243663 at 5e-007
ctotal: itotal/6.28meg=3.87998e-011
```

- FEMM:

```
itotal: i(v1)=0.000250035 at 5e-007
ctotal: itotal/6.28meg=3.98145e-011
```

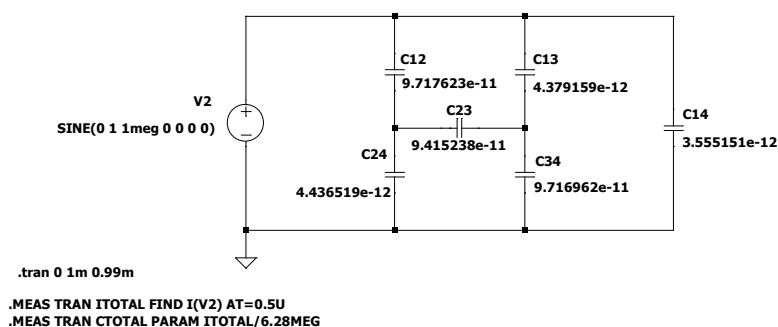


Fig.5.5. Circuito para 4 conductores reorganizado para calculo analítico de C equivalente.

5.3 Determinación de la distribución inicial de tensiones

Tal y como se describe en [3], la distribución de tensión en un arrollamiento ante transitorios de alta frecuencia puede dividirse en tres partes: distribución de tensión inicial, transitoria y final.

A efectos capacitivos solo se tiene en cuenta la distribución de tensión inicial. La separación temporal entre uno u otro es arbitraria en el sentido de que no hay separación estricta entre una fase u otra, aunque esta clasificación cualitativa es útil de cara a entender el proceso.

Cuando se introduce una onda de tensión en un devanado, se supone que la distribución inicial debería repartirse equitativamente entre todas las espiras, pero esto no sucede de esta forma

Esta situación se corresponde con el periodo inicial y acaba cuando los condensadores se cargan, lo que ocurre en un tiempo mínimo del orden de microsegundos. En esta etapa el transformador se comporta como una red puramente capacitiva y por lo tanto la tensión que aparece en cada nodo depende únicamente del valor de estas capacidades.

Se puede demostrar [3] que la distribución inicial de tensión en el punto x de un arrollamiento de longitud l , cuando se le aplica un escalón de tensión en un extremo con el opuesto a tierra, corresponde a la representada en la Fig.5.6.

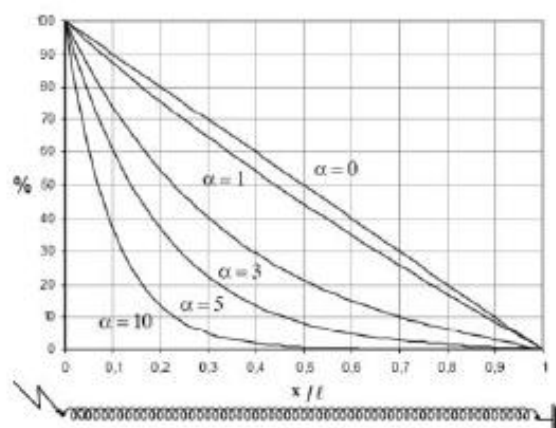


Fig.5.6. Distribución inicial de tensión para diferentes valores de α .

El factor α que aparece es una constante que depende de la geometría del transformador y es función de las capacidades a tierra C_g y en serie C_s de su circuito equivalente.

$$\alpha = \sqrt{\frac{(C_g)_{total}}{(C_s)_{total}}} \quad [5.3]$$

donde $(C_g)_{total}$ es la asociación paralela de las capacidades C_g , y $(C_s)_{total}$ es la asociación serie de las capacidades C_s :

La presencia de la capacidad a tierra causa una no uniformidad en la distribución de tensión. A mayor α , mayor es la sobretensión en el extremo del arrollamiento que recibe el impulso y menor es en el extremo a tierra, por lo que se comprende la necesidad de minimizar el valor de α [3].

Como hemos comentado, nuestro modelo no contempla las capacidades a tierra, por lo que, en una primera instancia, contemplamos una distribución inicial lineal, tal como dice la teoría que aparece en la Fig.5.6.

En mi trabajo he realizado la distribución de un modelo de 4 espiras, ya que he podido introducir las 4 capacidades a tierra de forma manual.

Para ello se ha tomado el circuito de la Fig.5.7, y se ha simulado bajo diferentes condiciones:

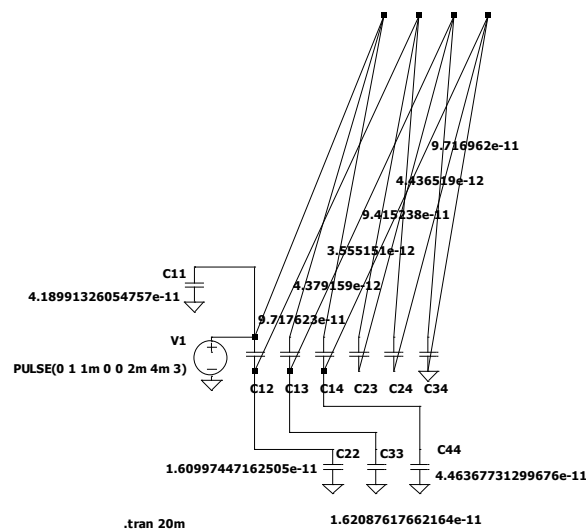


Fig.5.7. Circuito equivalente utilizado para calcular la distribución inicial de tensión para un modelo de 4 conductores.

- a) Considerando condensadores a tierra, incrementando el valor real que se obtiene en nuestro modelo en un orden de magnitud (orden de 10^{-11}) y con último punto desconectado de tierra.
- b) Considerando condensadores a tierra, incrementando el valor real que se obtiene en nuestro modelo en un orden de magnitud (orden de 10^{-11}) y con último punto a tierra.
- c) Considerando condensadores a tierra con el valor real que se obtiene en nuestro modelo (orden de 10^{-12}) y con último punto desconectado de tierra.
- d) Considerando condensadores a tierra con el valor real que se obtiene en nuestro modelo (orden de 10^{-12}) y con último punto a tierra.
- e) Sin considerar los condensadores a tierra (C_{ii}).

Los resultados obtenidos en LTSpice para cada uno de los casos se ven en la TABLA 5.1 y en la Fig.5.8.

Distribución de tensiones				
a) 4 cond con C tierra 10e11 masa	1,0000	0,5830	0,2797	0,0000
b) 4 cond con C tierra 10e11 NO masa	1,0000	0,6845	0,4910	0,3625
c) 4 cond con C tierra 10e12 masa (original)	1,0000	0,6511	0,3329	0,0000
d) 4 cond con C tierra 10e12 NO masa (original)	1,0000	0,9394	0,8964	0,8650
e) 4 cond sin C tierra	1,0000	0,6598	0,3398	0,0000

TABLA 5.1. Resultados obtenidos con LTSpice.

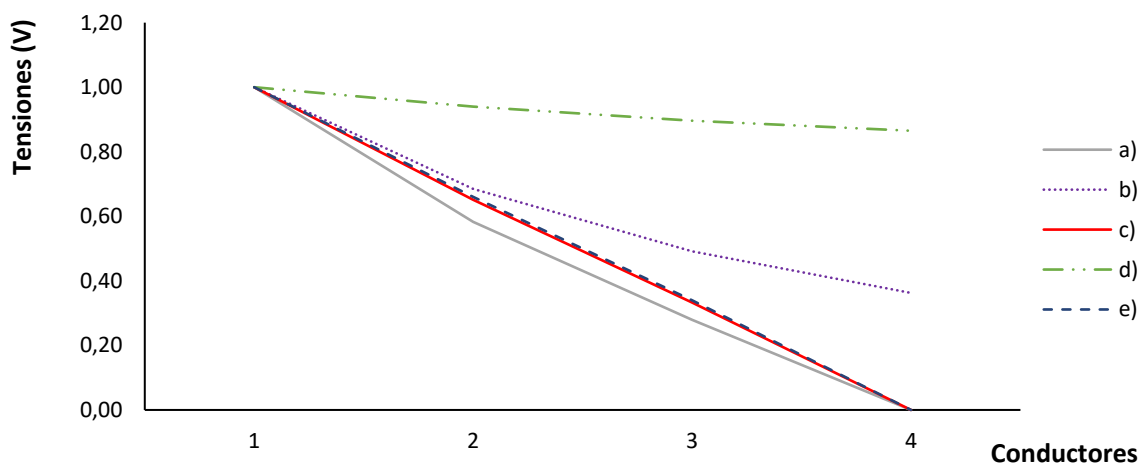


Fig.5.8. Distribución de tensión inicial para cada uno de los casos estudiados

Puede observarse, que los casos en los que se ha incrementado el valor de los condensadores a tierra, el coeficiente α se incrementa y las curvas van a tener ese pandeo característico que indica que la distribución no es uniforme.

En los casos en los que no se ha puesto el último condensador a tierra, se ha observado que no finalizan en 0, sino que, en una tensión flotante, al no estar referenciados a tierra.

Conclusiones

En este trabajo he descrito el estudio, la caracterización e implementación de los pasos y elementos necesarios para desarrollar un circuito equivalente que permite estudiar el fenómeno capacitivo en un transformador.

En el capítulo 2 se ha hecho un breve recorrido por el marco teórico de las ecuaciones que permiten conocer el comportamiento capacitivo de un transformador, así como la importancia de los coeficientes de capacidad y de inducción.

En el capítulo 3 se ha explicado cómo es posible construir un modelo circuital a partir de los fenómenos capacitivos que ocurren entre los conductores, bien de un transformador o de una bobina. He ilustrado con un ejemplo de cuatro conductores o espiras para ilustrar los procesos seguidos durante la realización del trabajo.

He explicado la necesidad de utilizar programas que resuelven problemas mediante Métodos de Elementos Finitos y se ha justificado él porqué han sido utilizados dos programas distintos para calcular los coeficientes de capacidad y de inducción. He observado que los valores obtenidos son similares.

He calculado el valor de los condensadores equivalentes a partir de los coeficientes de capacidad e inducción, lo que me permite crear un circuito en forma de red capacitiva automáticamente en función del número de conductores que consideremos.

Con estos resultados, he construido circuitos equivalentes obtenidos para calcular la capacidad equivalente y distribución de tensiones con programas de circuitos.

He comprendido cómo pasar de un problema perteneciente a un fenómeno físico difícil de resolver analíticamente, a un modelo que se puede resolver más fácilmente con las herramientas adecuadas; en definitiva, ese es el trabajo de un ingeniero bajo mi punto de vista.

6.1 Vías futuras

Las posibles vías futuras para el proyecto que aquí se ha presentado pueden ser:

En un primer lugar, podría continuarse la idea inicial del proyecto, estudiando los fenómenos de inducción magnética y los fenómenos disipativos en un pequeño transformador para aplicaciones de electrónica de potencia, para posteriormente de forma similar a como se ha realizado este proyecto, poder implementar un circuito equivalente con bobinas y resistencias que junto a la parte capacitiva que aquí se ha presentado, pueda formar un modelo completo del transformador. Esto nos permitiría estudiar transformadores de electrónica utilizados en convertidores de alta frecuencia de conmutación y observar la viabilidad de miniaturizar todavía más este tipo de componentes.

Por otro lado, podría realizarse un modelo 2D (o incluso 3D) de un transformador de distribución eléctrica con diversas tecnologías como discos o bandas en lugar de bobinados, para posteriormente estudiar e implementar también los fenómenos de inducción magnética y disipativos, ya que para estos fenómenos el tamaño del transformador sí que influye en la parametrización de los mismos. De esta forma se permitiría poder conocer mejor el comportamiento de un transformador de este tipo, pudiendo mejorar las técnicas de fabricación de los mismos.

Además, se podría modificar el código de Matlab para que fuese capaz de considerar las autocapacidades en el modelo de LTSpice.

Por último, hay que destacar que se podría utilizar el mismo código que ya se ha implementado para futuros modelos, solamente habría que modificar unos pocos parámetros. Esto mismo me hace pensar en la posibilidad de mejorar el código, de forma que pudiera crearse una especie de software que permitiera calcular estos fenómenos de forma más general y sencilla.

BIBLIOGRAFÍA

[1] Ceotto, E. “Modélisation des effets inductifs et capacitifs dans les transformateurs à haute fréquence”. Trabajo de fin d’études, Faculté des Sciences Appliquées de l’Université de Liège. [último acceso: 6 de septiembre de 2019]. Disponible en: <https://matheo.uliege.be/handle/2268.2/1403>

[2] Usón-Sardaña, A. “Aplicación del método de elementos finitos al perfeccionamiento de pinzas amperimétricas”. Tesis doctoral, Departamento de Ingeniería Eléctrica de la Universidad de Zaragoza. [último acceso: 6 de septiembre de 2019]. Disponible en: https://zaguan.unizar.es/record/1912/files/TUZ_0026_uson_finitos.pdf

[3] Roy Matín, C. “Estudio de la distribución de tensión frente a impulso tipo rayo en un transformador seco de potencia”. Proyecto fin de carrera, Departamento de Ingeniería Eléctrica de la Escuela de Ingeniería y Arquitectura de Zaragoza. [último acceso: 6 de septiembre de 2019]. Disponible en: <https://zaguan.unizar.es/record/14329/files/TAZ-PFC-2014-235.pdf>

[4] Wiki del Dpto. de Física Aplicada III en la ETSI de la Universidad de Sevilla. [último acceso: 6 de septiembre de 2019]. Disponible en: http://laplace.us.es/wiki/index.php/P%C3%A1gina_Principal
http://laplace.us.es/wiki/index.php/Coeficientes_de_capacidad

[5] Meeker, D. “Finite Element Method Magnetics User’s Manual version 4.2”. [último acceso: 6 de septiembre de 2019]. Disponible en: <http://www.femm.info/wiki/Documentation/>

[6] Bernal Méndez, J. “Tema 5: Conductores en equilibrio electrostático”. Apuntes asignatura Campos Electromagnéticos de Ingeniería Industrial de la Universidad de Sevilla, Dpto. de Física Aplicada III. [último acceso: 6 de septiembre de 2019]. Disponible en: <http://www.esi2.us.es/DFA/CEMI/Teoria/Teoria.htm>

[7] Usón Sardaña, A. y Letosa Fleta, J. “Unidad 1: Electrostática”. Apuntes de asignatura Física II de Grado en ingeniería eléctrica de la Universidad de Zaragoza. [último acceso: 6 de septiembre de 2019].

- [8] Tipler, P.A., Mosca, G. "Física para la ciencia y la tecnología". Volumen 2. 6ª Ed. Editorial Reverté. 2010.
- [9] Reitz, J.R., Milford, F.J., Christy, R.W. "Fundamentos de la teoría electromagnética". 4ª Ed. Addison-Wesley Iberoamericana. 1996.
- [10] Página web GetDP. [último acceso: 7 de septiembre de 2019]. Disponible en: <http://www.getdp.info/>
- [11] Página web de FEMM. [último acceso: 7 de septiembre de 2019]. Disponible en: <http://www.femm.info/wiki/HomePage>
- [12] Página web de Gmsh. [último acceso: 7 de septiembre de 2019]. Disponible en: <http://gmsh.info/>
- [13] Página web de Matlab. [último acceso: 7 de septiembre de 2019]. Disponible en: <https://es.mathworks.com/products/matlab.html>
- [14] Página web de LTSpice. [último acceso: 7 de septiembre de 2019]. Disponible en: <https://www.analog.com/en/design-center/design-tools-and-calculators/ltspice-simulator.html>
- [15] Friedel, S. "How to Calculate a Capacitance Matrix in COMSOL Multiphysics®". [último acceso: 7 de septiembre de 2019]. Disponible en: <https://www.comsol.com/blogs/how-to-calculate-a-capacitance-matrix-in-comsol-multiphysics/>

Códigos implementados por el autor

Código archivo “Transformer_613783664.pro”

Código implementado directamente en bloc de notas (.txt) el cual contiene las órdenes para que funcione el modelo de transformador en el software GetDP.

Las ordenes implementadas son la asignación de regiones a cada parte del dibujo (aislante, núcleo, aire, infinito, conductores). Además, indica el número de conductores a considerar en el modelo y la elección del modelo que estará a 1 Voltio para que el resto estén a 0 Voltios, realizando de esta forma el test que nos da las capacidades entre cables del caso seleccionado.

```
Group{
    //Surfaces in the mesh
    insulator = Region[{9000000,9000001,9000002}];
    core = Region[{9000003,9000004}];
    air = Region[{9000005,9000006,9000007}];
    infity = Region[{9000009}];
    //Boundary of conductors (lines) in the mesh
    For i In {1:16}
        cond~{i} = Region[ ( 1000020 + (i-1) * 13 ) ];
        cond_inside~{i} = Region[ ( 1000022 + (i-1) * 13 ) ];
    EndFor

    For i In {17:32}
        cond~{i} = Region[ ( 1000238 + (i-17) * 13 ) ];
        cond_inside~{i} = Region[ ( 1000240 + (i-17) * 13 ) ];
    EndFor

    //Symmetry line
    symmetry = Region[{9000008}];

    //Number of conductors to consider in the model
    DefineConstant[ numConductors = 3 ];
```

```

//Num of the conductor that should to 1V (the others are set to
ground)
DefineConstant[ oneVoltConductor = 1 ];

//Boundary of conductors where we apply the potential
SkinDomainC_Ele = Region[{}];

For i In {1:numConductors}
    SkinDomainC_Ele += Region[{cond~{i}}];
EndFor
    For i In {numConductors + 1 :32}
        air += Region[{cond_inside~{i}}];
    EndFor

//Generic region
DomainCC_Ele = Region[{insulator,core,air}];
}

Function {
    epsr[air] = 1;
    epsr[core] = 1;
    epsr[insulator] = 4;
}

Constraint {
    { Name ElectricScalarPotential ; Type Assign ;
        Case {
            { Region infity ; Value 0 ;}
        }
    }

    { Name GlobalElectricPotential ; Type Assign ;
        Case {
            { Region cond~{oneVoltConductor} ; Value 1 ; }
            For i In {1:32}
                If (i != oneVoltConductor)
                    Region cond~{i} ; Value 0 ;
                }
            EndIf
        }
        EndFor
    }
}

```

```

    { Name GlobalElectricCharge ; Type Assign ;
      Case {
        }
      }
}

Include "Integration_Lib.pro";
Include "Jacobian_Lib.pro";
Include "EleSta_v.pro";

PostOperation {
  { Name v ; NameOfPostProcessing EleSta_v;
    Operation {
      Print [ v, OnElementsOf DomainCC_Ele, File
        "capa_v_v.pos" ] ;

      //Print [ energy, OnRegion DomainCC_Ele ];
      Print [ Q, OnRegion SkinDomainC_Ele, Format Table ,
        File "charges.txt"];
      Print [ V, OnRegion SkinDomainC_Ele, Format Table ,
        File "voltages.txt"];
      Print [ C, OnRegion SkinDomainC_Ele, Format Table ,
        File "capacitances.txt"];
    }
  }
}

```

Código archivo “Fuente.m”

```
function [ coordW ] = Cable( Xi, Yi, Xf, Yf )
%% Cable
% Función que imprime en el archivo .asc la línea necesaria para
dibujar un cable en LTSPICE
global f;

% Dibuja un cable uniendo dos puntos según coordenadas iniciales y
finales dadas
coordW = fprintf( f, 'WIRE %d %d %d %d \n', Xi, Yi, Xf, Yf );

end
```

Código archivo “Cable.m”

```
function [ coordW ] = Cable( Xi, Yi, Xf, Yf )
%% Cable
% Función que imprime en el archivo .asc la línea necesaria para
dibujar un cable en LTSPICE
global f;

% Dibuja un cable uniendo dos puntos según coordenadas iniciales y
finales dadas
coordW = fprintf( f, 'WIRE %d %d %d %d \n', Xi, Yi, Xf, Yf );

end
```

Código archivo “Condensador.m”

```
function [ coordC, sig, w1, name, value, w2 ] = Condensador( Cx, Cy,
Valor, Nombre1, Nombre2, Nombre3, VLabelPos )
%% Condensador
% Función que imprime en el archivo .asc las líneas necesarias
para dibujar un condensador en LTSPICE
global f;

% Coloca el simbolo del condensador en las coordenadas dadas
coordC = fprintf( f, 'SYMBOL cap %d %d R0\n', Cx, Cy );

% Coloca la etiqueta que identifica al condensador. pe: C12
sig = fprintf( f, 'SYMATTR InstName %s%d%d \n', Nombre1, Nombre2,
Nombre3 );

% Posiciona la etiqueta que identifica al condensador según
coordenadas
w1 = fprintf( f, 'WINDOW 0 9 95 Left 2\n' );

% Coloca la etiqueta del valor del condensador
name = fprintf( f, 'SYMATTR Value ' );

% Asigna un valor en Faradios al condensador
value = fprintf( f, '%e\n', Valor );
```



```
% Posiciona la etiqueta valor del condensador según coordenadas
w2 = fprintf( f, 'WINDOW 3 23 %d Left 2\n', VLabelPos );
```

```
end
```

Código archivo “Computecapacitances_function.m”

```
function [ CapacitanceMatrix ] = Computecapacitances_function( numCond
)
%% Obtención matriz de coeficientes
% Función que accede al software de Elementos Finitos y recoge los
voltajes y cargas obtenidos de la simulación para a partir de ellos
calcular la matriz de coeficientes

% Acceso a los archivos del software GetDP
gmsh='C:\Compute_capacitances_GetDP\gmsh.exe ';
getdp='C:\Compute_capacitances_GetDP\getdp.exe ';
geo='C:\Compute_capacitances_GetDP\Transformer_613783664.geo';
pro='C:\Compute_capacitances_GetDP\Transformer_613783664.pro';
vfile='C:\Compute_capacitances_GetDP\voltages.txt';
qfile='C:\Compute_capacitances_GetDP\charges.txt';
cfile='C:\Compute_capacitances_GetDP\capacitances.txt';

% Modifica el número de conductores en el archivo
"Transformer_613783664.pro" para poner iterativamente uno a 1V y el
resto a 0V
test1=' -setnumber numConductors ';
numCondn=str2num(numCond);
V1=[];

for i = 1 : numCondn
    % Convierte el índice del bucle en string para pasárselo a
    GetDP
    n=num2str(i);
    test2=' -setnumber oneVoltConductor ';

    % Creamos string que le pasaremos a GetDP
    test=strcat(test1,32,numCond,32,test2,32,n);

    % Le pasamos todo el string a GetDP para que itere los test
    (1V-0V)
    system([getdp pro test ' -v 3 -solve EleSta_v -pos v']);

    % Carga iterativamente el fichero de tensiones obtenido de
    GetDP
    load(vfile)
    load(qfile)

    % Introduce en una matriz los valores de tensiones obtenidos
    del fichero voltages.txt que proporciona GetDP
    volt = voltages;
    volt(1) = [];

    % Introduce en una matriz los valores de cargas obtenidos del
    fichero charges.txt que proporciona GetDP
```

```

    charg = charges;
    charg(1) = [];

    % Recorremos las matrices de cargas y voltages, generando una
    nueva
    % matriz,C, que es la correspondiente a las capacidades debido
    a la división entre ambos valores.
    for j = 1: numCond
        C(i,j) = charg(j)/volt(i);
    end
end

% Realizamos los calculos para obtener la MATRIZ
C;
C0 = C;

% Cogemos la diagonal pcpal y realizamos la suma de los elementos
de la fila correspondiente (en realidad resta, porque tienen signo
"-")
D = diag(sum(C,1));

for a=1 : numCond

% Iteramos recorriendo la Matriz de Maxwell para dejar la diagonal
pcpal a cero y cambiar el signo de los elementos Cij
    for b = 1 : numCond
        C0(b,b) = [0];
    end
end

C0pos = -1*C0;

% Sumamos la diagonal pcpal que contiene las sumas (restas) de las
filas en cada elemento con la matriz a la que hemos eliminado la
diagonal y cambiado de signo a los elementos Cij
CapacitanceMatrix = D+C0pos

end

```

Código archivo "Computecapacitances_LTSPICE.m"

```

%% Este script crea un modelo capacitivo .asc de LTSPICE de
capacidades entre conductores.
%% Puede obtenerse la capacidad desde entre 2 conductores hasta entre
32
%% conductores, según el modelo físico sobre el que trabaja.

clear all;
clc;
global f;

% Entrada por teclado de los conductores a considerar en modelo entre
los cuales va a obtenerse su capacidad
numCond=input('Número de conductores a considerar en el modelo (2-32):
','s');
numCond=str2num(numCond);

```

```

% Llamada a la función "Computecapacitances_function( numCond )"
CapacitanceMatrix=Computecapacitances_function( numCond );

% Abrimos el fichero .asc que contendrá el circuito en LTSPICE
f = fopen( 'CircuitoEquivalente.asc', 'wt' );
fprintf( f, 'Version 4\n' );
fprintf( f, 'SHEET 1 880 680\n' );

% Llamada a la función "Fuente()"
Fuente(); % Fuente de tensión:

% Une la fuente con los condensadores
Cable( 0, 144, 80, 144 ); % Primer cable:

% Ground
fprintf( f, 'FLAG 80 208 0 0\n' );

% Variable que indica el nº de condensadores dependiendo de los
conductores que entren en juego
numberCapa = (numCond*(numCond-1))/2; %Resto de condensadores:

% Vectores para la localización de los condensadores según sus
coordenadas
CoordCn(1:numberCapa)= 64;
CoordCnn=cumsum(CoordCn);

% Matriz Combinación binomial utilizada como índice de los diferentes
condensadores
combos = nchoosek(1:numCond,2);

for b = 1 : length (CoordCnn)
    % Obtenemos reiteradamente la primera fila de la Matriz
    Combinación binomial
    Comb = combos (b, :);

    % Llamamos a la función "Condensador( Cx, Cy, Valor, Nombre1,
    Nombre2, Nombre3, VLabelPos )"
    Condensador(
    CoordCnn(b),144,CapacitanceMatrix(Comb(1),Comb(2)), 'C',Comb(1),Comb(2)
    , -CoordCnn(b)+32);

    %la cual nos pintará los condensadores involucrados en función de
    la variable "numberCapa"
    for c = 1 : length (CoordCnn)
        for d = 1 : numCond
            if c == Comb(1) && c == d

                % Condición para dibujar un cable que vaya desde el
                extremo superior del condensador hasta el punto que le
                corresponde, para unirse con el resto de condensadores
                Cable( CoordCnn(b)+16, 144, CoordCnn(c)+255, -448 );
            end

            if c == Comb(2) && c == d

                % Condición igual a la anterior, pero esta vez para el
                extremo inferior del condensador
                Cable( CoordCnn(b)+16, 208, CoordCnn(c)+255, -448 );
            end
        end
    end
end

```

```

        end
    end
end

```

```

% Cerramos el fichero .asc que ya ha generado el circuito en LTSPICE
% preparado este se llama "CircuitoEquivalente.asc"
fclose( f );
fclose all;

```

Código archivo "Modelo_Trafo_Automatizado.m"

```

function [ ] = Modelo_Trafo_Automatizado( numCond, OCV )
%% Funcion que automatiza Femm mediante comandos de lenguaje LUA

% Abrimos librería para poder trabajar con Femm desde Matlab
addpath('C:\femm42\mfiles');
savepath;

% Abrimos Femm
openfemm;
%Abrimos el modelo que hemos creado
opendocument('Modelo_Trafo.FEE')
%Maximizamos la ventana para mejor visualización
main_maximize;

% Ocultamos la rejilla de Femm
ei_hidegrid;
% Creamos el mallado
ei_createmesh;

%% Número de conductores a considerar en el modelo (2-32)
numConductorn = str2num(numCond);

%ei_seteditmode('group');

% Selecciona el conductor 1, 2, 3, ... hasta el N.º de conductores a
considerar en el modelo
for i = 1 : numConductorn
    iletra = num2str(i);
    icadena = strcat('Conductor',iletra);
    ei_selectgroup(i);
    ei_setblockprop('<No Mesh>', 1,1,i);
    ei_setarcsegmentprop(5, '<None>', 0, i, icadena)
    ei_clearselected;

end

% %Selecciona los conductores restantes para no considerarlos en el
modelo
for j = numConductorn + 1 : 32
    jletra = num2str(j);
    jcadena = strcat('Conductor',jletra);
    ei_selectgroup(j);
    ei_setblockprop('Aire', 1,1,j);

```

```

    ei_setarcsegmentprop(5, '<None>', 0, j, '<None>')
    ei_clearselected;
end

% Numero del conductor que estará a 1V (El resto estará a tierra)
oneVoltConductorn = OCV;
% Convertimos el número del conductor a string
oneVoltConductorl1 = num2str(oneVoltConductorn);
% Unimos el string para pasarselo a femm como exige su formato
oneVoltConductor=strcat('Conductor',oneVoltConductorl1);
% Primer 1 significa que modificamos el Vc del Conductor seleccionado.
El segundo 1 es = 1V
ei_modifyconductorprop(oneVoltConductor,1,1);

% Iteramos para poner a 0V el resto de conductores, también
convirtiendo
% todo a string que es como femm lo requiere
for k = 1 : 32
    if k ~= oneVoltConductorn
        oneVoltConductorl2 = num2str(k);
        oneVoltConductor2=strcat('Conductor',oneVoltConductorl2);

        ei_modifyconductorprop(oneVoltConductor2,1,0);
    end
end

% Lanzamos el análisis y cargamos la solución del análisis realizado
ei_analyze(0);
ei_loadsolution()

% Abrimos dos ficheros .txt para guardar las cargas y diferencias de
potencial que nos calcula femm para cada conductor
f1 = fopen( 'voltages.txt', 'wt' );
f2 = fopen( 'charges.txt', 'wt' );

% Iteramos para ir guardando los datos en el fichero
for p = 1:numConductorn
    numConductorl2 = num2str(p);
    numConductor=strcat('Conductor',numConductorl2);
    % Nos da las propiedades del conductor, es decir la ddp y la carga
    Prop = eo_getconductorproperties(numConductor);
    fprintf( f1,'%0f ', Prop(1));
    fprintf( f2,'%d ', Prop(2));
end

% Cerramos los ficheros .txt que han almacenado los datos
fclose( f1 );
fclose( f2 );

% Cerramos Femm
closefemm;

```

Código archivo "Computecapacitances_function_FEMM.m"

```
function [ CapacitanceMatrix ] = Computecapacitances_function_FEMM(
numCond )
%% Obtención matriz de coeficientes
% Función que accede al software de Elementos Finitos y recoge los
voltages y cargas obtenidos de la simulación para a partir de ellos
calcular la matriz de coeficientes

% Acceso a los archivos del software GetDP
modelo='C:\Compute_capacitances_Femm\Modelo_Trafo_Automatizado.m';
vfile='C:\Compute_capacitances_Femm\voltages.txt';
qfile='C:\Compute_capacitances_Femm\charges.txt';

% Modifica el número de conductores en el archivo
"Modelo_Trafo_Automatizado.m" para poner iterativamente uno a 1V y
el resto a 0V
numCondn=str2num(numCond);
V1=[];

for i = 1 : numCondn
    % Convierte el índice del bucle en string para pasarselo a
    Femm
    n=num2str(i);

    % Le pasamos todo el string a Femm para que itere los test
    (1V-0V)
    %system([getdp pro test ' -v 3 -solve EleSta_v -pos v']);
    Modelo_Trafo_Automatizado( numCond, i );

    % Carga iterativamente el fichero de tensiones obtenido de
    Femm
    load(vfile)
    load(qfile)

    % Introduce en una matriz los valores de tensiones
    obtenidos del fichero voltages.txt que proporciona GetDP
    volt = voltages;

    % Introduce en una matriz los valores de cargas obtenidos
    del fichero charges.txt que proporciona GetDP
    charg = charges;

    % Recorremos las matrices de cargas y voltages, generando una
    nueva matriz,C, que es la correspondiente a las capacidades
    debido a la división entre ambos valores.
    for j = 1: numCondn
        C(i,j) = charg(j)/volt(i);
    end
end

% Realizamos los calculos para obtener la MATRIZ
C;
C0 = C;

% Cogemos la diagonal pcpal y realizamos la suma de los elementos
de la fila correspondiente (en relaidad resta, porque tienen signo
"-")
D = diag(sum(C,1));
```

```

for a=1 : numCondn

    % Iteramos recorriendo la Matriz de para dejar la diagonal
    pcpal a cero y cambiar el signo de los elementos Cij
    for b = 1 : numCondn
        C0(b,b) = [0];
    end
end

C0pos = -1*C0;

% Sumamos la diagonal pcpal que contiene las sumas (restas) de las
filas en cada elemento con la matriz a la que hemos eliminado la
diagonal y cambiado de signo a los elementos Cij
CapacitanceMatrix = D+C0pos

end

```

Código archivo "Computecapacitances_LTSPICE_FEMM.m"

```

%% FEMM2D
%% Este script crea un modelo capacitivo .asc de LTSPICE de
capacidades entre conductores.
%% Puede obtenerse la capacidad desde entre 2 conductores hasta entre
32
%% conductores, según el modelo físico sobre el que trabaja.

clear all;
clc;
global f;

% Entrada por teclado de los conductores a considerar en modelo entre
los cuales va a obtenerse su capacidad
numCond=input('Número de conductores a considerar en el modelo (2-32):
','s');
numCondn=str2num(numCond);

% Llamada a la función "Computecapacitances_function( numCond )"
CapacitanceMatrix=Computecapacitances_function_FEMM( numCond );

% Abrimos el fichero .asc que contendrá el circuito en LTSPICE
f = fopen( 'CircuitoEquivalente.asc', 'wt' );
fprintf( f, 'Version 4\n' );
fprintf( f, 'SHEET 1 880 680\n' );

% Llamada a la función "Fuente()"
Fuente(); % Fuente de tensión:

% Une la fuente con los condensadores
Cable( 0, 144, 80, 144 ); % Primer cable:

% Ground
fprintf( f, 'FLAG 80 208 0 0\n' );

```

```

% Variable que indica el nº de condensadores dependiendo de los
conductores que entren en juego
numberCapa = (numCondñ*(numCondñ-1))/2;      %Resto de condensadores:

% Vectores para la localización de los condensadores según sus
coordenadas
CoordCñ(1:numberCapa)= 64;
CoordCññ=cumsum(CoordCñ);

% Matriz Combinación binomial utilizada como índice de los diferentes
condensadores
combos = nchoosek(1:numCondñ,2);

for b = 1 : length (CoordCññ)
    % Obtenemos reiteradamente la primera fila de la Matriz
    Combinación binomial
    Comb = combos(b,:);

    % Llamamos a la función "Condensador( Cx, Cy, Valor, Nombre1,
    Nombre2, Nombre3, VLabelPos )"
    Condensador(
    CoordCññ(b),144,CapacitanceMatrix(Comb(1),Comb(2)), 'C',Comb(1),Comb(2)
    , -CoordCññ(b)+32);

    %la cual nos pintará los condensadores involucrados en función de
    la variable "numberCapa"
    for c = 1 : length (CoordCññ)
        for d = 1 : numCondñ
            if      c == Comb(1) && c == d

                % Condición para dibujar un cable que vaya desde el
                extremo superior del condensador hasta el punto que le
                corresponde, para unirse con el resto de condensadores
                Cable( CoordCññ(b)+16, 144, CoordCññ(c)+255, -448 );
            end

            if c == Comb(2) && c == d

                % Condición igual a la anterior, pero esta vez para el
                extremo inferior del condensador
                Cable( CoordCññ(b)+16, 208, CoordCññ(c)+255, -448 );
            end
        end
    end
end

% Cerramos el fichero .asc que ya ha generado el circuito en LTSPICE
% preparado este se llama "CircuitoEquivalente.asc"
fclose( f );
fclose all;

```