

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

DEPARTAMENTO DE INGENIERÍA MECÁNICA Y DE MATERIALES



TRABAJO FIN DE MÁSTER EN INGENIERÍA MECÁNICA

*Evaluación a priori del nivel de sujeción de
implantes protésicos mediante análisis de
vibraciones con simulaciones 'patient specific'
realizadas con el Cartesian grid Finite Element
Method (cgFEM)*

Presentado por: D. Cristian Estévez Díaz

Dirigido por: Dr. D. Juan José Ródenas García

Codirigido por: D. Luca Giovannelli

Valencia, septiembre de 2017

Agradecimientos

A *Juanjo* y a *Luca* por darme la oportunidad de trabajar con ellos, por enseñarme tantas cosas y por las horas de dedicación que nos brindan a mí y al resto de alumnos.

Al resto de *profesores* del Máster en Ingeniería Mecánica por todo lo que he aprendido y por su tiempo.

A *mi familia* y especialmente a *Raquel*, por el apoyo incondicional y por ayudarme a llegar hasta aquí.

¡Muchas gracias a todos!

ABSTRACT

The *Department of Mechanical and Materials Engineering* of the *University Polytechnic of Valencia* has developed some efficient *Finite Element* codes in *Matlab* based on the use of geometry independent Cartesian grids (*FEAVox*). One of its main numerical features is that the code is able to perform 'patient specific' analyzes from medical images.

This project is focused on using *FEAVox* in order to simulate the effect of future surgical operations involving prosthetic implants in a 'patient specific' framework, with the main objective of estimating the quality of the connection between bone and object, once it is placed in the patient. This process aims to provide knowledge of the adequacy of the implant beforehand, bearing in mind the needs and the success of future surgical operations of the patient.

FEAVox is able to create a finite elements model that represents accurately the bone features near the implant, whose position requirements involve a previous pixelization process and the development of a new *Matlab* program that allows the user to precisely move and position the object.

The use of the techniques shown in this document will allow the development of a new validation process for the specific operation needs in a certain patient, based on the evaluation of changes in the natural frequencies and vibration modes according to the melding level of the bone-implant system. For this reason, a dynamic analysis code will be incorporated to *FEAVox*. Its results will be used as quality indicators.

RESUMEN

El *Departamento de Ingeniería Mecánica y Materiales* de la *Universidad Politécnica de Valencia* se encuentra en pleno desarrollo de un código de Elementos Finitos 3D (*FEAVox*), programado en *Matlab*, y cuya característica más notable reside en el empleo de mallados cartesianos independientes de la geometría. Entre sus muchas posibilidades numéricas se encuentra la de realizar análisis específicos de paciente a partir de imágenes médicas.

En este Trabajo Fin de Máster se pretende utilizar *FEAVox* para simular el efecto de futuras operaciones quirúrgicas que involucren la colocación de implantes protésicos en un entorno 'patient specific', con el objetivo de determinar el nivel de sujeción esperado de la prótesis una vez colocada en el paciente, con la meta última de estimar, a priori, la adecuación del implante a las características del enfermo y el nivel de éxito de la operación.

A partir de imágenes médicas (TAC o resonancia magnética), *FEAVox* es capaz de obtener un modelo de elementos finitos que representa con gran exactitud las características de la estructura ósea en las inmediaciones del implante, cuya ubicación requiere de su previa pixelización y del desarrollo de una nueva herramienta, también en lenguaje *Matlab*, que permita mover y posicionar el objeto con cierta precisión.

La utilización de las técnicas que aquí se desarrollan permitirá el desarrollo de un procedimiento novedoso de valoración de la necesidad específica de operación por parte de un paciente, pues se ha comprobado que las frecuencias naturales y modos de vibración del conjunto hueso-implante varían notablemente en función del nivel de sujeción. Por este motivo, y como parte del procedimiento definido en este documento, se dotará a *FEAVox* con la posibilidad de realizar análisis dinámicos, cuyos resultados serán utilizados como indicadores de la sujeción de la unión.

RESUM

El Departament d'Enginyeria Mecànica i de Materials de la Universitat Politècnica de València es troba en ple desenvolupament d'un codi d'Elements Finites 3D (FEAVox), programat en Matlab. La característica més notable d'aquest codi resideix en la utilització de mallats cartesianes independents de la geometria. Entre les seves moltes possibilitats es troba la de realitzar anàlisis específiques de pacient a partir d'imatges mèdiques.

En aquest Treball Fi de Màster es pretén utilitzar FEAVox per simular l'efecte de futures operacions quirúrgiques que involucren la col·locació d'implants protètics en un entorn 'patient specific'. Amb això es pretén determinar el nivell de subjecció esperat de la pròtesi una vegada col·locada en el pacient, amb la finalitat d'estimar, a priori, l'adequació de l'implant a les característiques del malalt i el nivell d'èxit de l'operació.

A partir d'imatges mèdiques (TAC o ressonància magnètica), FEAVox és capaç d'obtenir un model d'elements finits que representa amb el suficient nivell d'aproximació les característiques de l'estructura òssia en els voltants de l'implant. La ubicació del implant requereix de la prèvia pixelització i del desenvolupament de una nova eina, també en llenguatge Matlab, que permeti moure i posicionar l'objecte amb certa precisió.

La utilització de les tècniques implementades permetrà el desenvolupament d'un procediment nou de valoració de la qualitat de subjecció de la pròtesi per a un determinat pacient, ja que s'ha comprovat que les freqüències naturals i modes de vibració del conjunt os-implant varien notablement en funció de l'esmentat nivell de subjecció. Per aquest motiu, i com a part del procediment definit en aquest document, es dotarà FEAVox amb la possibilitat de realitzar anàlisis modals, els resultats dels quals seran utilitzats com a indicadors de la fixació de la unió.

ÍNDICE

1. Objetivo.....	1
2. Estado del Arte.....	1
3. Punto de Partida.....	3
4. Procedimiento.....	4
5. Introducción.....	5
5.1. ¿Qué es un TAC?.....	5
5.2. ¿Qué es una prótesis?.....	6
5.3. Materiales en Prótesis Internas.....	7
5.4. El Formato <i>DICOM</i>	8
5.4.1. <i>Ejes de Referencia</i>	10
5.4.2. <i>Planos de Corte</i>	13
5.4.3. <i>Información de Interés en Ficheros DICOM</i>	15
5.5. Fundamentos del Método de los Elementos Finitos.....	16
5.5.1. <i>Elementos Finitos en Dinámica y Vibraciones</i>	19
5.6. Análisis Modal.....	21
5.6.1. <i>Clasificación de la vibración</i>	22
5.6.2. <i>Sistemas de N Grados de Libertad</i>	23
5.6.3. <i>Vibración Libre de Sistemas de N G.D.L. No Amortiguados</i>	23
5.7. La Escala Hounsfield.....	24
6. GiD. Software de Pre/Post Procesado.....	26
6.1. ¿Por qué GiD?.....	28
6.2. Nociones Básicas de GiD.....	29
6.3. Obtención de una Malla Cartesiana Tridimensional en GiD.....	33
6.3.1. <i>Opciones de Mallado</i>	35

6.3.2.	<i>Importación de la Geometría.....</i>	36
6.3.3.	<i>Generación de Malla.....</i>	37
6.3.4.	<i>Evaluación de la Calidad de la Malla.....</i>	39
6.4.	Problemas Encontrados en GiD.....	40
6.4.1.	<i>Geometrías Mal Definidas.....</i>	40
6.4.2.	<i>Mallado Cartesiano con Defectos de Forma.....</i>	42
6.4.3.	<i>Error por Tamaño de Elemento Mínimo.....</i>	43
6.4.4.	<i>Errores en la Numeración de Elementos y Nodos.....</i>	44
6.5.	Información a Exportar desde GiD.....	45
6.6.	Comunicación GiD-Matlab.....	46
6.6.1.	<i>Estructura del archivo de malla.....</i>	46
6.7.	Cambio de Variables en GiD.....	48
6.7.1.	<i>Renumerar Elementos y Nodos en GiD.....</i>	49
6.8.	Giros de Geometría en GiD.....	51
7.	Programación en Matlab.....	52
7.1.	Implant Viewer.....	54
7.1.1.	<i>Consideraciones Previas.....</i>	55
7.1.2.	<i>Principio de Funcionamiento de Implant Viewer.....</i>	56
7.1.3.	<i>Matrices de Datos Principales.....</i>	59
7.1.4.	<i>Lectura de Archivos.....</i>	62
7.1.5.	<i>Movimientos del Implante.....</i>	65
7.1.6.	<i>Problemas Derivados del Movimiento.....</i>	67
7.1.7.	<i>Tratamiento de Imágenes en Implant Viewer.....</i>	69
7.1.8.	<i>Renderizado 3D. Marching Cubes.....</i>	72
7.1.9.	<i>Ángulos de Euler y la Orientación del Implante.....</i>	75
7.2.	FEAVox.....	78
7.2.1.	<i>Estructura del Programa.....</i>	79
7.2.2.	<i>Ejecución de FEAVox.....</i>	81

7.2.3.	<i>Estructura de Variables en FEAVox.....</i>	82
7.2.4.	<i>Definición de un Análisis Modal.....</i>	83
7.2.5.	<i>Incorporación del Análisis Modal al código de FEAVox.....</i>	87
7.2.6.	<i>Resultados y Representación de Modos de Vibración.....</i>	90
8.	Ejemplos.....	91
8.1.	Implante Dental.....	92
8.1.1.	<i>Propiedades del Material asociadas a la Imagen.....</i>	93
8.1.2.	<i>Discretización y Ubicación del Implante.....</i>	94
8.1.3.	<i>Análisis Modal del Implante.....</i>	95
8.1.4.	<i>Análisis de la Unión Implante-Hueso.....</i>	99
8.1.5.	<i>Comparación de Resultados.....</i>	102
8.1.6.	<i>Análisis de Sensibilidades.....</i>	104
8.2.	Implante de Cadera.....	106
8.2.1.	<i>Propiedades del Material asociadas a la Imagen.....</i>	107
8.2.2.	<i>Discretización y Ubicación del Implante.....</i>	108
8.2.3.	<i>Análisis Modal del Implante.....</i>	109
8.2.4.	<i>Análisis de la Unión Implante-Hueso.....</i>	112
8.2.5.	<i>Comparación de Resultados.....</i>	115
8.2.6.	<i>Análisis de Sensibilidades.....</i>	117
9.	Conclusiones.....	118
10.	Líneas de Mejora.....	119
11.	Bibliografía.....	121
	ANEXO I. Manual de Usuario de <i>Implant Viewer</i>.....	1

1. Objetivo.

El presente Trabajo Fin de Máster (*TFM*) pretende ofrecer una solución al problema de la valoración de la viabilidad de los implantes en pacientes a través del estudio de las frecuencias naturales del conjunto. El objetivo principal se identifica con la programación de la unión implante-hueso, así como desarrollar para FEAVox un algoritmo de estimación de frecuencias naturales. Más concretamente, las metas que persigue el trabajo se identifican en los siguientes ítems.

- Modelado del implante con una malla cartesiana.
- Obtención de la imagen médica del paciente.
- Colocación del implante sobre la imagen médica del paciente.
- Implementación del análisis de frecuencias naturales y modos de vibración
- Resolución de ejemplos.
- Redacción y presentación de resultados.

La consecución individual de los pasos mencionados permitirá alcanzar el propósito principal de este trabajo: el desarrollo de un procedimiento fiable que permita determinar si un implante se fijará al hueso del paciente adecuadamente de manera previa a la realización de una operación quirúrgica, de tal manera que el especialista pueda conocer de antemano esa información y actuar consecuentemente.

2. Estado del Arte.

Los avances tecnológicos de los últimos años han posibilitado notables avances en el desarrollo de nuevas técnicas en el análisis por elementos finitos. Tanto es así que

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

actualmente se están perfeccionando técnicas de elementos finitos en tiempo real, lo que tiene importantes aplicaciones en el campo de la medicina.

El *Departamento de Ingeniería Mecánica* de la *Universidad Politécnica de Valencia* se encuentra actualmente en desarrollo de un software propio de análisis por elementos finitos con lenguaje Matlab (*FEAVox*) que, entre sus muchas características, permite llevar a cabo análisis numéricos de imágenes en función del color individual de los píxeles que la conforman, teniendo en cuenta el valor en escala *Hounsfield* del píxel para determinar las propiedades elásticas locales.

En este sentido, tiene especial interés analizar la interacción prótesis-hueso de manera previa a su implantación en un determinado paciente, ya que ofrece grandes ventajas desde el punto de vista médico (como por ejemplo la posibilidad de valorar la viabilidad de una operación quirúrgica). Es por esto que el TFM que ocupa este documento trata de proponer una primera implementación de esta herramienta..

Para ello, se partirá de un modelo tridimensional de una prótesis interna y se mallará el mismo con una malla cartesiana en tres dimensiones. Esto va a permitir discretizar el objeto y colocarlo en la ubicación correcta dentro de la imagen médica del paciente para realizar los análisis modales pertinentes. De esta manera, estudiando las frecuencias naturales resultantes de los análisis se puede caracterizar la fijación del implante al hueso en función de las propiedades mecánicas del mismo.

Las dificultades asociadas a este procedimiento son varias, y en términos generales pueden resumirse en los siguientes puntos:

- Es necesario averiguar cómo exportar la malla tridimensional a *Matlab* y resolver las posibles incompatibilidades existentes por el formato propio de los archivos generados.
- Se debe analizar el código ya realizado de elementos finitos (*FEAVox*) e incorporar al mismo la formulación propia del análisis en vibraciones libres.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

- Deberán llevarse a cabo simulaciones con diferentes ejemplos, para lo cual ha de conseguirse archivos con información médica de pacientes.
- Es posible que sea necesario averiguar cómo identificar qué elementos son atravesados por cada superficie del objeto, con el objetivo de saber dónde se encuentran en relación con la superficie real del objeto, lo que será importante en el proceso de ensamblado de la prótesis con la imagen médica del TAC.
- Un paso fundamental será averiguar cómo pasar la información necesaria desde *GiD*, que es el preprocesador empleado, a *Matlab* para involucrar ésta en el código propio del programa FEM (FeaVox).

La solución progresiva de todos los puntos considerados, unida al desarrollo de un software que permita la colocación de un implante sobre la imagen médica, permitirá conseguir el objetivo principal que da soporte a este documento.

3. Punto de Partida.

En el proceso de desarrollo y realización de este *TFM* se deben emplear varios programas, así como lenguajes de programación, siendo indispensable comprenderlos plenamente para la consecución efectiva de los objetivos propuestos.

El manejo pleno del software a emplear permitirá la consecución efectiva de los objetivos propuestos. El primero de esos programas es *GiD*, con el que se va “voxelizar” el modelo de implante de cadera. Los conocimientos de partida en la utilización de este programa son prácticamente nulos, con lo que es necesario un aprendizaje previo.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Por otra parte, dadas las necesidades específicas de este proyecto, será imperativo recobrar los conocimientos adquiridos de programación, pues habrá que realizar tareas de análisis del código de *FEAVox*, realizado por diferentes personas, con sendas formas de utilización del lenguaje de programación. Esto también será necesario para poder desarrollar algún tipo de programa que permita la colocación del implante en posición correcta sobre la imagen médica del paciente.

4. Procedimiento.

El procedimiento a seguir en el desarrollo de este *TFM* refleja de manera ordenada todos y cada uno de los puntos mencionados en los objetivos de este documento. En primer lugar, se debe conseguir el modelo del implante que se va a utilizar en el proceso. Seguidamente, será necesario conocer la imagen médica del paciente con el objetivo de conocer los tamaños de voxel de la misma. Esto permitirá mallar posteriormente el implante de acuerdo a los tamaños de la imagen médica establecidos.

Una vez mallado el implante, éste deberá colocarse adecuadamente sobre la imagen médica para poder determinar qué píxeles de la misma deben cambiarse al color correspondiente al implante en la escala *Hounsfield*. La colocación del objeto se realizará a través del desarrollo de una interfaz de usuario sencilla lo más independiente posible de otros programas comerciales.

Por último, cuando se haya modificado la imagen médica del implante, habrá que incorporar en *FEAVox* la formulación propia del análisis modal y evaluar las frecuencias naturales tanto del conjunto como del implante por separado, lo que permitirá determinar el grado de fijación del objeto al tejido biológico.

5. Introducción.

La realización de este TFM pasa inevitablemente por la comprensión de algunos conceptos teóricos que abarcan diferentes campos científicos. Estos conceptos se desarrollan a continuación, y pretenden ser una referencia simplificada que facilite el contexto en el que este trabajo está inmerso.

5.1. ¿Qué es un TAC?

La tomografía axial computerizada (TAC) es una técnica de imagen médica que utiliza rayos X para obtener secciones o cortes de objetos anatómicos con fines de diagnóstico. Se diferencia de la radiografía en que en lugar de obtener una imagen de proyección, el TAC obtiene múltiples imágenes al efectuar la fuente de rayos X y los detectores de radiación movimientos de rotación alrededor del cuerpo.



Figura 1.- Máquina de imágenes por resonancia magnética

La reconstrucción de la imagen tomográfica se obtiene mediante la captura y posterior procesamiento de imágenes transversales a modo de múltiples planos que pueden incluso regenerarse para formar imágenes en tres dimensiones. Dicha información está

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

contenida en uno o varios archivos DICOM (véase apartado 5.4), que es el formato estándar reconocido para aplicaciones médicas.

En términos generales, la tomografía computarizada utiliza un equipo especial de rayos X para ayudar a detectar una variedad de enfermedades y condiciones. La exploración es rápida, indolora, no invasiva, precisa y en casos de emergencia puede identificar lesiones y hemorragias internas lo suficientemente rápido como para ayudar a salvar vidas.

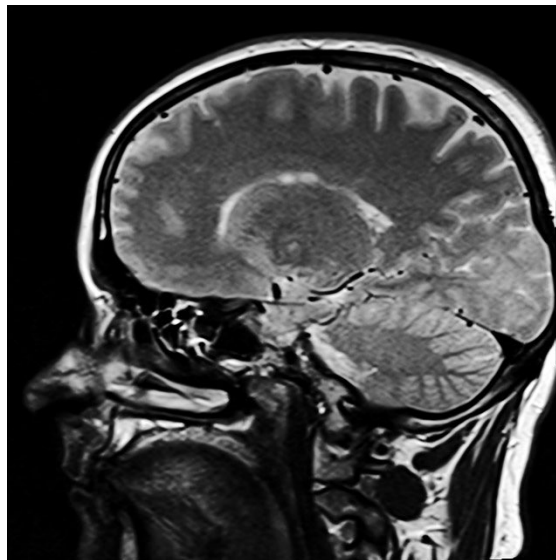


Figura 2.- Ejemplo de la información que se obtiene mediante un TAC

5.2. ¿Qué es una prótesis?

Una prótesis es una extensión artificial que reemplaza o provee una parte del cuerpo que falta por diversas razones. Debe reemplazar un miembro del cuerpo dando casi la misma función que un miembro natural. Es habitual confundir un aparato ortopédico con una prótesis. [1]

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Hay diferentes tipos de prótesis, entre las que destacan:

a) **Endoprótesis**

Se trata de una inclusión en el interior del organismo de una pieza extraña, metalada o de materia plástica, destinada a reemplazar de forma permanente un hueso, una articulación o cualquier otra parte o función del organismo.

b) **Exoprótesis**

Es un dispositivo que sustituye total o parcialmente un miembro del aparato músculo-esquelético, por amputación; cuya colocación o remoción no requiere de medios quirúrgicos, y que son utilizados para obtener una rehabilitación biomecánica o la restitución anatómica que permita una adecuada integración al medio laboral y social.

El tipo de prótesis que se va a tratar en este documento, por las características intrínsecas del trabajo, pertenece a la clasificación denominada endoprótesis.

5.3. Materiales en Prótesis Internas.

Aquellos materiales aptos para pertenecer temporal o definitivamente al cuerpo humano sin que se produzca biotoxicidad se denominan biomateriales, y todos ellos tienen en común las siguientes propiedades:

- **Biocompatibilidad:** Es la capacidad del material para ser usado en alguna aplicación específica dentro del cuerpo humano sin que éste reaccione de manera inadecuada y detecte la prótesis como cuerpo extraño, activando de este modo las diferentes líneas de de defensa del organismo.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

- **Bioinercia:** La capacidad para no provocar reacción alguna en los tejidos vivos circundantes después de su implantación en el organismo, aunque en ciertos casos sí que se produce una misma respuesta inmunológica.
- **Bioactividad:** En algunos casos, los materiales tienen la capacidad de estimular o provocar una acción biológica en el tejido receptor. Esta acción biológica debe tender a favorecer su integración.

A modo de ejemplo se expone una breve tabla con los biomateriales más usados para implantes en el sistema esquelético. [2]

Aplicación	Materiales Utilizados
SISTEMA ESQUELÉTICO	
Reemplazo de articulaciones	Titanio, aleaciones Ti-Al-V, acero inoxidable, polietileno (PE)
Placas óseas	Acero Inoxidable, aleaciones de cobalto-cromo
Cemento óseo	Polimetilmetacrilato (PMMA)
Reparación de defectos óseos	Hidroxiapatita (HAP)
Tendones y ligamentos artificiales	Teflón, dracón
Implantes dentales	Titanio, óxido de aluminio, fosfato de calcio

Tabla 1.-Relación de los materiales habituales para prótesis con sus aplicaciones más habituales

5.4. El Formato **DICOM**.

El formato *Digital Imaging and Communication in Medicine (DICOM)* es un estándar reconocido mundialmente para el intercambio de pruebas médicas. Está pensado para su manejo, visualización, almacenamiento, impresión y transmisión, ya que todos los sistemas

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

médicos de obtención de imágenes (como rayos X, ultrasonidos o la resonancia magnética) soportan y emplean este estándar, cuyo núcleo radica en la definición de un formato de fichero y de un protocolo de comunicación de red. [3]

- **El fichero DICOM:** Todas las imágenes médicas están guardadas en extensión .dcm o .DICOM. El equipo médico genera archivos *DICOM* y el personal médico emplea software de visualización *DICOM* para sus diagnósticos. Los ficheros *DICOM* se componen de una cabecera con campos estandarizados y de un cuerpo con datos de imagen. Por otro lado también disponen de información del paciente (nombre, ID, sexo, fecha de nacimiento...), el tipo de equipo empleado en la adquisición de los datos y datos del contexto de estudio que sirven para enlazar la imagen con el campo de la medicina pertinente. Un objeto *DICOM* admite el almacenamiento de bloques con varios fotogramas, en los que los formatos de imagen pueden estar comprimidos usando gran variedad de estándares, como *JPEG* o *TIFF*.
- **Protocolo de Comunicación de Red:** Todas las aplicaciones que están conectadas a la red del hospital usan el protocolo para el intercambio de información, que se identifica principalmente con imágenes *DICOM*, paciente e información sobre procedimientos. La comunicación de red suele emplearse para buscar imágenes en archivos y restaurar los estudios de imagen en la estación de trabajo con el objetivo de poder mostrarlos. Sin embargo, también permite llevar un control sobre el tratamiento, procedimientos e informes de estado, siendo posible el intercambio de información entre doctores y dispositivos de imagen.

El formato *DICOM* permite la integración de escáneres, servidores, estaciones de trabajo, impresoras y hardware de red de múltiples proveedores dentro de un sistema de almacenamiento y comunicación de imágenes.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Una de las ventajas de este tipo de estándares es que, independientemente del uso que se le de, siempre se emplea el mismo tipo de formato. Además, se diferencia de otros tipos de archivos en que *DICOM* agrupa la información dentro de un conjunto de datos, esto es, por ejemplo, una radiografía de Tórax contiene el ID del paciente junto a ella, de manera que la imagen no puede ser separada por error de su información.

5.4.1.Ejes de Referencia.

A la hora de visualizar una imagen médica en un programa específico para este fin es notable que los ejes de referencia considerados no aparecen como tales en la representación. Por lo general, en este tipo de software se ofrece una visión por planos de corte de la imagen médica, con indicaciones de la posición de los ejes por medio de letras. Esto, si bien puede resultar relativamente sencillo para el personal médico, es más complicado de cara a la superposición de un implante mallado sobre una nube de píxeles. Es por esto que es necesario conocer qué ejes de referencia se tienen en cuenta en la captura de la imagen y en qué dirección y sentido están ubicados.

El formato de archivos DICOM define un sistema de coordenadas de referencia bastante intuitivo a partir del cual orientarse. Para ello establece una postura del paciente estándar orientada frente al observador. Los ejes se definen de la siguiente manera:

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

- **Eje X:** Su dirección viene definida desde la derecha a la izquierda del paciente.

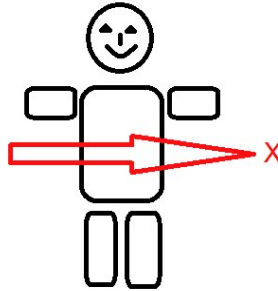


Figura 3.- Dirección del eje x en formatos *DICOM*

- **Eje Y:** Considera la parte anterior y posterior del paciente, en ese orden.

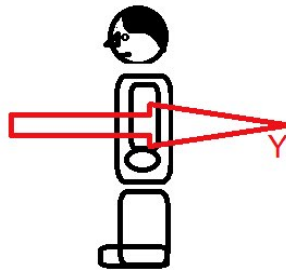


Figura 4.- Dirección del eje Y definida por el formato *DICOM*

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

- **Eje Z:** Va desde los pies del paciente hasta la cabeza.

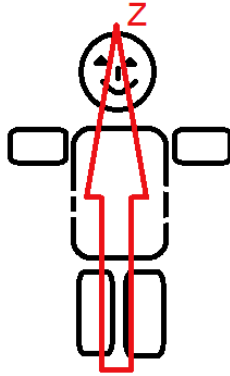


Figura 5.- Dirección del eje Z definida por el formato DICOM

En la nomenclatura de los programas de visualización comerciales, la equivalencia es:

[R] - Right - Dirección X decreciente.

[L] - Left - Dirección X creciente.

[A] - Anterior - Dirección Y decreciente.

[P] - Posterior - Dirección Y creciente.

[F] - Feet - Dirección Z decreciente.

[H] - Head - Dirección Z creciente.

En la *figura 6* se tiene un ejemplo de lo expuesto anteriormente. Se puede apreciar de manera simplificada un plano axial de una imagen médica con las letras y la ubicación de los ejes de referencia.

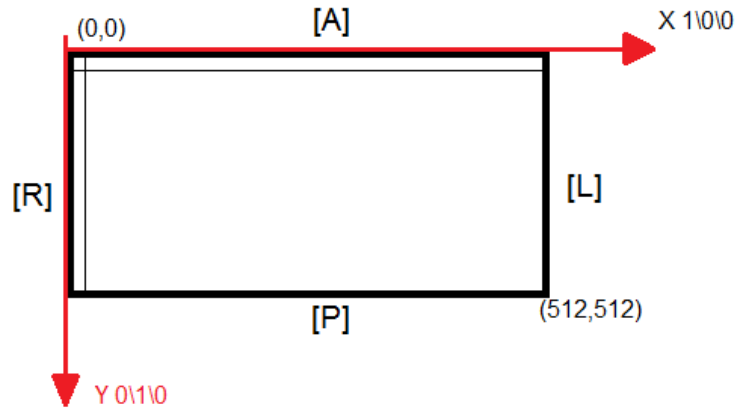


Figura 6.- Correspondencia de las letras de dirección con los ejes de un formato *DICOM*

Se puede apreciar la matriz de píxeles en el plano axial (color negro). En la esquina superior izquierda se tiene el origen de coordenadas y en la opuesta el punto de coordenadas máximas (en este caso 512 en X y 512 en Y). En color rojo se muestran los ejes de referencia tal y como se han descrito al principio de este apartado.

5.4.2. Planos de Corte.

Desde un punto de vista más intuitivo, puede considerarse el formato *DICOM* como un conjunto de planos de corte sucesivos en las direcciones marcadas por los ejes de referencia. De este modo, la información que se extrae puede ser visualizada en forma de imágenes bidimensionales que conforman, entre sí, las diferentes vistas de un mismo objeto, que en este caso es la zona de estudio del paciente.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

En términos ingenieriles, estas vistas se identifican con los conceptos de alzado, planta y perfil. No obstante, en el ámbito de la medicina y dentro de los estándares asociados al formato DICOM, los planos de corte se asocian con los términos coronal, sagital y axial, que se explican en las figuras siguientes:

- **Plano Coronal:** Divide al paciente entre las partes anterior y posterior. Define un corte perpendicular al eje Y.



Figura 7.- Plano de corte Coronal

- **Plano Sagital:** Divide al paciente entre extremidades. Corte perpendicular al eje X.

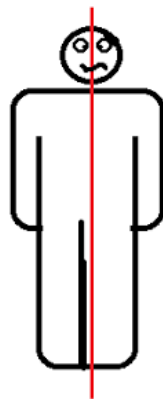


Figura 8.- Plano de corte Sagital

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

- **Plano Transversal o Axial:** Divide al paciente en dos entre la cabeza y los pies. Corte perpendicular al eje Z.

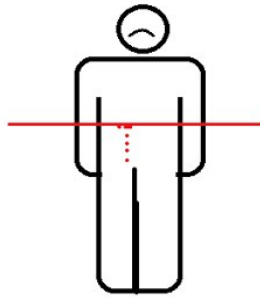


Figura 9.- Plano de corte Transversal o Axial

5.4.3. Información de Interés en Ficheros DICOM.

Como ya se ha mencionado, en un fichero con extensión *DICOM* no solo se tiene información en formato imagen, también es posible encontrar información inherente al paciente y datos propios del sistema de adquisición. Si bien en el contexto de este trabajo fin de máster no es tan interesante la información puramente médica, dentro de estos archivos sí que se albergan datos de imagen que deben ser tenidos en cuenta, en la medida de lo posible, ya que no aparecen en todas las imágenes médicas. Las variables que definen esos datos son:

- **PixelSpacing:** Es un vector que contiene los tamaños en milímetros de los píxeles en las direcciones definidas por los ejes X e Y.
- **SliceThickness** o espesor de capa. Da información sobre la dimensión en milímetros de los píxeles en la dimensión definida por el eje Z.

- **SliceLocation:** Ubicación de cada capa en milímetros dentro de la referencia del eje axial (Z).
- **NumberofFrames, Rows y Columns:** Ofrecen respectivamente la información sobre el número de píxeles en las direcciones Z, X e Y.

5.5. Fundamentos del *Método de los Elementos Finitos*.

El *Método de los Elementos Finitos (MEF)* es un procedimiento numérico de resolución de problemas de Mecánica de Sólidos de gran importancia práctica. Se trata de una herramienta de cálculo muy potente que posibilita la resolución de infinidad de problemas.

La contrapartida principal de este método, sin embargo, es que proporciona soluciones aproximadas a la realidad, pero que pueden ser más que suficientes para el diseño mecánico.

Este método se puede definir como un procedimiento de discretización de problemas continuos planteados mediante la formulación de ecuaciones matemáticas, de tal forma que: [4]

- a) El dominio se divide en un número finito de partes (elementos) cuyo comportamiento se aproxima mediante un número finito de parámetros
- b) La solución del sistema completo se obtiene como un ensamblaje de los elementos que discretizan el continuo.

El procedimiento básico en el que se basa esta técnica sigue un partón perfectamente definido. Dado un sólido sometido a un sistema de cargas, condiciones

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

externas y restricciones cinemáticas, se subdivide el continuo en pequeñas partes (elementos) interconectados entre sí a través de los puntos extremos o nudos. Esto hace que el campo de desplazamientos en el interior de cada elemento puede expresarse en función de los desplazamientos que sufren los nudos del elemento (desplazamientos nodales).

Una vez discretizado el continuo, se podrá determinar la matriz de rigidez de cada elemento. El ensamblado de todas las matrices de rigidez individuales dará paso a una matriz de rigidez global que permitirá la obtención de los desplazamientos en los nudos de cada elemento. De esa manera, una vez conocidos dichos desplazamientos, se podrían determinar, de una forma aproximada, las tensiones y deformaciones en el interior del elemento.

El procedimiento anteriormente descrito puede ser asimilado más fácilmente si se describe con un sencillo ejemplo. Se expone el caso de una placa de dimensiones finitas, sometida a tracción en sus lados superior e inferior, y que posee un agujero circular, de un determinado radio, en su centro.

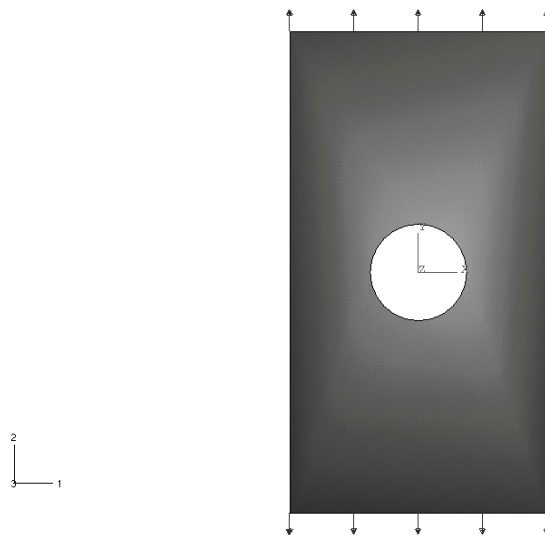


Figura 10.- Ejemplo de una placa con un taladro de un determinado diámetro

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Tras discretizar el conjunto con elementos triangulares, de forma que estos elementos se suponen interconectados entre sí a través de sus vértices, se aprecia que, por ejemplo, el círculo ya no es tal, sino que se ha transformado en un dodecágono. Esto implica que la solución que se obtenga tras el proceso, como se mencionó anteriormente, sea una aproximación a la solución real del problema. No obstante, al disminuir el tamaño de la malla (refinar) se podría reducir ese error por discretización.

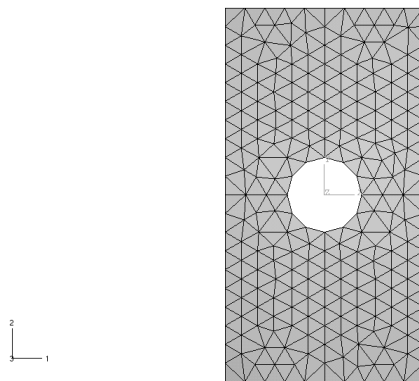


Figura 11.- Discretización del modelo de placa utilizado como ejemplo

En términos generales, en un problema a analizar por el Método de los Elementos Finitos, se conocen las fuerzas exteriores aplicadas y los desplazamientos en algunos de los nudos de la mala. La clave del método está en que, tras su aplicación, se obtiene una expresión del tipo:

$$[F] = [K] \cdot \{u\}$$

Donde F es un vector columna en el que se encuentran todas las fuerzas exteriores (las directamente aplicadas y las reacciones producidas por las condiciones externas) que actúan sobre el sólido. Por su parte, el vector u es el vector de desplazamientos nodales que representa, de manera ordenada, todos los desplazamientos que experimentan los nodos del modelo. Por último, la matriz K se conoce como la matriz de rigidez del sistema estructural, cuya formación es el núcleo esencial de este método.

$$[K^e] = \int_{V^e} [B]^T \cdot [D] \cdot [B] \cdot dV$$

5.5.1. Elementos Finitos en Dinámica y Vibraciones.

Si las fuerzas que actúan sobre un sistema son dependientes con el tiempo, la respuesta también variará con el tiempo. Si el movimiento resultante es tal que las fuerzas de inercia que produce no son significativas, en comparación con las aplicadas, el problema puede tratarse considerándolo como cuasi-estático, es decir, es posible utilizar la ecuación $[F] = [K] \cdot \{u\}$ para resolver el problema, considerando que tanto las fuerzas como los desplazamientos varían con el tiempo.

Cuando las fuerzas excitadoras tienen frecuencias mayores que un tercio de la frecuencia natural más baja, el efecto inercial es importante. Lo mismo ocurre si la estructura vibra libremente, es decir, en ausencia de fuerzas excitadoras. Las matrices de masa tienen en cuenta los efectos inerciales, y son una representación discreta de la masa distribuida.

En el caso de problemas de dinámica estructural, los que ocupan el objeto de este documento, interesa el caso de obtención de las frecuencias naturales de vibración y los correspondientes modos. Los métodos de dinámica estructural son bastante independientes del método de análisis por elementos finitos, debido a que los primeros presuponen la disponibilidad de las matrices de rigidez, amortiguamiento y masa, y no requieren que provengan de una discretización mediante el método de los elementos finitos. No obstante, la práctica habitual es usar el propio *MEF* para estimar de forma aproximada estas matrices.

Las ecuaciones que gobiernan la respuesta dinámica se pueden obtener considerando que el trabajo de las fuerzas externas debe ser igual al trabajo de las internas, inerciales y viscosas para cualquier movimiento virtual cinemáticamente admisible. En términos generales, y tras aplicar conservación de energía, se tiene que la matriz de masa es:

$$[M^e] = \int_{V_e} \rho \cdot [N]^t \cdot [N] \cdot dV$$

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

La matriz de masa es una representación discreta de la distribución continua de masa. Tal y como ha sido definida en la ecuación anterior, es consistente, lo cual quiere decir que se obtiene a partir de las mismas funciones de forma con las que se estimó la matriz de rigidez. Las matrices de masa consistentes no son diagonales, pero presentan la ventaja de que son positivo definidas (energía cinética positiva para cualquier vector de velocidades nodales).

En el límite, cuando el tamaño del elemento tiende a cero, el único movimiento experimentado se corresponde con una aceleración translacional de cuerpo rígido a nivel de elemento. Por este motivo, para que exista una convergencia de la solución de elementos finitos a la solución exacta, es necesario que se puedan representar movimientos de cuerpo rígido y que la matriz de masa represente la masa traslacional cuando el tamaño de elemento tiende a cero.

Si la malla representa correctamente el volumen de la estructura, los elementos son compatibles, no se utilizan reglas de integración reducida y las matrices de masa son consistentes. Las frecuencias naturales calculadas son superiores a las reales, con un error del orden h^{2p} , siendo h el tamaño del elemento y p el grado del mayor polinomio completo en la interpolación de desplazamientos. Si no se cumple alguna de estas condiciones las frecuencias naturales estimadas podrán ser mayores o menores que las reales.

Existe otro tipo de matrices de masa que son más simples en su formulación, ocupan menos almacenamiento y requieren de un esfuerzo computacional menor. Las matrices de masa concentradas son, por definición, matrices diagonales y semidefinitas positivas o indefinidas si aparecen masas nulas o negativas respectivamente en la diagonal. El cálculo de las frecuencias naturales con este tipo de matrices es, por lo general, inferior al valor exacto, pero tiene la ventaja de que, a menudo, la concentración de masa puede compensar el error de discretización introducido por elementos finitos. Su formulación tiende a ser intuitiva para elementos sencillos.

Los dos tipos de matrices de masa tienen sus ventajas, y no es posible determinar cuál es mejor para cualquier tipo de problema. No obstante, existen métodos de

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

concentración de la masa (HRZ, concentración óptima) que permiten concentrar una matriz consistente.

5.6. Análisis Modal.

Si un sistema mecánico se separa de su posición de equilibrio y se libera comenzará a vibrar, esto es, a oscilar en torno a una posición de equilibrio. El término vibración está normalmente asociado a la variación repetitiva de la configuración de un sistema alrededor de esa posición de equilibrio. En general, esa variación estará asociada con fuerzas actuantes en el sistema y dependientes del tiempo.

A pesar de que pueda parecerlo, los términos de vibración, movimiento y oscilación no son sinónimos. Toda vibración es una oscilación, y toda oscilación es un movimiento, pero no es así en sentido inverso. La diferencia específica que delimita el significado del concepto de vibración puede ser encontrada haciendo intervenir el concepto de energía, y más concretamente, el de energía de deformación (o potencial elástica).

La importancia del estudio de las vibraciones radica en que el comportamiento de un sistema mecánico (máquina o estructura) es muy diferente cuando las fuerzas aplicadas varían con el tiempo y cuando no lo hacen.

El número de aplicaciones del análisis de vibraciones en la ingeniería es muy amplio, y entre ellas se encuentran algunas de gran envergadura en las que los requerimientos de seguridad y fiabilidad son muy importantes. En muchas máquinas, por ejemplo, los requerimientos de funcionamiento abarcan el confort humano, y la exposición a vibraciones determina en gran medida el alcance de esa comodidad.

A pesar de que es posible modelar teóricamente un sistema vibrante, en muchas ocasiones es necesario realizar ensayos sobre prototipos o sobre el sistema real, con lo que se consigue, entre otras cosas, complementar o construir completamente los modelos teóricos.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

A continuación, y con el objetivo de clarificar la comprensión de los análisis modales objeto de este documento, se muestran algunos conceptos teóricos destacados. [5]

5.6.1. Clasificación de la vibración.

La vibración puede clasificarse de varias formas, siendo importante destacar la siguiente nomenclatura:

- **Vibración libre y vibración forzada:** Si un sistema mecánico se separa de su posición de equilibrio y se libera, éste comenzará a vibrar libremente (*vibración libre*). Las frecuencias en las que vibra el sistema en tales condiciones se denominan *frecuencias naturales*. Si un sistema se somete a una fuerza externa repetida cíclicamente, la vibración resultante se conoce como *vibración forzada*. Si la frecuencia de esa fuerza externa al sistema está próxima a alguna frecuencia natural del mismo se entra en la condición denominada resonancia, y las amplitudes de oscilación aumentan peligrosamente, pudiendo llegar a producirse el fallo.

Los problemas relacionados con vibraciones forzadas se suelen resolver eliminando o reduciendo la magnitud de la fuerza de excitación, lo cuál es relativamente más sencillo que identificar problemas relacionados con vibraciones libres, cuya solución suele estar relacionada con modificaciones físicas, es decir, de rigideces y/o reparto de masas en el diseño del sistema.

- **Vibración no amortiguada y vibración amortiguada:** Si durante la vibración no se disipa energía en fricción u otra resistencia, la vibración se llama *no amortiguada*. En caso contrario, la vibración es *amortiguada*.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

- **Vibración determinista y vibración aleatoria:** Si el valor de la excitación que actúa en un sistema vibrante es conocido para todo instante de tiempo, la vibración se denomina *determinista*. Si, como ocurre en algunos casos, la excitación no es conocida en cada instante, la vibración se llama *aleatoria* y el análisis de la misma debe realizarse en términos estadísticos.

5.6.2. Sistemas de N Grados de Libertad.

Un sistema de N g.d.l. es aquel en el que para definir la configuración del mismo es necesario utilizar un mínimo de N coordenadas independientes. Un ejemplo de este tipo de sistemas son aquellos constituidos por un conjunto de elementos rígidos (con propiedades inerciales asociadas) interconectados por elementos elásticos y disipativos. Estos sistemas se denominan discretos. No obstante, hay que indicar que en la realidad los sistemas son continuos, lo que quiere decir que las propiedades inerciales, elásticas y disipativas están distribuidas.

En términos generales, el tratamiento de los sistemas continuos de múltiples grados de libertad es complejo, pero siempre es posible modelarlos mediante un sistema de N g.d.l. lineales con el suficiente grado de aproximación.

5.6.3. Vibración Libre de Sistemas de N G.D.L. No Amortiguados.

Se considera ahora el análisis en vibraciones libres de un sistema lineal de N g.d.l., que es el caso que se tiene en cuenta en este TFM. Para el caso de amortiguamiento nulo, el sistema de ecuaciones diferenciales se escribe como:

$$[M]\{\dot{U}\} + [K]\{U\} = \{0\}$$

La solución de este problema puede expresarse como:

$$\{U(t)\} = \{\bar{U}\} \cdot e^{\lambda t}$$

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Sustituyendo la solución en la ecuación diferencial, se obtiene un problema generalizado de valores y vectores propios.

$$([K] + \lambda^2 [M])\{\bar{U}\} = \{0\}$$

Esta ecuación representa un sistema de ecuaciones lineales homogéneo. Para que este sistema de ecuaciones tenga solución distinta de la trivial es necesario que la matriz de coeficientes sea singular, o lo que es lo mismo, que el determinante de la matriz de coeficientes sea nulo.

$$\det [[K] + \lambda^2 [M]] = 0$$

Al resolver el polinomio resultante y suponiendo que las matrices de masa y rigidez son simétricas, de coeficientes reales y definido positivas o semi-definido positivas, es posible demostrar que las raíces se pueden expresar como:

$$\lambda_r = \pm i\bar{\omega}_n; r = 1, \dots, N$$

Siendo $\bar{\omega}_n$ las denominadas frecuencias naturales del sistema no amortiguado. El problema de valores propios también permite obtener los denominados vectores propios o *modos de vibración* (para cada uno de los valores propios hallados previamente), que determinan cómo va a vibrar el sistema para cada frecuencia natural obtenida.

Cabe destacar que en este documento la evaluación de $[K]$ y $[M]$ se realizará mediante el *Método de los Elementos Finitos*.

5.7. La Escala Hounsfield.

El modelo virtual formado durante una tomografía axial computerizada (TAC) está constituido por capas de imágenes apiladas secuencialmente que se unen para formar el modelo tridimensional. En cada una de esas capas, las imágenes individuales están constituidas por píxeles con valores numéricos individuales, que obedecen a valores en unidades *Hounsfield*. [6]

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

La escala *Hounsfield* se define como una medida de la densidad del objeto irradiado proporcional al grado de atenuación del haz de rayos X al pasar por cada material. Así, el valor en estas unidades de cada vóxel de la imagen médica será mayor cuanto más atenuación sufra la radiación emitida en la fase de escaneo del paciente.

Las unidades *Hounsfield*, denominadas así en honor a su inventor Godfrey *Hounsfield*, se encuentran normalmente en el rango de valores situado entre las ± 1000 unidades, siendo: -1000 correspondiente al aire, 0 al agua destilada y 1000 al hueso de elevada densidad (en condiciones normales de presión y temperatura).

A pesar de este rango orientativo para tejidos propios del cuerpo humano, la escala *Hounsfield* se define como una escala abierta, ya que otros materiales más densos que el hueso darán valores elevados no contemplados. La evaluación del valor correspondiente a este tipo de materiales puede obtenerse mediante la siguiente ecuación:

$$HU = 1000 \times \frac{\mu_t - \mu_{agua}}{\mu_{agua}}$$

Donde μ_t representa el coeficiente de atenuación lineal del tejido de estudio y μ_{agua} el correspondiente al agua destilada.

Teóricamente, el diagnóstico por imagen establece un rango de valores *Hounsfield* útiles comprendido entre -1024 y 3071 HU, lo que se corresponde con 4096 niveles de gris. Dado que el ojo humano solamente permite distinguir 32 niveles de gris a simple vista, y las pantallas de ordenador solo pueden representar 256 niveles, resulta necesario trasladar la escala a estos niveles de grises de cara a la representación.

A continuación se muestran valores orientativos de la escala *Hounsfield*.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Sustancia	HU
Aire	-1000
Pulmón	-500
Grasa	-100 a -50
Agua	0
Líquido Cerebroespinal	15
Riñón	30
Sangre	30 a 45
Músculo	10 a 40
Sustancia Gris	37 a 45
Sustancia Blanca	20 a 30
Hígado	40 a 60
Hueso Trabecular (Esponjoso)	700
Hueso Cortical (Compacto)	1000

Tabla 2.- Ejemplos de valores *Hounsfield* para diferentes órganos en el cuerpo humano

6. GiD. Software de Pre/Post Procesado.

El proceso de colocación del implante tridimensional sobre la imagen médica del paciente requiere "voxelizarlo" de alguna manera, es decir, poder discretizarlo mediante elementos simples que puedan ser manipulados con relativa sencillez. GiD es el software que se emplea en esta tesis para hacer efectivo el pixelizado.



GiD es un pre y post procesador universal de interfaz intuitiva y plenamente adaptable a cualquier código de simulación, que permite realizar múltiples tareas; desde la

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

definición de la geometría hasta la visualización de resultados, pasando por la discretización del modelo. Su punto fuerte radica en su carácter multidisciplinar, que se evidencia en su capacidad de comunicación con varios tipos de software de métodos numéricos.

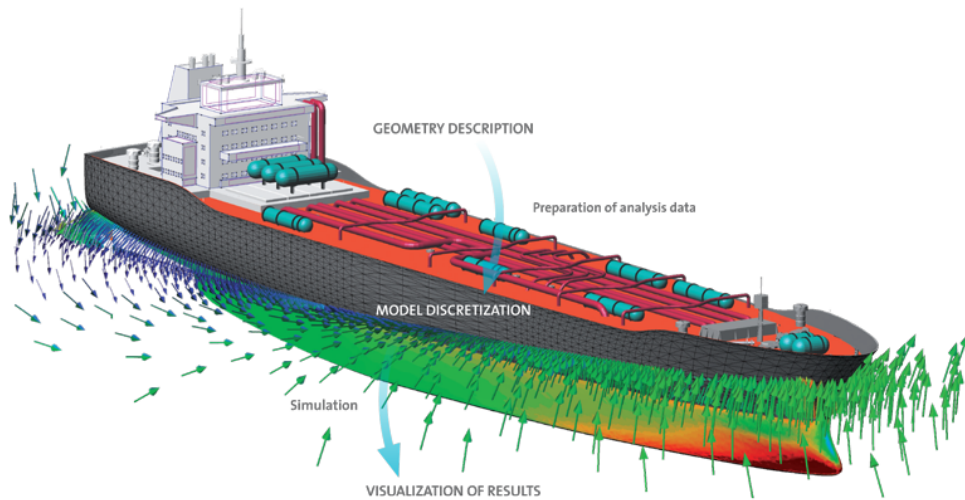


Figura 12.- Representación simbólica de las capacidades de *GiD*

Este programa posee unos formatos de entrada y salida adaptables en cierta medida, con lo que la comunicación con software externo de cálculo por métodos numéricos se hace más sencilla, permitiendo, además, iniciar y controlar los procesos de cálculo desde la propia interfaz de *GiD*, aprovechando así las herramientas que ofrece.

La integración con cualquier tipo de *solver* se puede realizar de manera sencilla, siendo el propio *GiD* el responsable de generar la información útil para el cálculo. Por otro lado, el programa cuenta con extensiones que facilitan su interconectividad con algunos códigos.

6.1. ¿Por qué GiD?

Una de las características principales de GiD es que permite unificar en un solo entorno las entradas y resultados del problema cuando se trabaja con múltiples códigos de elementos finitos. Además, sus propias posibilidades de comunicación entre programas posibilitan su compatibilidad con nuevos códigos, como es el caso de *ImplantViewer* (sección 7.1), la interfaz gráfica desarrollada en este trabajo fin de máster para favorecer la colocación del implante sobre la imagen médica del paciente.

Debe tenerse en cuenta que el verdadero valor de este software sale a la luz cuando se considera al programa como una plataforma intuitiva donde poder unificar varios códigos de simulación en un único entorno, lo que sumado a su gran soporte técnico, permite reducir considerablemente los costes de implementación derivados del desarrollo de un código propio de conexión entre software diferente.

En el ámbito específico de esta tesis, GiD será empleado fundamentalmente como preprocesador, pues será útil en la discretización del modelo 3D del implante en elementos hexaédricos (malla cartesiana), de igual tamaño al del vóxel de la imagen médica del paciente. Conseguir esos tamaños de los elementos de la malla permite garantizar las dimensiones correctas del implante sobre la imagen médica. Además, los requerimientos propios de *Implant Viewer* pueden implicar el giro del modelo CAD del objeto, con el objetivo de conseguir una malla con las dimensiones adecuadas en la posición final del implante sobre la imagen médica.

Los modelos *CAD* de las prótesis a emplear en este trabajo fin de máster se han obtenido en la red, aprovechando que hay plataformas donde se comparten archivos de modelado 3D. Dado el desconocimiento inicial sobre las dimensiones reales de este tipo de implantes, se han obtenido varias geometrías diferentes en formato *step*, con el objetivo de importarlos a *GiD* y de esta manera realizar la malla cartesiana tridimensional.



Figura 13.- Representación simplificada del proceso que se lleva a cabo en *GiD*.

6.2. Nociones Básicas de *GiD*.

La sencillez de uso de *GiD* radica en su interfaz de usuario, ya que resulta extremadamente fácil de comprender, incluso para un usuario que no está habituado a software común de representación y visualización *CAD*. En este apartado se pretende ofrecer una visión general sobre los principios de funcionamiento elementales que presenta *GiD*.

Se puede decir que este programa trabaja fundamentalmente en dos modos: preprocesado y postprocesado. El cambio entre esos modos se realiza a partir del menú *Files->Pre-process* o *Files->Postprocess*. En esta tesis se abarca únicamente el modo de preprocesado.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

En función del modo de trabajo en el que el usuario se encuentre, la interfaz de usuario de *GiD* cambia ligeramente. No obstante, ésta se compone fundamentalmente de los mismos elementos. Dispone de algunas barras herramientas (totalmente personalizables), de la zona gráfica donde se muestra el modelo (zona central de la interfaz), y de una barra de comandos inferior.



Figura 14.- Identificación de las zonas esenciales en *GiD*

Cabe destacar que las barras de herramientas disponibles cambian en función del modo en el que el programa está trabajando. Una explicación más detallada de las opciones disponibles en las barras de herramientas está disponible en el manual de referencia de *GiD*.

Además de la interfaz gráfica, y como parte de las nociones elementales sobre el funcionamiento de este software, es necesario explicar brevemente las opciones de control

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

del modelo tridimensional en *GiD*. Existen varias opciones disponibles en *GiD* que permiten hacer zoom sobre el objeto, rotarlo o trasladarlo.

- **Zoom:** Se puede realizar directamente con el ratón, mediante la rueda del mismo o pulsando el botón central del mismo mientras se presiona la tecla *shift*. La otra manera es haciendo click derecho en la zona gráfica y seleccionando el menú *zoom*.

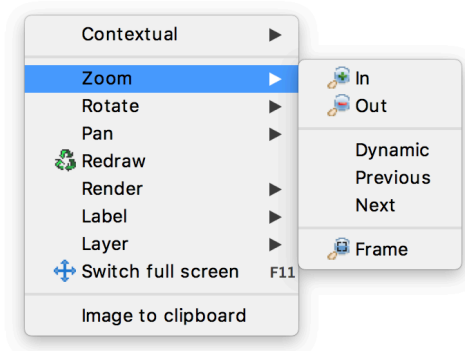


Figura 15.- Menú *Zoom* de la zona gráfica

- **Rotación:** Es posible rotar la vista del objeto mediante el movimiento del ratón mientras la tecla *shift* y el botón izquierdo del ratón están pulsados. También se puede rotar mediante el menú correspondiente de la ventana gráfica.

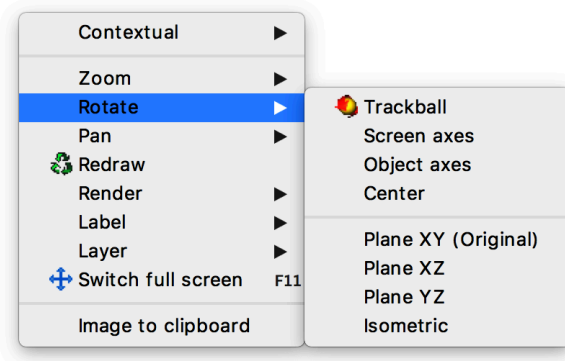
MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Figura 16.- Menú *Rotate* de la ventana gráfica

- **Traslación:** De igual manera que en la rotación, la traslación del objeto se puede realizar mediante el movimiento del ratón mientras la tecla *shift* y el botón derecho del ratón están pulsados. También mediante el menú correspondiente.

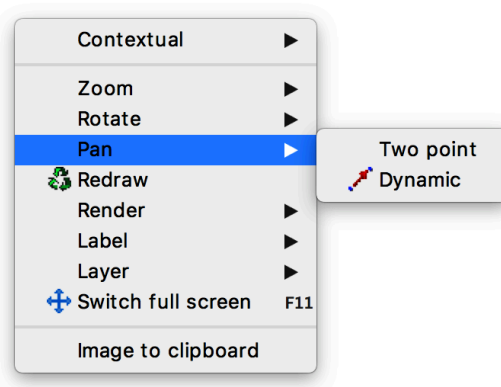


Figura 17.- Menú *Pan* de la ventana gráfica

Otro aspecto importante del funcionamiento básico de *GiD* es la posibilidad de poder cambiar el modo en el que el modelo es visualizado. Esto se cambia en el menú contextual de la zona gráfica (pestaña *Render*). Resulta especialmente interesante tener en

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

cuenta la representación alámbrica del objeto, pues en ella *GiD* ofrece un código de colores en la representación que identifica líneas, superficies y volúmenes. Esto puede ser útil para comprobar si el programa está detectando o no el volumen del objeto antes de mallarlo.

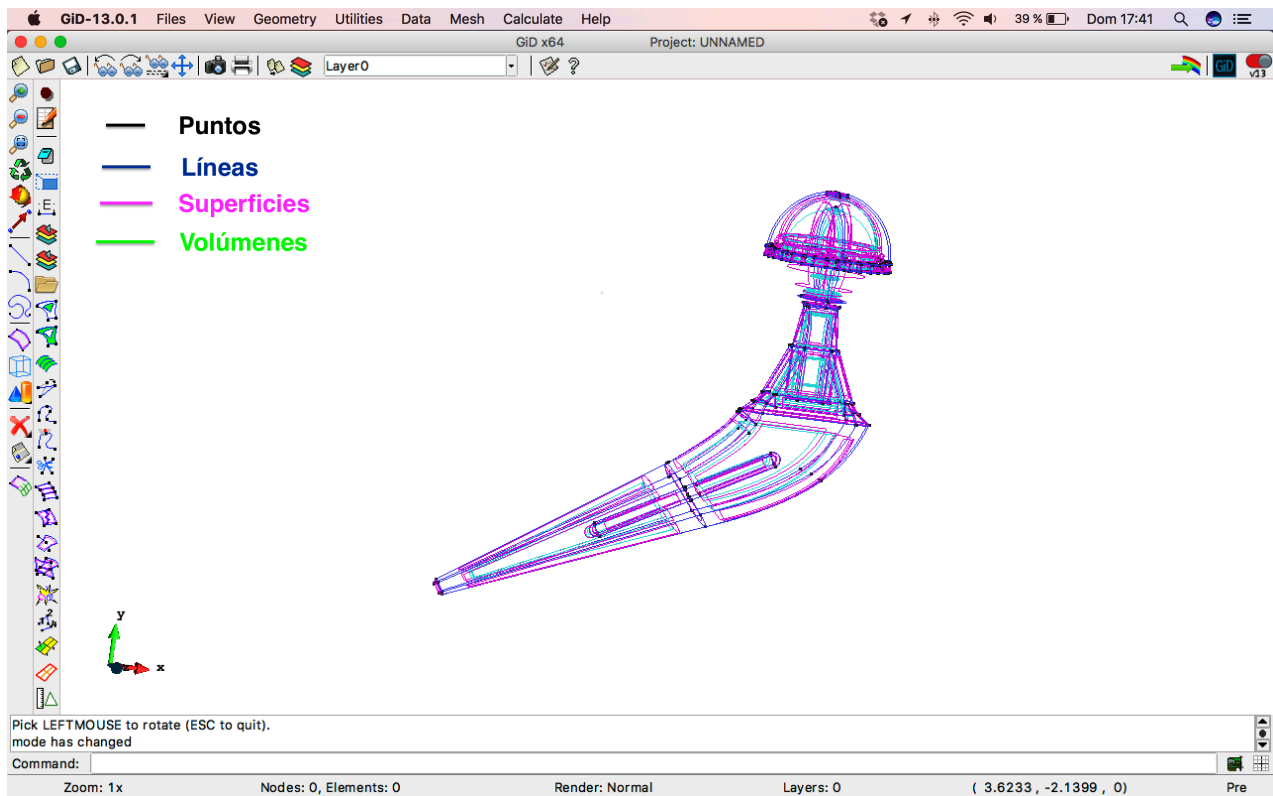


Figura 18.- Representación alámbrica y código de colores en *GiD*

Antes de finalizar la sección sobre nociones básicas en *GiD*, es importante resaltar un aspecto a tener en cuenta siempre que se utilice este programa: las unidades internas. Hasta el momento, *GiD* no contempla internamente el concepto de unidades en sus datos, de forma que para él éstos son una mera cadena de texto.

6.3. Obtención de una Malla Cartesiana Tridimensional en *GiD*.

En términos generales, el objeto tridimensional que se importa en *GiD* para iniciar el proceso de voxelizado se trata, desde el punto de vista interno de este programa, como un

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

volumen. Este volumen es el que debe ser reducido a una cuadrícula en tres dimensiones, de tal forma que represente su forma aproximada por elementos cuyas dimensiones estén alineadas con los ejes cartesianos principales. En otras palabras, debe realizarse una malla cartesiana tridimensional del mismo.

Una malla cartesiana es aquella en la que todos los elementos son “bloques” (en adelante serán designados como “voxels”) exactamente iguales, y de lados paralelos a las tres direcciones principales XYZ. Este tipo de mallado resulta útil en algunas aplicaciones de cálculo por métodos numéricos, como es el caso de la problemática que trata este documento.

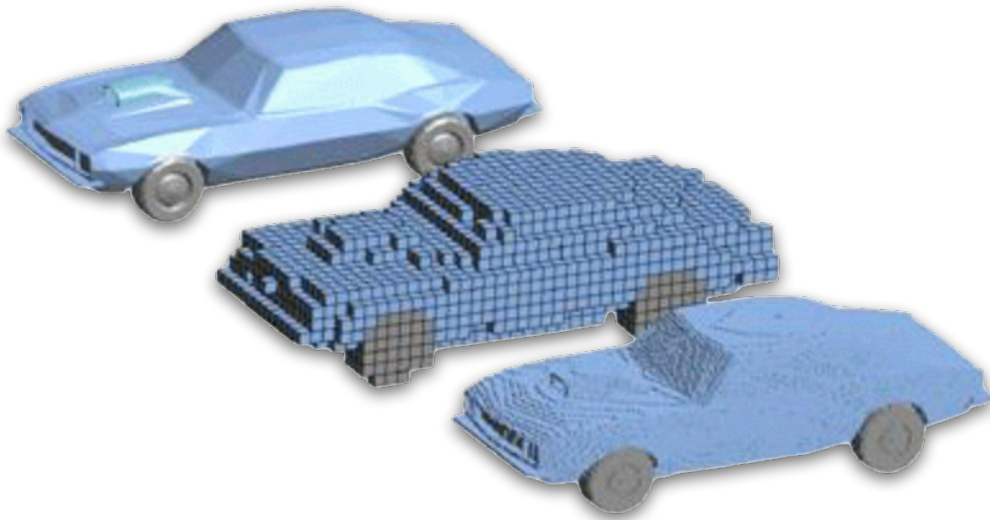


Figura 19.- Discretización de un objeto mediante mallados cartesianos con diferente orden de magnitud

GiD dispone de varios ajustes para la realización de mallas cartesianas. No obstante, destaca la sencillez de configuración y de control sobre el tamaño de los elementos. (Ver *figura 19*). Pese a que el proceso de mallado de un objeto CAD es relativamente sencillo en GiD, existen algunas comprobaciones previas a tener en cuenta para lograr una malla con la suficiente calidad como para que ésta sirva como input en Implant Viewer.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

6.3.1. Opciones de Mallado.

El proceso de voxelización necesario para la introducción del objeto tridimensional en *Implant Viewer* implica obtener, como ya se ha mencionado, una malla cartesiana tridimensional perfecta. La consecución de este objetivo, a su vez, pasa por la correcta configuración previa de las opciones de mallado disponibles para este tipo de malla.

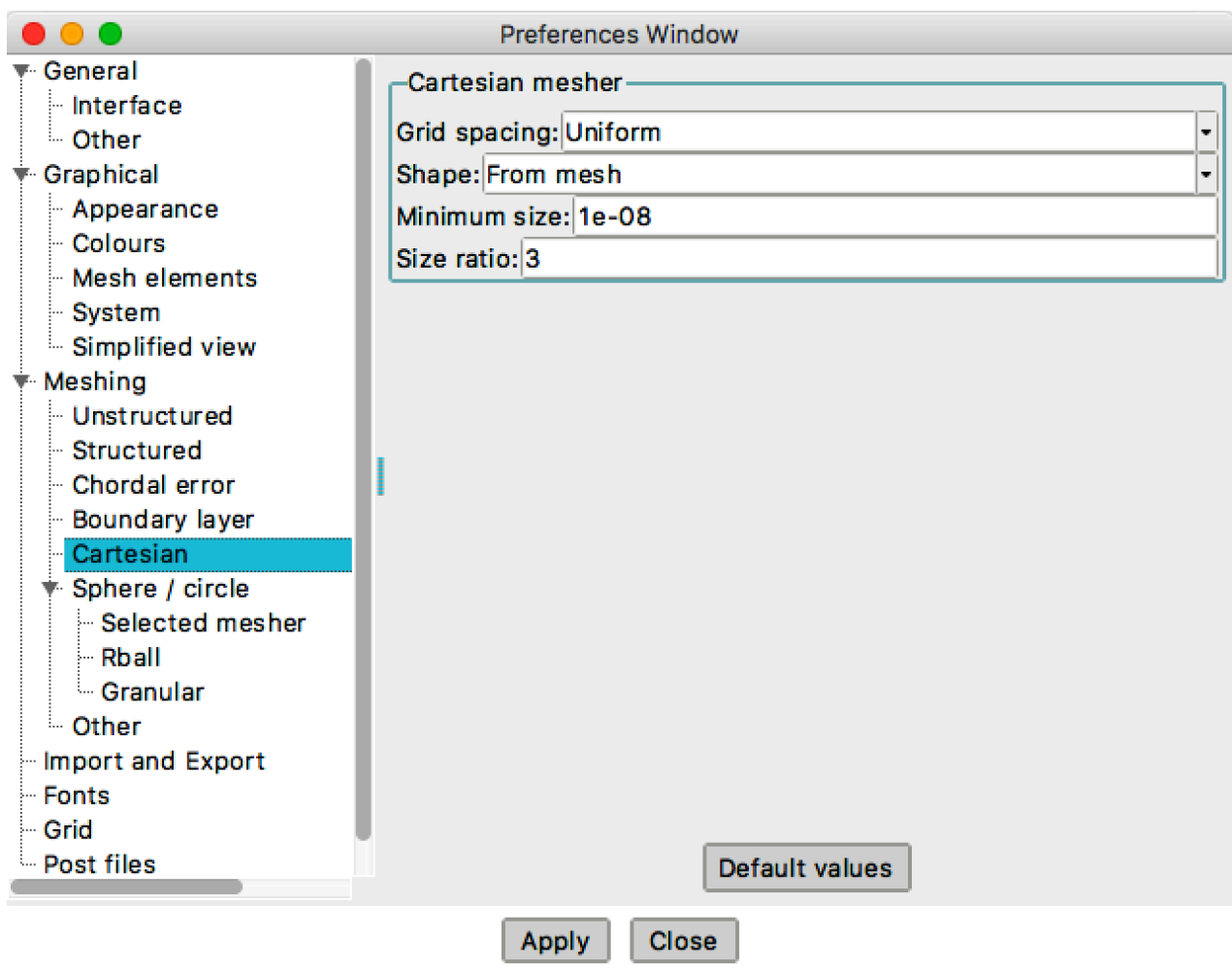


Figura 20.- Ventana de preferencias para mallados cartesianos en *GiD*

Para acceder a las opciones de configuración basta con acceder al menú *Utilities->Preferences*. Aparecerá entonces una *Preference Window* en pantalla, tras lo cual se hará

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

click en el apartado *Cartesian* (*dentro de Meshing*) y se verificará que las dos primeras opciones se encuentran tal y como aparecen en la *figura 20*.

El motivo por el cual las opciones de *Grid Spacing* y *Shape* deben verificarse es por un error en el mallado detectado en la versión 13 de GiD, y que se explica en la sección 6.4.

6.3.2.Importación de la Geometría.

Una vez se ha comprobado la correcta configuración del mallado cartesiano es posible cargar la geometría del implante. En principio, debe realizarse en este orden, de lo contrario puede ocurrir el error de mallado detallado en la sección 6.4.

Para importar la geometría en *GiD* simplemente hay que dirigirse al menú *Files -> Import -> STEP* y cargar el correspondiente archivo *STEP*, o el tipo de archivo de geometría, en su defecto. En general, la experiencia demuestra que en *GiD*, el mejor formato de archivo CAD para realizar la importación es *STEP*, pues es el que ha dado menos problemas de compatibilidad.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

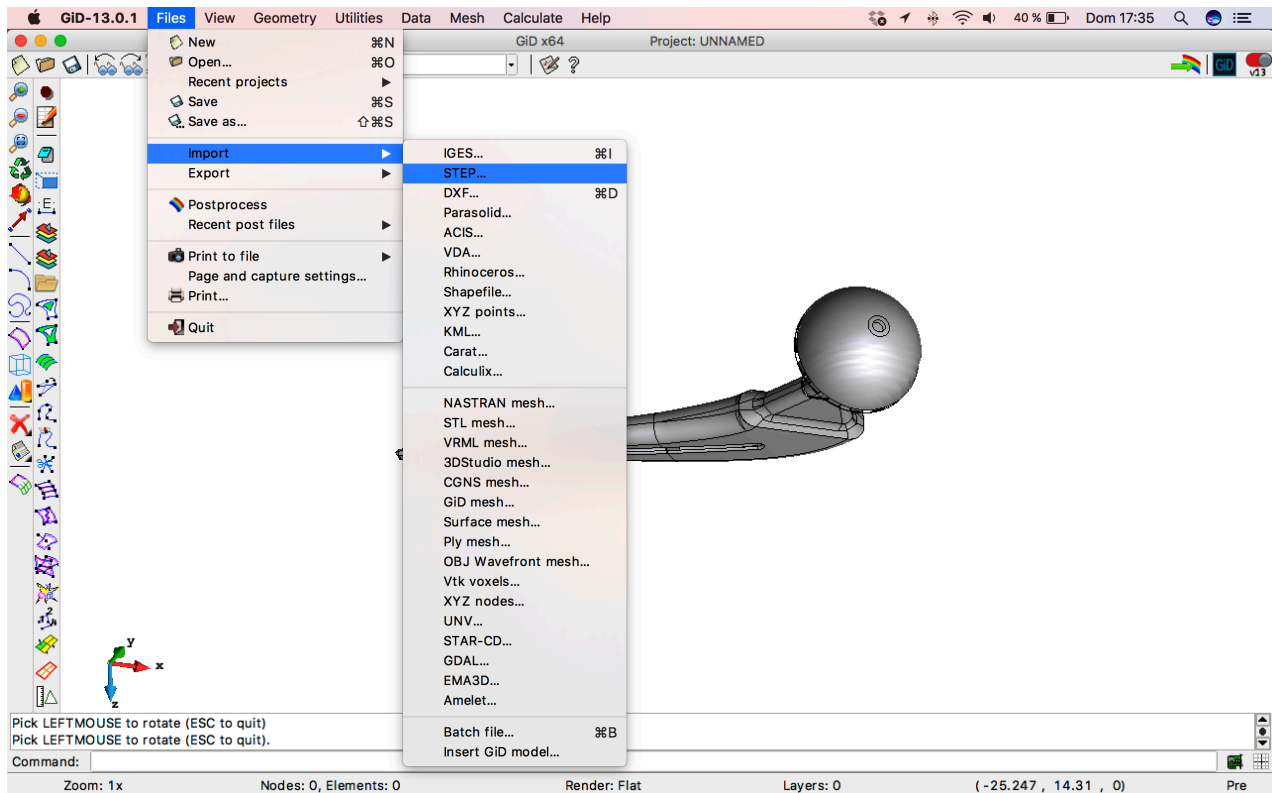


Figura 21.- Menú de importación de geometría en *GiD*.

6.3.3. Generación de Malla.

La generación de la malla es un proceso relativamente sencillo. Simplemente se debe asignar la malla cartesiana al volumen que conforma el objeto, definir las dimensiones del elemento base e iniciar el proceso de mallado.

Para asignar el tipo de malla a la geometría del objeto hay que dirigirse al menú *Mesh -> Cartesian -> Volumes*. Seguidamente, el programa mostrará una ventana en la que deberán indicarse las dimensiones de los elementos hexaédricos asociados a la malla cartesiana. Aquí deben introducirse los tamaños de vóxel que tiene el archivo *DICOM* asociados.

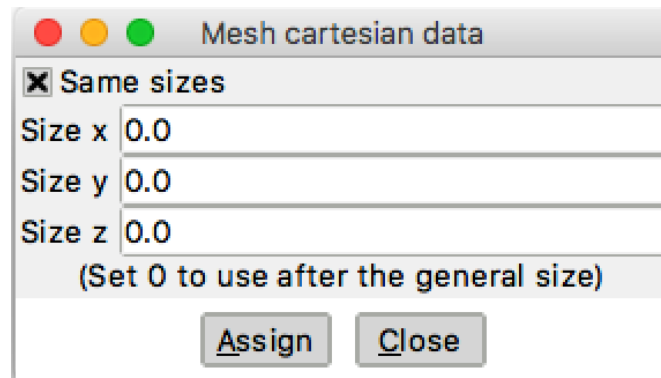
MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Figura 22.- Ventana de indicación de los tamaños de los lados del elemento hexaédrico

Posteriormente a la definición de los tamaños de elemento, *GiD* pedirá la selección del volumen a mallar, con lo que hay que seleccionar manualmente la geometría generando un rectángulo de selección directamente con el ratón izquierdo del ratón (o haciendo click sobre los volúmenes en el caso de la representación alámbrica). Para comenzar con el proceso de mallado no hay más que dirigirse a *Mesh -> Generate Mesh* o teclear *Ctrl + G*.

Previamente al inicio proceso de mallado *GiD* volverá a pedir un tamaño de elemento. Se trata de un valor global al que recurrirá en caso de que las dimensiones de malla que se especificaron inicialmente no puedan cumplirse. En ocasiones este valor puede originar problemas de mallado, con lo que se recomienda asignar un valor alto en comparación con las dimensiones de elemento especificadas.

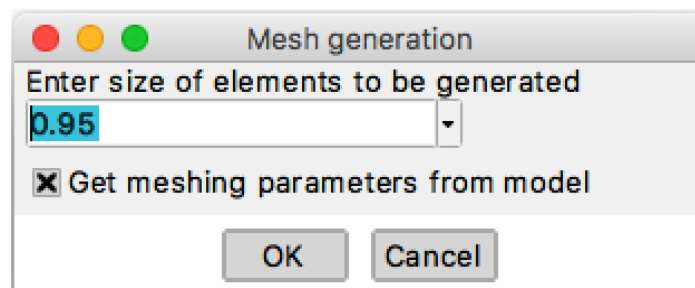


Figura 23.- Ventana de selección del tamaño general de malla cuando se inicia el proceso de mallado

6.3.4. Evaluación de la Calidad de la Malla.

Tras el proceso de mallado, es conveniente siempre revisar que los elementos resultantes tienen una forma adecuada, pues para asegurar la compatibilidad con las dimensiones reales de la imagen médica conviene saber si se ha voxelizado correctamente el implante. Para revisar la forma basta con ir a *Mesh -> Mesh quality...* y seleccionar *Shape quality* como criterio de calidad. Aparecerá un histograma que reflejará, en porcentajes sobre los ejes de referencia, cómo de ideal es la forma de los hexaedros en relación a su forma teórica. Si el proceso se ha realizado correctamente los elementos deberán tener un valor 1.

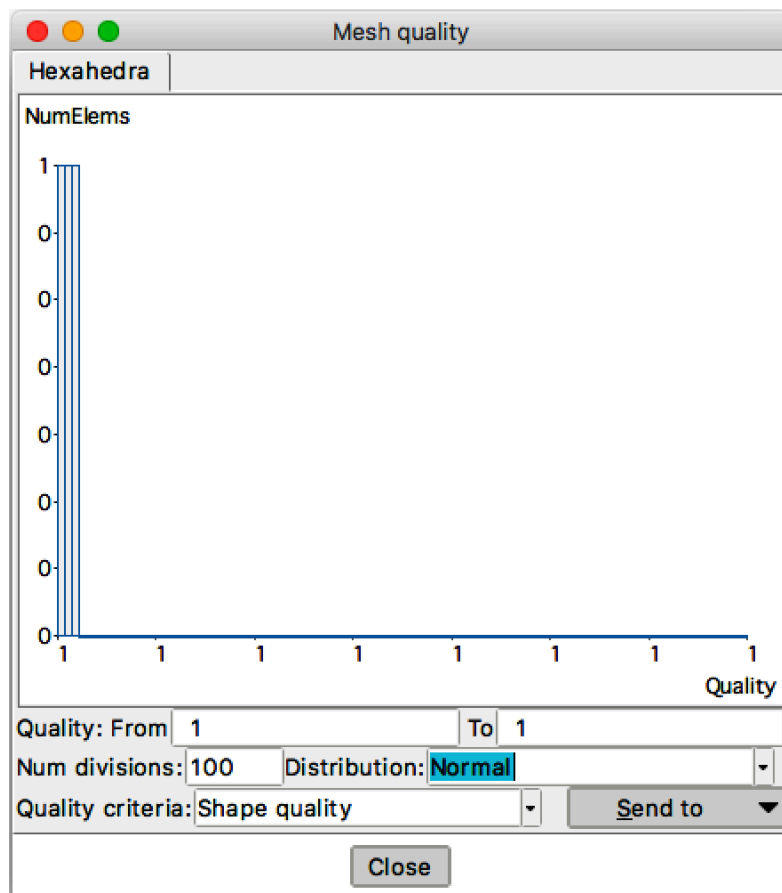


Figura 24.- Ventana de evaluación de la calidad de malla

6.4. Problemas Encontrados en *GiD*.

El proceso de importación de la geometría y mallado cartesiano no está exento de algunos errores debidos al propio procedimiento o a pequeños *bugs* de la versión de *GiD* empleada (13.0.1). Estos problemas han sido solventados de alguna u otra manera, y se exponen en este apartado.

6.4.1. Geometrías Mal Definidas.

A la hora de importar a *GiD* algunos archivos CAD, se han descubierto algunos errores en la definición de la geometría, tales como superficies duplicadas o infinitamente pequeñas, puntos no conectados con líneas, superficies no reconocidas... Por este motivo, se ha tratado de llevar a cabo, con las herramientas propias de *GiD*, un proceso de corrección y mejora de estos defectos.

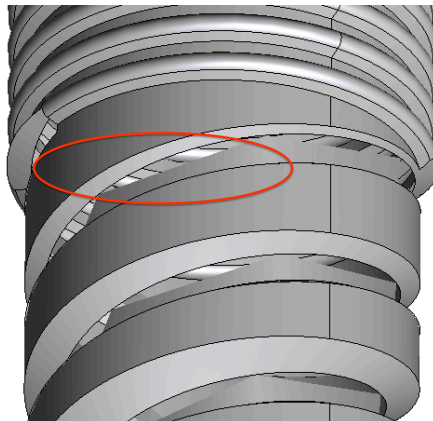


Figura 25.- Errores de definición de la geometría más comunes. Superficies que no quedan representadas

Lo primero ha sido, una vez importado el modelo a *GiD*, iniciar la propia herramienta interna que tiene el programa para la corrección de la geometría. Esta función (denominada *Repair Model*), si bien soluciona la mayoría de problemas, no es perfecta, por lo que debe

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

revisarse bastante bien el modelo para comprobar si ha reparado lo que se pretende o si, por contra, ha generado algún tipo de problema adicional.



Figura 26.- Error de definición de la geometría de la geometría de la pieza al tratar de reparar la geometría

Una vez comprobado el resultado de la ejecución de la autoreparación, se debe corregir el resto de la geometría. Esto, a su vez, lleva a varias complicaciones. Por ejemplo, si se debe borrar una línea para luego corregirla y dibujar otra en su lugar debe tenerse especial cuidado, ya que hay que borrar los volúmenes y superficies a los que esa línea está asociada para luego actuar. Es importante destacar que dibujar una línea con GiD en 3D sobre un modelo no es nada sencillo.

Especial mención debe tener la existencia de superficies o líneas que no se ven a simple vista, pero que pueden afectar a la creación de superficies (en el caso de que haya que hacerlo). Esto es un problema derivado de la importación del fichero, dado que el programa no lo reconocerá plenamente igual a como es el archivo en realidad.

Ante este problema geométrico se ha encontrado una solución que implica no tener que corregir la geometría: verificar si en el objeto *GiD* reconoce un volumen que represente el objeto. Si el objeto está mal representado y existe además un volumen reconocido (eso

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

puede verse mediante la representación alámbrica) el mallado será satisfactorio aunque en la representación sigan existiendo esos errores.

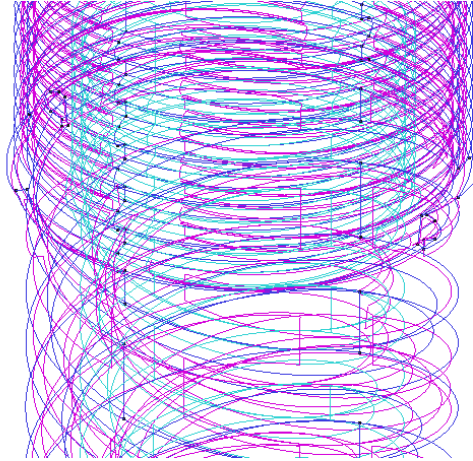


Figura 29.- Representación alámbrica del objeto. Se aprecia en verde que GiD reconoce un volumen en el interior de la geometría.

6.4.2. Mallado Cartesiano con Defectos de Forma.

El mallado cartesiano, como su propia definición indica, requiere de una igualdad dimensional de los vóxels, además de un espaciado uniforme entre elementos. GiD permite en las opciones de mallado establecer, entre otras cosas, el tipo de espaciado, que por defecto aparece variable.

Si se realiza una malla cartesiana sobre un volumen sin haber verificado previamente el tipo de espaciado como uniforme en las opciones del programa, éste generará una malla cuyos elementos no presentan la condición de que todos los hexaedros tienen iguales dimensiones, que es precisamente lo que se busca en esta tesis.

El problema parece estar en un *bug* del programa, ya que si no se cambia esta opción antes de mallar el modelo por primera vez, se generará esta malla, que poco tiene

que ver con una cartesiana, y de alguna forma, aunque se cambie la opción de espaciado a uniforme, el programa hará caso omiso.

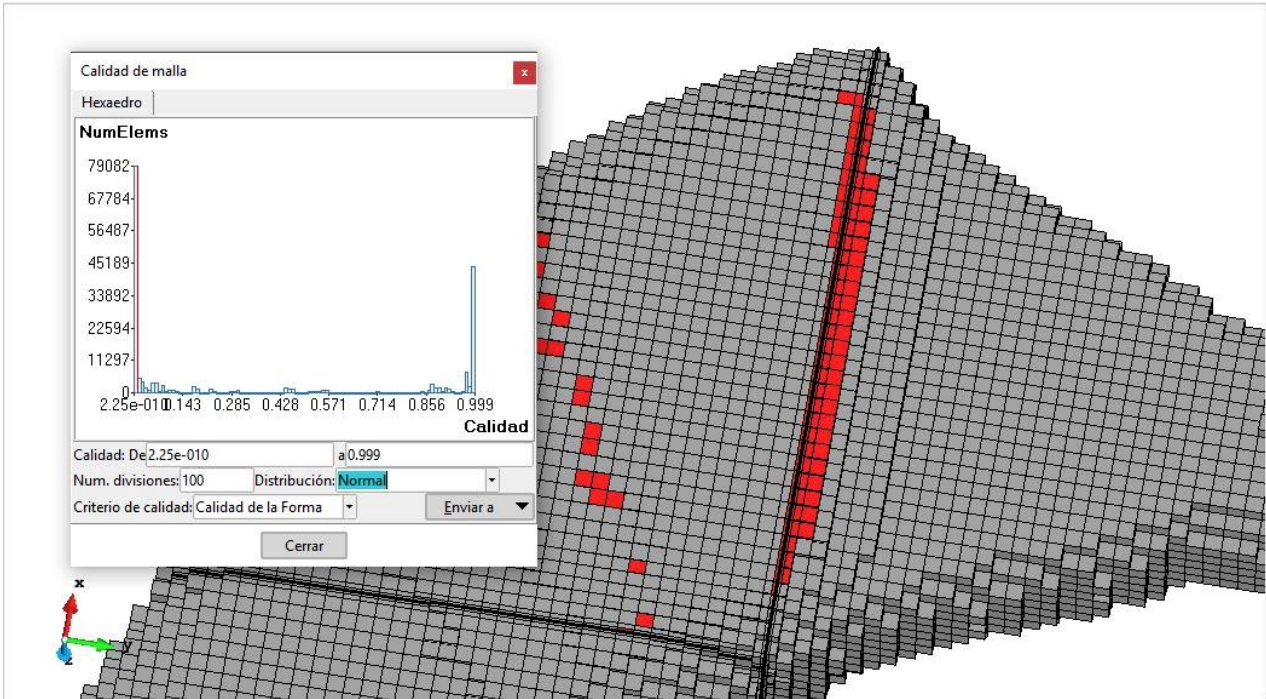


Figura 28.- Defectos en la forma de los elementos de la malla cartesiana

La solución, por lo tanto, pasa por seleccionar la opción *uniforme* desde el principio, incluso de manera previa a la carga de la geometría. El error parece ser un *bug* que se solucionará futuras versiones (la empleada es la 13.0.1).

6.4.3. Error por Tamaño de Elemento Mínimo.

Como se mencionó en la sección 6.3 que hablaba sobre el proceso de obtención de la malla cartesiana, *GiD* pide dos tamaños de elemento: el primero es específico y tiene prioridad a la hora de generar la malla, y el segundo es general y se emplea cuando no hay tamaño específico.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

De todas formas, *GiD* suele tener en cuenta el segundo dato de tamaño aunque se hayan especificado unas dimensiones en la primera opción. Esto hace que, si se define un valor general que el programa considera como muy pequeño para el objeto, aparecerán errores de mallado que impedirán discretizar el conjunto. La mejor opción en este sentido es indicar un tamaño general razonable pero que sea grande en comparación con las dimensiones dadas al elemento en un primer momento.

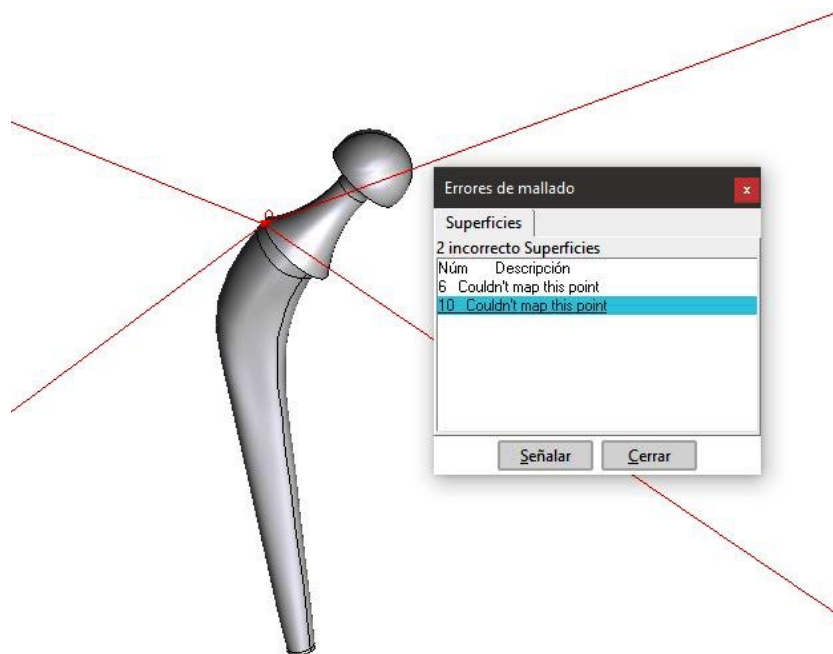


Figura 29.- Error típico por tamaño de elemento mínimo

6.4.4. Errores en la Numeración de Elementos y Nodos.

Aunque no es un error muy habitual, ha ocurrido que la numeración de nodos o elementos deja huecos numéricos grandes. Esto afecta notablemente al rendimiento de *Implant Viewer* (sección 7.1), pues produce un aumento innecesario el tamaño de algunas matrices internas y, por tanto, la velocidad directa de cálculo del programa. Por este motivo,

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

es necesario, en ocasiones utilizar la opción de reenumerar. Para ello, se debe cambiar el valor de alguna variable interna de *GiD*, lo que se desarrolla en la sección 6.7.

6.5. Información a Exportar desde GiD.

Para poder realizar en *Implant Viewer* (sección 7.1) el proceso de colocación de la geometría sobre la imagen médica, es necesario establecer qué tipo de información es necesario pasar cuando se realiza el mallado cartesiano en *GiD*. Este apartado pretende aclarar esa cuestión: [11]

- Se hace necesario saber **cómo ha realizado** el software **la malla**, esto es, saber de antemano cómo será la distribución interna de la información contenida en el archivo de malla exportado desde *GiD*.
- Es conveniente conocer el **número de elementos y nodos** presentes en la malla, pues esto determina en gran medida el tamaño de algunas matrices de cálculo.
- Las **coordenadas del centro de** cada **elemento** en referencia al sistema global podrían ser de utilidad, así como el número total de nodos y elementos y las dimensiones de los hexaedros generados, ya que en las imágenes médicas los vóxeles no tiene por qué ser iguales en todas las direcciones.
- Se debe determinar el **patrón de numeración de los nodos de los elementos**, pues resulta útil en los algoritmos internos de transformación de coordenadas de *Implant Viewer*.

6.6. Comunicación GiD-Matlab.

Durante el preproceso, será necesario realizar una comunicación efectiva entre GiD y Matlab (FEAVox e *Implant Viewer* en este caso). Este intercambio de información se va a realizar en forma de archivos exportados por parte de GiD.

Los archivos en cuestión tienen, por defecto un formato ASCII. De esta forma es más sencillo poder adaptarlos a una estructura común, pues GiD no está adaptado a todas las extensiones del mercado.

Para realizar la exportación basta con dirigirse a la siguiente opción del menú: *Files -> Export -> GiD Mesh*. Además, habrá que indicarle un nombre y una ubicación donde guardar la información que va a generar. Esta acción generará un archivo **.mesh*, que contendrá toda la información referente a la malla. Estos serán los datos que tendrá que usar *Implant Viewer* para obtener los datos de la prótesis de cadera.

La estructura del archivo de malla *ASCII* generado con *GiD* no está explicada en la documentación del programa como tal. No obstante, el archivo generado tendrá la misma estructura que las mallas de postproceso explicadas en los manuales del programa.

6.6.1. Estructura del archivo de malla.

Como ya se ha explicado, exportar la información referente a la malla del objeto tridimensional se exporta en formato **.msh*, que incluye los datos relativos a los nodos y elementos existentes en la malla. El mencionado archivo, que se encuentra en formato *ASCII*, tiene una estructura bien definida.

- **Cabecera:** En ella se determina el tipo de malla realizada, el número de dimensiones físicas que abarca la malla, el tipo de elemento y el número de nodos por elemento. La forma de la cabecera sigue el siguiente patrón:

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

MESH "mesh_name" dimension mesh_dimension Elemtyp element_type Nnode
element_num_nodes

- **Nodos:** Esta sección, que comienza y termina con las palabras “*coordinates*” y *end coordinates*”, contiene toda la información referente al número de nodo y su localización en las tres coordenadas del espacio.

COORDINATES

node number coordinate_x coordinate_y coordinate_z

...

END COORDINATES

- **Elementos:** En esta parte del archivo se muestran datos sobre los elementos existentes en la malla y los nodos que corresponden a cada uno de ellos, de acuerdo con la numeración nodal expuesta en la parte anterior. Se inicia con la palabra “*elements*” y termina con “*end elements*”.

ELEMENTS

element node_1 node_2 node_3 material_number

...

END ELEMENTS

6.7. Cambio de Variables en GiD.

En referencia a los errores encontrados durante el proceso de mallado en GiD (sección 6.4), es posible que sea necesario en algún momento cambiar ajustes de funcionamiento en este software. El programa informático GiD tiene definidas algunas opciones internas en forma de variables que determinan el comportamiento ante diferentes funciones (mallado, creación de geometría...). Estas opciones, pese a tener un valor por defecto, pueden ser cambiadas por el usuario, en función de lo que el manual de *GiD* establece como posible.

Principalmente, el acceso a las variables es posible de dos maneras. Un camino sencillo es abrir las preferencias del programa y ajustar ahí las opciones disponibles. No obstante, para algunas variables internas, no existe posibilidad de cambio desde la ventana de preferencias del programa. En estos casos es útil emplear la barra de comandos que se sitúa en la parte inferior de la ventana principal.

Para cambiar el valor de alguna de las variables se debe escribir "*Mescape Utilities Variables*", tras lo cual el programa pedirá el nombre de la variable a cambiar y el nuevo valor de la misma.

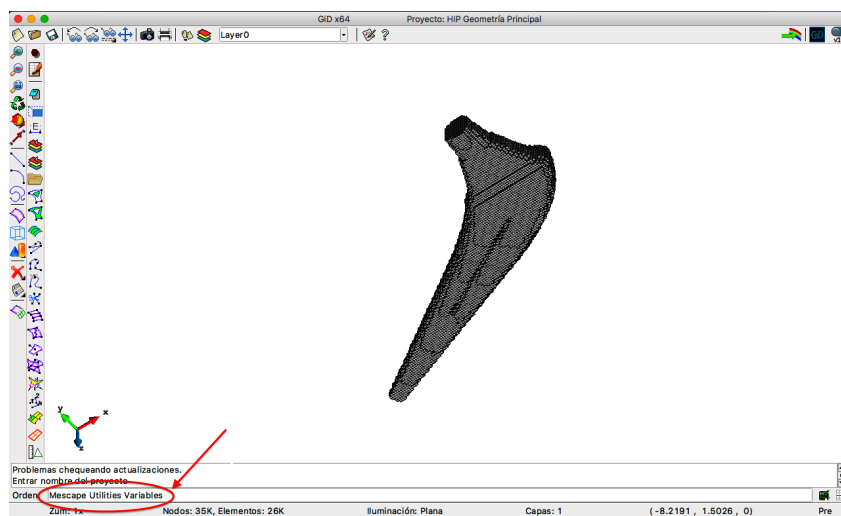


Figura 30.- Línea de Comandos en GiD

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

Una vez introducido el comando arriba descrito, es posible conocer las variables disponibles mediante el botón derecho del ratón y el apartado "Contextual" disponible en el menú que aparece. La información sobre los posibles valores de la variables está disponible en la ayuda del programa.

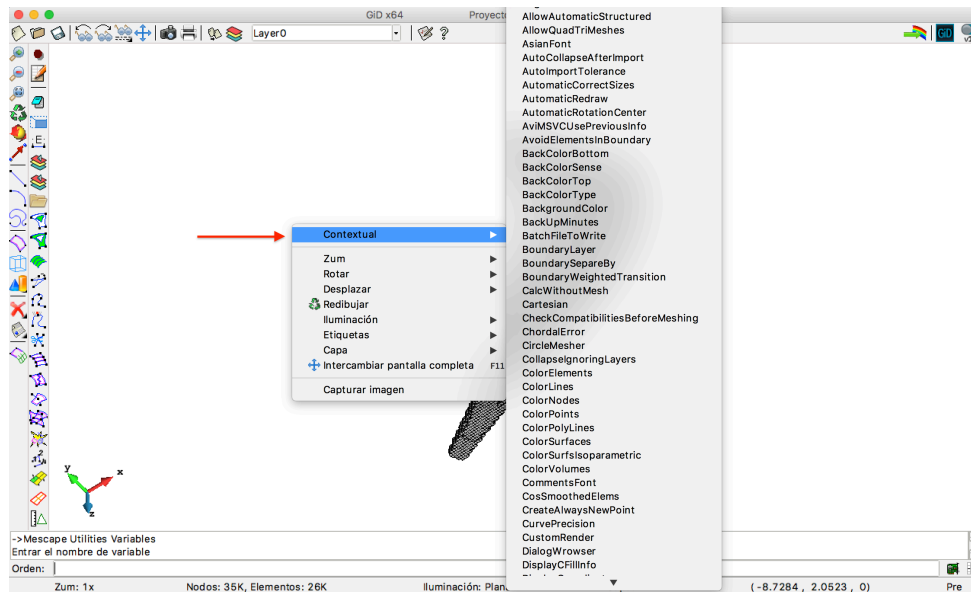


Figura 31.- Menú contextual del menú de la ventana gráfica en el que aparecen las variables disponibles de *GiD*

6.7.1. Renumerar Elementos y Nodos en *GiD*.

Cuando se crean nuevas entidades, las identificaciones asociadas a estas se corresponden con los números más bajos mayores que cero que no existan ya en las entidades existentes en el dibujo. Si por contra se eliminan entidades, se deja un hueco en la numeración, hueco que será asignado a las nuevas entidades que se creen posteriormente.

En algunas ocasiones, es de utilidad eliminar los huecos en las identidades numéricas de los elementos del modelo. Por este motivo se hace imprescindible poder reenumerar el conjunto de nuevo. *GiD* permite realizar una reenumeración de entidades, sin problemas con los materiales o condiciones de contorno aplicadas al modelo.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

Cabe destacar que esta función puede aplicarse tanto a la geometría como a la malla generada. En el modo geometría se modifican puntos, líneas, superficies y volúmenes. En el modo malla, se puede cambiar la numeración de nodos y elementos, y no sólo se eliminan los saltos en la numeración, sino que se minimizan las identificaciones de los nodos para reducir la diferencia entre ellos para cada elemento.

El algoritmo de renumeración puede ser cambiado mediante el valor de la variable "*RenumberMethod*". Por defecto el valor se sitúa en cero, pero hay otras opciones disponibles:

- -2 El método de renumeración evita modificar nodos y elementos. Los números de entidades se generan incrementalmente a medida que se generan.
- -1 En este caso no se modifican los nodos y se genera numeración incremental a medida que éstos se generan.
- 0 Éste es el valor por defecto. los nodos se ordenan en función a la distancia a algún sistema de referencia.
- 1 Algoritmo inverso Cuthill-McKnee, similar al anterior pero teniendo en cuenta la conectividad entre los elementos.
- 2 Reordenación en base a los ejes XYZ, útil en casos de geometría alineada con los ejes.

Una ampliación de esta información se encuentra disponible en el *GiD Reference Manual*. [7]

6.8. Giros de Geometría en *GiD*.

Debido al tipo de procedimientos internos de giro del objeto que incorpora *Implant Viewer* y a las características propias de las imágenes médicas en formato *DICOM*, es posible que en algún momento sea necesario girar la geometría para remallarla en su orientación final. En este sentido, se hace necesario explicar cómo pueden realizarse estos giros en las coordenadas del objeto CAD.

En primer lugar, en caso de que no se haya realizado previamente, es necesario borrar la malla previa existente. Esto se realiza en el menú *Mesh -> Erase Mesh*. Posteriormente habrá que dirigirse al menú *Utilities -> Move...* para poder visualizar la interfaz de giro.

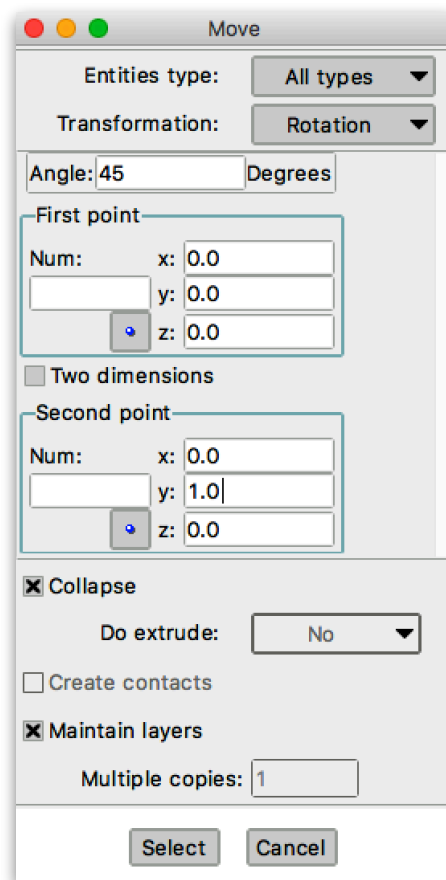


Figura 32.- Menú disponible para realizar rotaciones de la geometría en *GiD*

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Una vez en la interfaz (*figura 32*), se deberá especificar *All Types* como tipo de entidades a rotar, y *Rotation* como transformación de coordenadas. Se especificará un ángulo de giro y se indicará mediante puntos cuál es el sentido y dirección del eje de giro. Esto implica que se deben introducir los puntos por los que pasa el eje de giro. Si, por ejemplo es Z, en el segundo punto deberá ponerse un 1.

Tras estos pasos, el programa pedirá que se seleccione la geometría y finalmente ejecutará el giro en las coordenadas internas del programa. Esto supone, a su vez, que el giro se traducirá en que la malla que se regenere tendrá la misma orientación que el objeto CAD.

7. Programación en Matlab.

Como bien es sabido, uno de los objetivos principales de este *TFM* es proveer una primera aproximación que solviente la problemática que afronta el personal médico a la hora de evaluar la posible evolución de un paciente que necesita un implante. Con este objetivo en mente, el proceso propio de este trabajo se ha centrado en el desarrollo de códigos que permitan alcanzar este fin, por lo que se ha utilizado el lenguaje *Matlab* tanto para crear herramientas que permitan colocar adecuadamente el implante sobre la imagen médica del paciente como para integrar el análisis de vibraciones del mismo. *Implant Viewer* y *FeaVox* son los programas que permiten conseguir este objetivo.

En términos generales, el lenguaje de programación propio de *Matlab* no se compila, por lo que al ejecutar un programa se debe ir traduciendo a lenguaje máquina cada línea. Esto hace que sea menos eficiente frente a otros lenguajes que sí se compilan, como pueda ser C++ o FORTRAN. Por este motivo, *Matlab* se suele emplear para crear programas relativamente simples, dejando aquellos más complejos para otros lenguajes. [8]

A pesar de estas características, se pueden aplicar técnicas de mejora de la eficiencia con el fin de reducir los tiempos de cálculo, a la hora de ejecutar el código. La

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

programación de software aquí mostrado obedece al uso de algunas técnicas de optimización con el objetivo de paliar el problema de la eficiencia.

Existen múltiples técnicas de optimización de códigos en *Matlab*, y cada una de ellas se aplica en determinadas condiciones. Resulta útil saber cuál es la mejor opción en función de las condiciones del código en cada caso. Algunas de las alternativas más habituales de optimización son: [9]

- **Cambio de Algoritmo:** Cuando se trabaja con programas complejos no es sencillo encontrar un algoritmo que se ajuste a las necesidades específicas de cálculo. Es habitual cuando se programan grandes problemas que el primer código realizado sea extremadamente ineficiente, por lo que habrá que buscar algoritmos alternativos que presenten una mayor eficiencia desde el punto de vista del tiempo.
- **Vectorización:** Como es sabido, *Matlab* es capaz de trabajar con matrices y arrays, operando los mismos con una gran eficiencia. Esto incluye no sólo las típicas operaciones matemáticas, sino también las operaciones de lectura y escritura de diferentes elementos de la matriz.
- **Procesado paralelo a través de la GPU:** Las tarjetas gráficas no sólo son empleadas para renderizado gráfico, también como co-procesadores en grandes cálculos en computación, imágenes médicas o análisis financiero. Esto se denomina generalmente *General-Purpose GPU* (GPGPU). Hoy en día todos los ordenadores personales tienen una GPU, ideadas en un inicio para renderizado 3D de videojuegos. Los fabricantes liberan APIs (OpenGL o Direct3D) que permiten desarrollar software para propósitos generales. **CUDA, NVIDIA.** Cabe destacar también la posibilidad de poder paralelizar con CPU, técnica usada con frecuencia en este *TFM*.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

Las pautas de programación de *Implant Viewer* y *FEAVox* obedecen a estas tres estrategias principales de optimización de cálculo, siendo la más destacada la vectorización, que se aplica teniendo en cuenta además la forma en que *Matlab* guarda las posiciones de las matrices para hacer más rápida la indexación y la velocidad de cálculo.

A continuación se explica, de manera simplificada y desde el punto de vista de la programación realizada, el funcionamiento de *Implant Viewer* y *FEAVox*, cuya combinación permite conseguir el objetivo fundamental de esta tesis fin de máster.

7.1. Implant Viewer.

Implant Viewer es un programa informático desarrollado dentro del marco específico de este trabajo fin de máster, que pretende ofrecer una solución sencilla al problema de colocar virtualmente un modelo de implante sobre la imagen médica del paciente. Se trata de un software realizado íntegramente en *Matlab* que permite generar, de manera automatizada, una imagen en formato *MAT* de un paciente con un implante colocado en la posición correcta, para su posterior análisis. Para ello utiliza como inputs una malla cartesiana (*GiD*) y un archivo *DICOM*.



La característica principal que diferencia a este programa radica en que permite al usuario colocar el modelo CAD del implante de una manera sencilla e intuitiva, mediante su interfaz gráfica controlada principalmente por los gestos habituales del ratón. El programa, además, incorpora también algunas características útiles para el usuario que permiten complementar la función principal.

En esta sección se procederá a explicar, de manera simplificada, el funcionamiento del software a nivel interno. Para conocer más sobre el funcionamiento a nivel de usuario véase el ANEXO I. *Guía de Usuario de Implant Viewer*.

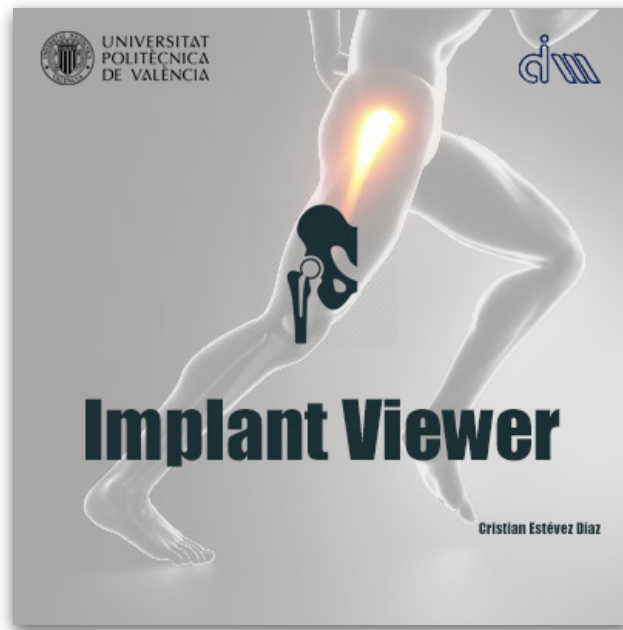


Figura 33.- Imagen de inicio de *Implant Viewer*

7.1.1. Consideraciones Previas.

La tesis que ocupa este documento se basa, principalmente, en el desarrollo de una herramienta capaz de facilitar la ubicación de objetos en un archivo *DICOM*. Esto permitirá que, tras un análisis posterior en *FEAVox*, se abran las puertas a múltiples aplicaciones, destacándose esencialmente las referentes al campo específico de la medicina.

Inicialmente, el desarrollo de *Implant Viewer* se planteó con tres objetivos principales en mente:

- El software debía ser sencillo de utilizar, en previsión de que lo empleen diferentes tipos de usuario.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

- Se debía asegurar una colocación y orientación espacial del implante lo suficientemente precisa.
- Se debía asegurar la total compatibilidad con *FEAVox*, de cara a facilitar un posterior análisis modal de las imágenes médicas.
- El *output* de *ImplantViewer* debía ser una imagen médica donde los vóxels del tejido biológico son sustituidos por vóxels con valor en la escala *Hounsfield* correspondiente al del material de la prótesis que ocupará dichos vóxels.

En términos generales, la idea preliminar de *Implant Viewer* era la de desarrollar una herramienta útil y sencilla con la que poder ensalzar las características del código de elementos finitos de *FEAVox*, además de poder abrir nuevas líneas de investigación biomecánica mediante los análisis numéricos propios de la ingeniería convencional.

7.1.2.Principio de Funcionamiento de *Implant Viewer*.

El posicionamiento y orientación de un objeto sobre una imagen presenta, de entrada, un gran problema que solventar, ya que a priori el objeto no se encuentra dividido en unidades mínimas procesables, como podría ser un píxel o un vóxel (en el caso tridimensional). Además, aunque se consiguiera discretizar la geometría del objeto, queda por abordar un problema importante: la forma de manipular los datos sobre la imagen médica. Estos dos problemas iniciales se solventan a partir de este software, y constituyen la base de funcionamiento de todo el código.

El voxelizado puede considerarse como una aproximación mediante componentes elementales de la forma geométrica del implante, o lo que es lo mismo, aplicar sobre el objeto una malla de elementos finitos cuyos componentes (hexaedros) tengan las

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

dimensiones reales de la imagen que se desea modificar. De este modo, se consigue mantener las escalas dimensionales entre los dos tipos de datos, a la par que se unifica el tratamiento de los datos a un solo procedimiento.

Dada la complejidad asociada a la programación de una malla de estas características, se ha optado por realizar el mallado del objeto CAD en el software externo de pre/post procesado, *GiD* (véase la sección 6 de este documento), que permite, entre otras cosas, crear mallas cartesianas con elementos tipo hexaedro de idénticas dimensiones.

Una vez el objeto ha sido voxelizado, se debe establecer una manera de manipular los datos de la geometría para introducirlos en la imagen médica. La representación tridimensional de ambos tipos de datos en *Matlab* implica cierta complejidad computacional, además de aumentar las dificultades de ubicación del objeto. Por este motivo resulta más sencillo tratar el conjunto mediante tres planos de corte principales (véase la sección 5.4, en la que se tratan las particularidades sobre los archivos *DICOM* y sus sistemas de referencia). El empleo de estos planos de corte implica un mayor control sobre la posición del implante, así como un menor coste computacional.

Dado que la imagen determinada por un plano de corte puede ser dividida en función de sus píxeles, existe la posibilidad de usar éstos como coordenadas, de tal manera que el objeto (con sus datos transformados a este tipo de unidades) pueda dibujarse sobre la imagen médica sin modificarla inicialmente. La introducción de la geometría se haría efectiva mediante el cambio de los valores *Hounsfield* contenidos en aquellas posiciones de la matriz *DICOM* que están determinadas por las coordenadas de los vóxeles del implante.

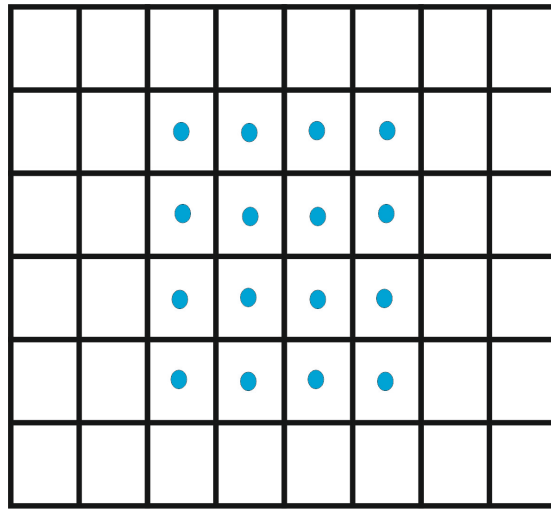


Figura 34.- Representación del tratamiento de datos en *Implant Viewer*. En azul se muestran los puntos del implante. Los rectángulos representan la imagen médica (uno de sus planos de corte).

Como parte del planteamiento esencial en la programación de *Implant Viewer* cabe destacar, por último, que los ejes que componen el sistema de referencia sobre el cual parte el origen de coordenadas siguen las direcciones determinadas por el estándar DICOM (sección 5.4), lo que se identifica con un sistema levógiro.

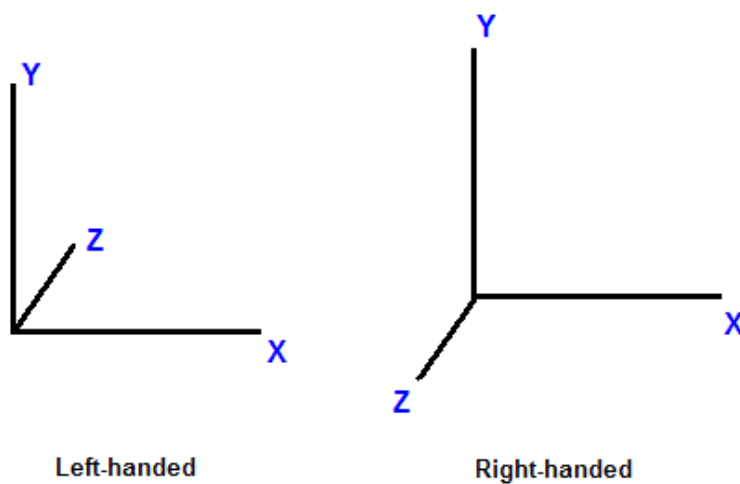


Figura 35.- Sistema de referencia levógiro (izquierda) y dextrógiro (derecha)

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA**7.1.3. Matrices de Datos Principales.**

El funcionamiento esencial de *Implant Viewer* se basa en la manipulación, modificación y guardado en matrices de una serie de datos esenciales. Estos datos esenciales son: matriz topológica, matriz de coordenadas del implante, matriz de datos de imagen, matriz de coordenadas de elementos externos y matriz de rotación.

- **Matriz topológica:** Esta matriz, denominada *top* en el código del programa, contiene la información relacionada con la numeración de elementos y su correlación con la identificación de los nodos que pertenecen a cada uno de ellos.

TOP

Elemento 1	Elemento 2	Elemento 3	...
Nodo A1	Nodo A2	Nodo A3	...
Nodo B1	Nodo B2	Nodo B3	...
Nodo C1	Nodo C2	Nodo C3	...
Nodo D1	Nodo D2	Nodo D3	...
Nodo E1	Nodo E2	Nodo E3	...
Nodo F1	Nodo F2	Nodo F3	...
Nodo G1	Nodo G2	Nodo G3	...
Nodo H1	Nodo H2	Nodo H3	...

Figura 36.- Representación simplificada de la estructura de la matriz *TOP*

- **Matriz de Coordenadas del Implante (xyzm):** Es el resultado del tratamiento de los datos obtenidos a raíz del archivo de malla generado con GiD. La matriz contiene información sobre la ubicación del centro de cada elemento en coordenadas de píxeles sobre la imagen médica.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

- **Matriz de Datos de Imagen (*Matrix*):** Se trata de la matriz que guarda la información referente a los valores, en la escala Hounsfield, de cada píxel que representa la imagen médica del paciente.

Matrix

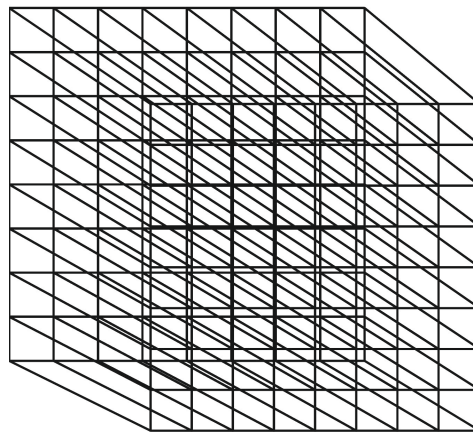


Figura 37.- Representación simplificada de la estructura de la matriz que contiene la imagen *DICOM*

- **Matriz de Coordenadas de los Elementos Externos (*xyzext*):** Ofrece las coordenadas (en términos de píxeles) de los elementos que conforman la frontera externa de la geometría del implante. Se usa fundamentalmente para mostrar de una manera rápida la información de la ubicación y orientación del implante a través de las ventanas gráficas. Por tanto, esta matriz será sometida a las mismas modificaciones que la matriz de coordenadas del implante (*xyzm*).

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

XYZ

X1	X2	X3	X4	X5	X6	...
Y1	Y2	Y3	Y4	Y5	Y6	...
Z1	Z2	Z3	Z4	Z6	Z6	...

Figura 38.- Representación simplificada de la estructura de las matrices cuyo nombre comienza por xyz

- **Matriz de rotación (xyzref):** Contiene vectores unitarios perpendiculares entre sí que son sometidos a los mismos giros que el implante. Esto permite obtener en todo momento una matriz de rotación que puede ser utilizada para conocer en cualquier instante la orientación del objeto en el espacio (ángulos de euler).

XYZREF

x	1	0	0
y	0	1	0
z	0	0	1

Figura 39.- Representación simplificada de los valores iniciales de la matriz de rotación (xyzref)

Además de estas cinco matrices elementales del programa, *Implant Viewer* dispone de otros muchos conjuntos de datos que facilitan el funcionamiento del mismo. En términos generales, todas aquellas matrices cuyo nombre comience por “xyz” se refieren a

coordenadas de algún tipo, mientras que en las restantes, su denominación suele ser un indicativo del tipo de datos que contiene.

7.1.4.Lectura de Archivos.

Implant Viewer necesita dos inputs principales para iniciar el proceso de colocación del objeto: un archivo que contenga los datos de imagen y un archivo de malla. En primer lugar, el programa pedirá la información perteneciente a la imagen, que puede estar en formato *DICOM* o *MAT*. La lectura de éste tipo de archivos resulta sencilla, pues se utilizan las propias funciones internas de *Matlab* para obtener los datos. Si el número de archivos de imagen es mayor que uno, el programa automáticamente forma la matriz de datos a través de la unión de las capas que va leyendo.

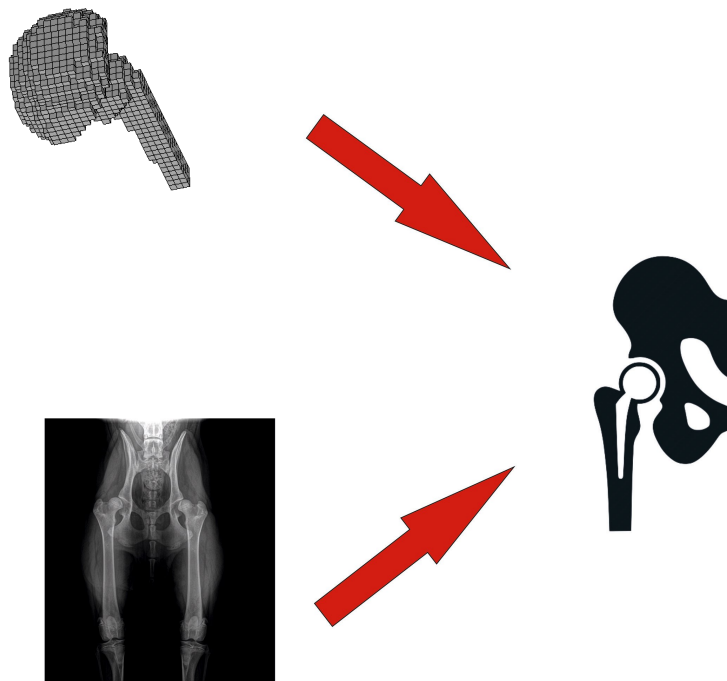


Figura 40.- Inputs necesarios para el funcionamiento de *Implant Viewer*

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

No obstante, cobra especial interés analizar el modo de lectura y tratamiento de datos de la malla del implante, pues resulta fundamental para abordar otras cuestiones relacionadas con la posición y orientación del objeto.

Por defecto, el programa empleado para la discretización del implante (GiD) crea un archivo de malla que contiene, entre otras cosas, las coordenadas de los nodos de malla en unas ciertas unidades de medida, que no tienen por qué corresponderse con milímetros o metros. Estas unidades deben ser transformadas, ya que se deben compatibilizar con los datos provenientes de la imagen médica.

Cuando se tiene una imagen con un objeto en algún punto de la misma y se desea tener dicho objeto ubicado con respecto a algún sistema de referencia, resulta sencillo, al no disponer del factor de escala real del objeto, dividir el continuo de la foto en cuadrantes que permitan relacionar el sistema de referencia con la ubicación del objeto. En *Implant Viewer* esos cuadrantes son los propios píxeles de la imagen, y la ubicación de cada píxel es determinada por su distancia en píxeles con respecto al origen de coordenadas. De esta manera, la localización y modificación de la información contenida en uno en concreto es más sencilla.

Por este motivo, y dado que la información proveniente de la malla viene dada en unas unidades no determinadas, se necesita convertir estas unidades a píxeles. De este modo se puede ubicar con mayor facilidad el objeto en la imagen. El proceso de conversión se detalla a continuación.

Inicialmente se obtienen, a partir de los datos del archivo de malla, dos matrices: la topológica y una de coordenadas de nodo en las unidades de GiD (véase la sección 7.1.4). Con estas dos matrices iniciales, se inicia el proceso de transformación de coordenadas a dimensiones en píxeles. En primer lugar, se construye una nueva matriz de coordenadas (xyz), cuyas columnas indican el número de nodo al que se hace referencia.

Posteriormente se calcula un vector denominado *delta*, que contiene las dimensiones en unidades de *GiD* del primer elemento que aparece en el archivo de malla. Esas dimensiones se consideran representativas del tamaño general de los hexaedros de la

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

mallado, pues en el proceso de mallado ha debido imponerse la igualdad de todos los cubos generados.

$$\Delta = \begin{Bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{Bmatrix}$$

El vector *delta* se emplea para generar otra matriz que contiene las coordenadas de los centros de cada elemento de la malla (*mctr*). Esta matriz se puede estimar sabiendo que la numeración de los nodos de elemento en GiD sigue una nomenclatura sistemática. Esto permite coger el nodo inferior de cada uno de ellos y sumarle la mitad de las distancias *delta* calculadas en el paso anterior.

En este momento, ya se dispone de las coordenadas de los centros de elemento, que se corresponderían con las coordenadas de los píxeles del objeto. No obstante, todavía queda el paso de transformar las unidades a píxeles. Si se simplifica el problema a una sola dimensión, se tiene que las coordenadas de los píxeles estarán alejadas con respecto a un sistema de referencia cualquiera situado en el mismo eje de referencia. Por lo tanto, el primer paso será situar el conjunto de datos en el origen de referencia. Para ello, se debe restar al conjunto el mínimo valor de coordenadas existente.

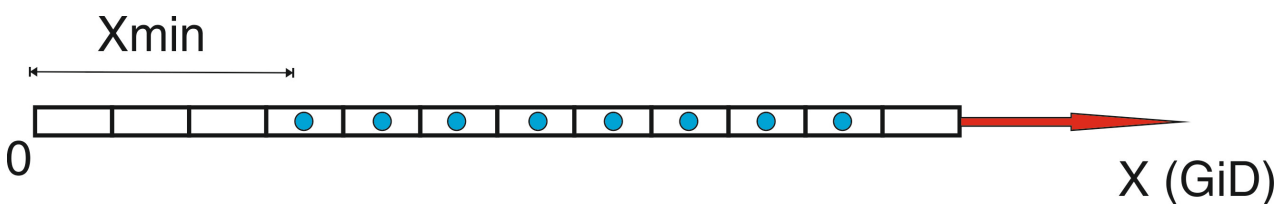


Figura 41.- Transformación de coordenadas entre las propias de GiD y unidades en píxeles. En cada una de las tres dimensiones del espacio se resta el valor de la mínima coordenada para que el implante se sitúe siempre en la parte positiva de los ejes en píxeles.

Por último, se dividen estos nuevos datos individualmente entre el incremento de distancia real que representa un píxel (δx), se redondean a cero decimales y se les resta

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

media unidad a cada uno de ellos. Con esto se consigue tener el punto central del elemento justo en el centro de un píxel de la imagen médica. De esta manera se hace efectiva la conversión de unidades, con lo que se facilita la manipulación del objeto dentro del marco determinado por las dimensiones de la imagen médica.

7.1.5. Movimientos del Implante.

Como se indicó anteriormente, el principio de funcionamiento del software emplea un sistema de coordenadas en píxeles para definir la ubicación del implante sobre la imagen médica. Este planteamiento simplifica notablemente los procesos de manipulación de los datos de la geometría, pues simplemente hay que modificar las coordenadas de los puntos que representan los centros de elementos, sin modificar los datos contenidos en la matriz que contiene la imagen médica.

Gracias a esta solución, el hecho de tener que trasladar el implante se traduce en sumar a las coordenadas del objeto una determinada distancia (en píxeles) a lo largo de uno o varios ejes de referencia (definidos según el estándar *DICOM*). En términos generales, *Implant Viewer* realiza los giros teniendo en cuenta las indicaciones del usuario a través de las ventanas de vistas que contienen los planos de corte (véase el *Anexo I. Manual de Usuario de Implant Viewer*).

En lo referente al giro del objeto, se ha ideado un procedimiento sencillo. Como se explicó en apartados anteriores, la ubicación y orientación del implante resulta más sencilla si se realiza a través de tres planos de corte (coronal, sagital y axial) que pueden ser colocados por el usuario. En este sentido, el giro de los píxeles del implante se realiza individualmente sobre el vector perpendicular a cada plano, esto es, se realiza de manera independiente a lo largo de los tres ejes que conforman la referencia y no de manera simultánea.

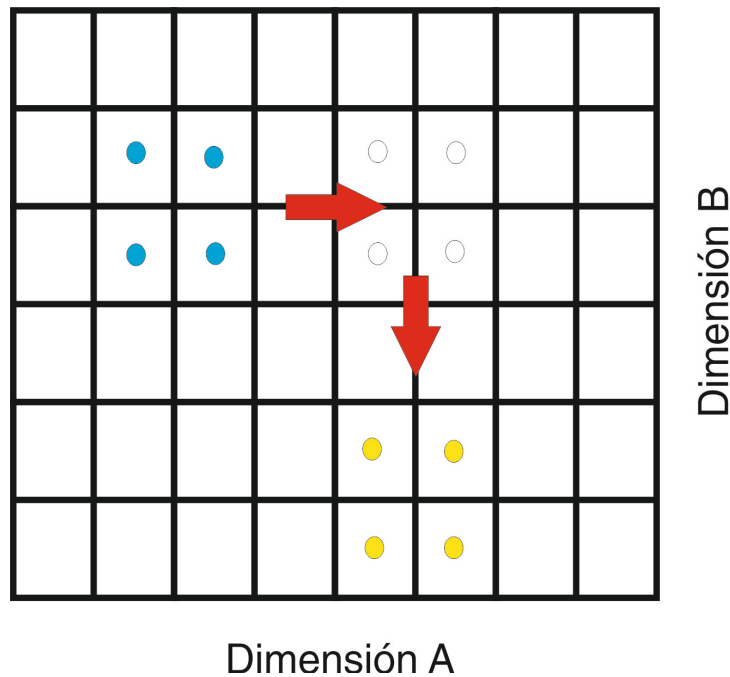


Figura 42.- Representación del proceso de traslación de la prótesis en *Implant Viewer*. En amarillo se aprecia la posición final. Se trata de una suma y resta de coordenadas en las dimensiones oportunas.

Este planteamiento, si bien a priori no parece la mejor opción, permite una orientación rápida del objeto si se le dota con una característica elemental: la posibilidad de que el usuario determine el píxel con respecto al que girar el conjunto en el plano que esté considerando. Esto hace que el software posea un gran dinamismo de cara al usuario, y permite una mayor agilidad de todo el proceso.

Desde el punto de vista de la programación el giro se realiza de una forma muy simple. Una vez determinado el punto o píxel de referencia, se calculan las distancias de todo el implante sobre el plano de giro correspondiente y se estiman las nuevas coordenadas de cada píxel de objeto mediante la aplicación de las ecuaciones paramétricas de una circunferencia. De este modo, la rotación de la geometría conserva las distancias iniciales entre vóxels de implante.

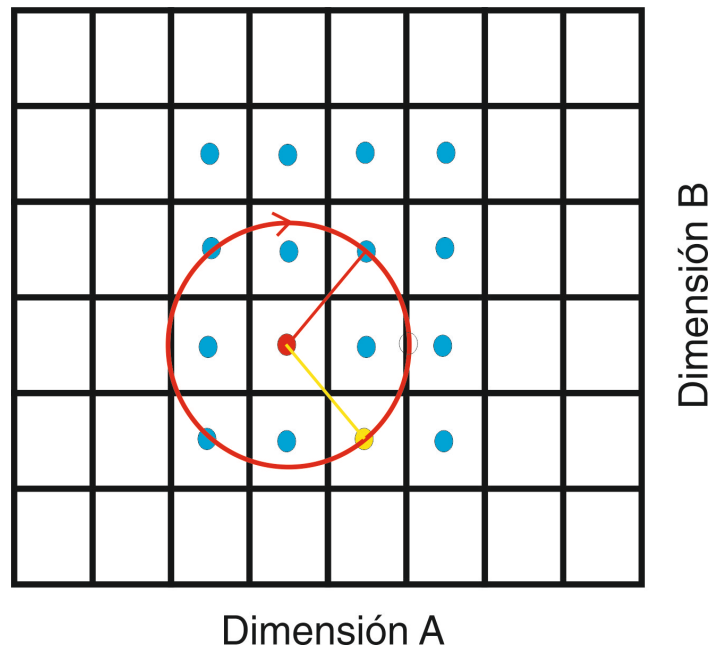


Figura 43.- Representación del proceso de rotación de la prótesis en Implant Viewer. Se especifica un píxel como centro y se aplica la ecuación de una circunferencia para calcular la nueva posición (amarillo)

7.1.6. Problemas Derivados del Movimiento.

Los algoritmos que controlan los movimientos del implante, si bien están bien definidos, presentan algunos problemas de concepto que se ven en la práctica y no tanto en la teoría. Estos inconvenientes se han ido solventando en el desarrollo propio del software, pero es importante conocerlos para entender correctamente las rutinas presentes en el código del programa.

Los movimientos del objeto se realizan siempre en el marco de coordenadas que aporta la imagen médica, y el sistema de referencia asociado, pero existen algunos factores externos e internos que hacen que en muchas ocasiones el movimiento deseado no sea el correcto o bien que el implante no modifique totalmente la imagen. Estos problemas se detallan a continuación y se refieren a situaciones en las que dimensiones del objeto difieren

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

con la realidad o con los límites establecidos por la imagen o a las dificultades de cálculo asociadas al proceso. Los problemas más comunes son:

- **El objeto sobresale de los límites establecidos por la imagen:** Este caso se puede dar en el uso habitual de *Implant Viewer*, y se debe normalmente a que en la posición del objeto alguna de las dimensiones del mismo es mayor que el número de píxeles en esa misma dirección de la imagen. El propio programa genera un aumento de píxeles de imagen en esa dirección con valor nulo, salvo los pertenecientes al implante, con la excepción de que no va a originar píxeles para valores de coordenadas negativas.
- **Píxeles de imagen con dimensiones desiguales:** Esto ocurre en la mayoría de archivos *DICOM* y se debe a que la toma de imágenes en un *TAC* se realiza por capas de datos apilados, lo que a su vez origina que los vóxeles normalmente posean dos dimensiones iguales y una diferente (generalmente Z). El problema que genera esto es que al girar el implante en un plano que contenga la dimensión que es desigual, las distancias de giro no se calculan correctamente, en términos de dimensiones reales del objeto, y éste tendría unas medidas que difieren de las originales. La solución a este problema se implementa mediante la sustitución de las ecuaciones paramétricas de una circunferencia por las correspondientes a una elipse, en las que la relación a/b sea la correspondiente a las dimensiones reales de un vóxel.
- **Vóxeles huecos:** La inclusión de las ecuaciones de una elipse para realizar un giro provoca otro inconveniente, que es que las dimensiones del implante en píxeles aumentan y disminuyen en función del ángulo girado. Esto en realidad debe ser así, pues de esa manera se asegura el mismo factor de escala dimensional en términos reales. No obstante, a la hora de modificar la imagen

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

médica, la nube de puntos genera huecos al modificar la imagen médica, con lo que los resultados del posterior análisis modal serían erróneos. La solución a este error pasa por realizar el giro en GiD de nuevo y reintroducir la malla del objeto (la orientación del objeto puede obtenerse por ángulos de Euler en *Implant Viewer*, sección 7.1.9) o utilizar la rutina de reparación propia del software, denominada *hfix* (véase el ANEXO I).

- **Ralentización del cálculo del movimiento:** En los casos en los que la malla del objeto es muy densa, es decir, tiene muchos elementos, los algoritmos de movimiento parecen ralentizarse, por el elevado coste computacional. Por este motivo, de cara al usuario el software muestra en pantalla solamente los elementos frontera del objeto, que son los primeros que mueve, mientras que a bajo nivel continua calculando con la totalidad de la nube de puntos.

La resolución de estos problemas ha marcado claramente el estilo de programación empleado en la fase de desarrollo del programa, y el conocimiento de estos errores derivados del movimiento puede ser de utilidad en otros proyectos o en futuras mejoras del propio programa.

7.1.7. Tratamiento de Imágenes en *Implant Viewer*.

El código implementado en este software, si bien no está pensado para la modificación pura de imágenes médicas, presenta algunas características útiles que permiten tratar y modificar levemente la imagen que se pasará posteriormente a *FEAVox*. Se destaca en esta sección su uso para facilitar al lector la comprensión de los algoritmos empleados en la fase de desarrollo.

En el apartado anterior se desarrolló el problema de los vóxeles huecos debidos al tipo de giro que se realiza en el programa. Este problema puede solucionarse, mediante el

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

uso de una de las rutinas de tratamiento de datos de imagen que incorpora *Implant Viewer*, la denominada *hFix*.

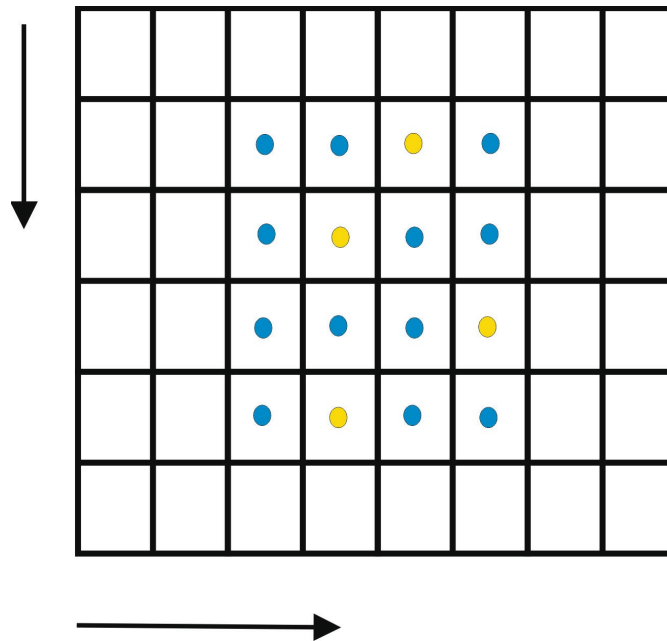


Figura 44.- Dirección de actuación de la rutina *hFix* durante el proceso de reparación de una imagen *DICOM*. En amarillo se muestran huecos donde esta rutina colocaría un píxel. Las flechas indican el recorrido de verificación que haría *hFix*.

El procedimiento en cuestión depende de si se ha movido o no la nube de puntos después de haber modificado la matriz *DICOM*. Lo ideal es que esa nube de puntos no se modifique, pues la rutina la emplea para detectar las posiciones de los datos que debe corregir. Una vez detectadas las mencionadas posiciones, el proceso de reparación consiste en localizar capa por capa (en orden ascendente de *z*) los píxeles de implante y reparar los huecos mediante la comprobación de los valores *Hounsfield* de los píxeles vecinos. Esto suele generar un objeto igual al original.

Además de estos procesos de corrección de errores, hay otro posible problema derivado del tamaño de los datos de imagen que puede solventarse desde el propio *Implant Viewer*, y se corresponde con la complejidad de cálculo asociada a *FEAVox* cuando se introducen imágenes de grandes dimensiones. En este sentido, y con la meta de reducir

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

el coste computacional, se han tenido que desarrollar una serie de herramientas de recorte de imagen.

Implant Viewer posee dos rutinas de recorte de imagen, en función de las necesidades del usuario. Son las denominadas *bCut* y *pCut*. La primera realiza una detección de la zona de la imagen en la que se encuentra la nube de puntos (por lo que ésta debe estar en la zona oportuna) y recorta de la matriz *DICOM* un cubo de datos de dimensiones iguales a las máximas del implante en las direcciones de los ejes de referencia. Por lo tanto, se obtiene una pequeña matriz que solamente contiene el tamaño justo para albergar los datos Hounsfield propios del objeto y los pertenecientes a la imagen que puedan estar más cerca.

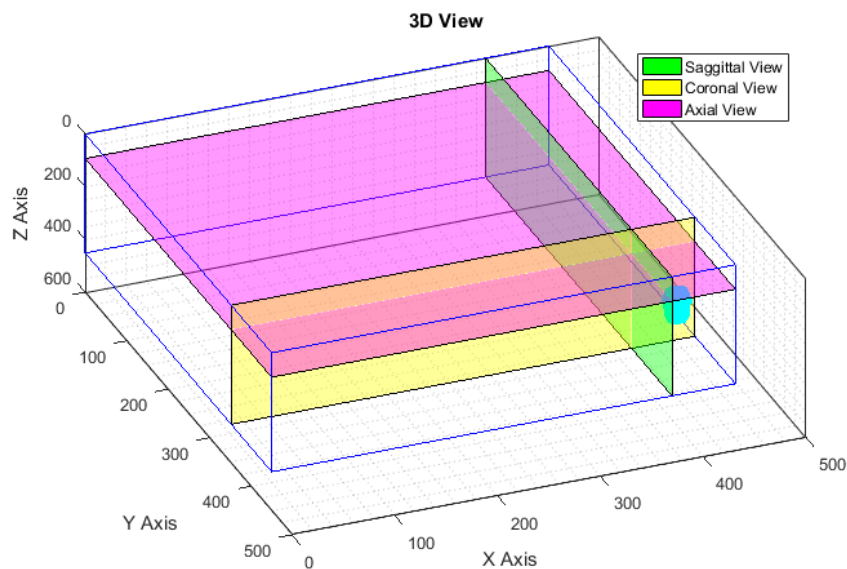


Figura 45.- Planos de visualización en *Implant Viewer*. Pueden ser utilizados como planos de corte para las rutinas *pCut* y *bCut*.

Por otro lado, la segunda rutina de recorte de imágenes (*pCut*) se ha implementado para cubrir el caso en que se necesita un mayor número de datos del tejido biológico que rodea el implante. Al igual que en el caso anterior, depende de la posición actual de la nube de puntos que representa al objeto, por lo que es necesario que ésta se sitúe en el lugar correcto. El funcionamiento elemental hace uso de la posición de los planos de corte para

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

recortar la imagen, es decir, el algoritmo detecta dónde están los puntos en relación a los planos y emplea éstos como límites de corte para generar una nueva matriz de datos recortada. De este modo, modificando la posición de los planos pueden obtenerse matrices resultantes de dimensiones variables en función de los deseos del usuario.

Todas las opciones de tratamiento de imágenes aquí mostradas no dejan de ser complementos al funcionamiento básico del software. No obstante, su programación resultó útil para hacer de *Implant Viewer* un programa más independiente y efectivo en el objetivo principal que se le había encomendado.

7.1.8. Renderizado 3D. Marching Cubes.

La adecuada representación gráfica de los datos que se estudian es algo a tener en cuenta, sobre todo en aquellas aplicaciones que requieren de la información visual que estos datos pueden aportar. La representación volumétrica de datos es actualmente de gran utilidad en la investigación científica (matemáticas, medicina, ingeniería...), y es utilizada para representar datos que pueden provenir de sensores, como en el caso de tomógrafos o hasta de satélites. Existen algunos algoritmos que permiten obtener representaciones de datos, pero para el caso que ocupa este documento solamente se destacará *Marching Cubes.*, ya que esta rutina se ha incorporado en *Implant Viewer* con el objetivo de facilitar al usuario la comprensión espacial de lo que está realizando sobre las imágenes definidas por los planos de corte principales. [10]

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Figura 46.- Muestra de la representación tridimensional por superficies que realiza *Marching Cubes*

Marching Cubes es uno de los algoritmos de rendering volumétrico más difundido y utilizado. Su funcionamiento se basa en estimar superficies a partir de una matriz volumétrica de datos escalares. Su origen se remonta a 1987, cuando fue propuesto por *Lorensen* y *Cline*. Su función principal es extraer una malla poligonal de isosuperficies en un campo de vóxels escalares en el espacio.

Un vóxel, unidad cúbica mínima procesable que compone un objeto tridimensional, está delimitado en el espacio por 8 vecinos próximos. La posibilidad de que en ese vóxel exista una superficie umbral viene determinada por los valores escalares de sus vecinos. Así, si uno de ellos tiene un valor inferior al establecido como umbral y el otro ligeramente superior podrá estimarse dicha superficie. Cada celda se clasifica en diferentes configuraciones según los valores de sus vértices respecto al valor umbral. Una celda pertenece a la superficie umbral si por lo menos uno de sus vértices está por debajo del valor umbral y por lo menos otro está por encima. En este caso, cada uno de los ocho vértices de una celda puede asumir un valor por debajo o por encima del umbral. El total de todas las configuraciones posibles es de 256, pero por consideraciones de simetría se reducen solamente a 15 (véase *figura 47*).

Este planteamiento, no obstante, ponía de manifiesto un problema importante en la construcción de la superficie umbral entre celdas vecinas. En algunos casos se forman

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

superficies con agujeros. Esto se produce fundamentalmente cuando dos celdas adyacentes comparten una cara, pero la conexión de los cuatro puntos que dividen las aristas se realiza en una de ellas de tal forma que los puntos con valor superior al umbral queden separados mientras que en la otra cara esos puntos quedan unidos. Hasta el momento se han resuelto varios casos en los que esto ocurre, pero no existe un remedio definitivo.

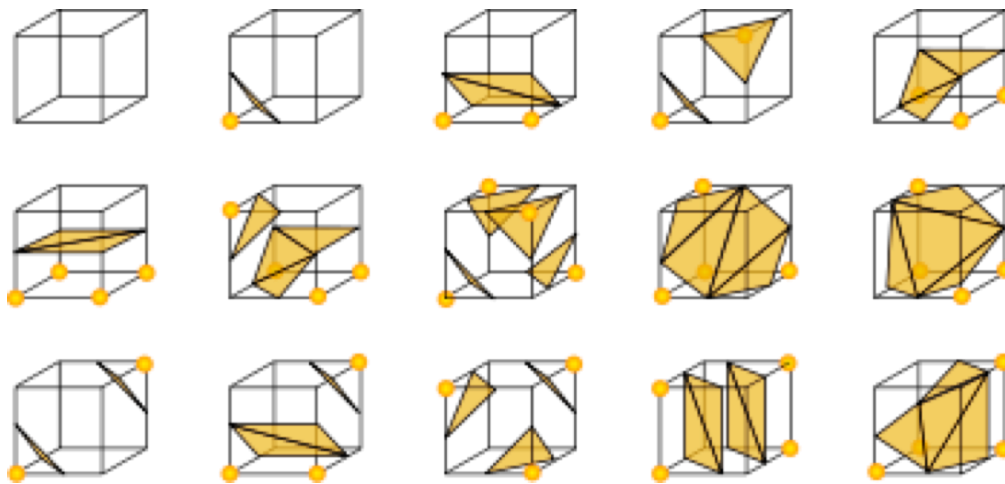


Figura 47.- Configuraciones de superficie de *Marching Cubes*

En *Marching Cubes* se clasifican las celdas que pertenecen a una superficie umbral. Dado un valor umbral y una rejilla cúbica conteniendo los datos del objeto a modelar, el algoritmo la procesa considerando cada celda en un orden determinado, generando localmente una isosuperficie, que se produce cuando el valor umbral queda acotado inferior y superiormente por los valores de los vértices de la arista. Cada arista puede tener, a lo sumo, una intersección con la isosuperficie.

El algoritmo, tal y como se planteó originalmente en los años ochenta, presenta un problema en algunas superficies generadas, ya que pueden aparecer superficies con agujeros. Esto es algo que se puede paliar modificando ese algoritmo original, pero no es el objetivo que ocupa a este *TFM*.

En el ámbito específico que ocupa a este documento, *Marching Cubes* es empleado en la representación 3D de la imagen médica del paciente, con el objetivo de facilitar al usuario la colocación y visualización del implante a partir de la interfaz gráfica desarrollada. Es destacable el hecho de que se ha implementado el algoritmo original, que está afectado por algunos problemas de continuidad de la superficie en determinadas condiciones. Esto no es un problema, ya que el objetivo de implementación de *Marching Cubes* es un mero complemento en el funcionamiento de esa interfaz gráfica.

7.1.9. Ángulos de Euler y la Orientación del Implante.

El problema de los píxeles huecos explicado en el apartado 7.1.6 implica que se deben efectuar correcciones de giro que permitan reintroducir en el programa un nuevo objeto que refleje las dimensiones reales del implante y sus propiedades mecánicas de manera uniforme. La primera alternativa es la utilización de los algoritmos internos propios de *Implant Viewer*, pero también es posible realizar un remallado en *GiD* y volver a leer el archivo de malla.

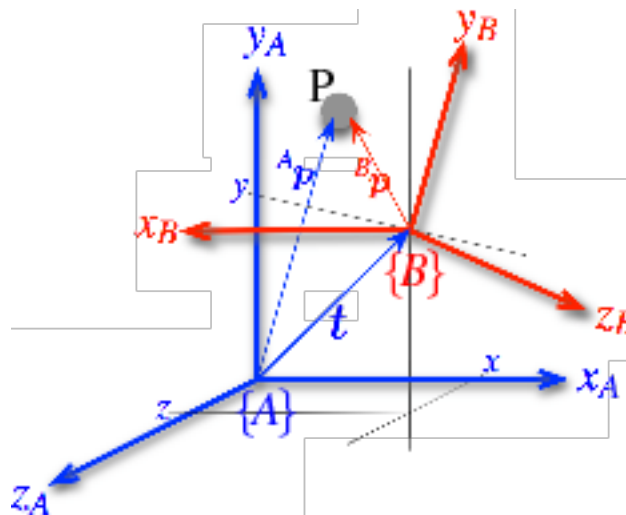


Figura 48.- Representación de la orientación de un punto a partir de las rotaciones realizadas sobre un sistema de referencia inicial

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

En el caso de que el usuario pudiera optar por el remallado en *GiD*, para asegurar totalmente que no va a haber fallos en la voxelización del implante, se ha incluido en el código una pequeña rutina externa que permite obtener los ángulos de Euler, derivados del correspondiente teorema de Euler. Para ello, se utiliza como input la matriz de rotación del programa (*xyzref*) y se obtienen los ángulos y la secuencia en que deben ser introducidos en *GiD*.

El teorema de Euler dice que cualquier rotación entre dos sistemas de referencia puede ser considerada como una secuencia de rotaciones alrededor de los diferentes ejes de coordenadas, de tal manera que dos rotaciones sucesivas no estén referenciadas al mismo eje de giro. Debe tenerse en cuenta que las rotaciones en las tres dimensiones no tienen la propiedad conmutativa, por lo que es muy importante el orden de las rotaciones individuales a lo largo de los ejes (véase la *figura 49*).

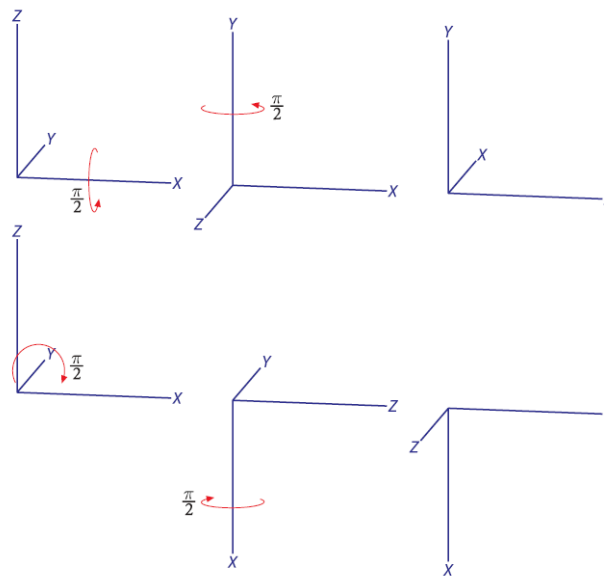


Figura 49.- Demostración de la importancia de la sucesión de giros sobre los ejes. En la parte superior se realizan las mismas rotaciones que en la inferior, sólo que el orden de giro sobre los ejes no concuerda.

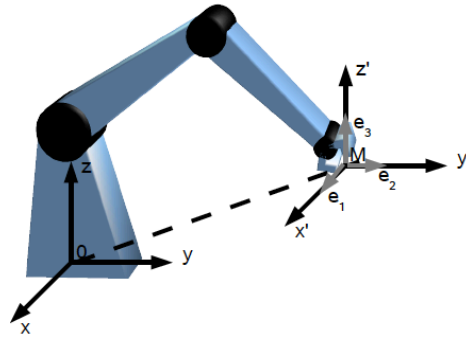
En un primer momento, *Implant Viewer* poseía su propio algoritmo interno de estimación de los ángulos de Euler, pero tras varias pruebas se demostró que, debido a las

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

singularidades propias de la definición de estos ángulos, existían algunas posiciones del objeto para las cuales los valores obtenidos no eran satisfactorios. Por este motivo, se decide incorporar una rutina externa que tenga en cuenta estas singularidades y avise al usuario en caso de que éstas se produzcan. De todas maneras, la secuencia de ángulos de Euler empleada en el software ha sido elegida para reducir al mínimo estos problemas.

A pesar de que la obtención de la rotación del objeto queda a cargo de un código externo, éste necesita un input para poder estimar los ángulos. Ese input es la matriz de rotación, cuyas características están explicadas en la sección 7.1.3.

La matriz de rotación se construye en el momento de inicialización del programa, y está basada en las coordenadas de tres vectores unitarios ortogonales entre sí que se someten a las mismas operaciones de rotación que los datos que representan el implante. De esta manera, al pasar a la rutina de cálculo los resultados de estos movimientos, es posible estimar con gran precisión la orientación del implante en cada momento.



$$\mathbf{R} = [\mathbf{e}_1 \quad \mathbf{e}_2 \quad \mathbf{e}_3] = \begin{bmatrix} e_{1x} & e_{2x} & e_{3x} \\ e_{1y} & e_{2y} & e_{3y} \\ e_{1z} & e_{2z} & e_{3z} \end{bmatrix}$$

Figura 50.- Construcción de la matriz de rotación a partir de la posición de los ejes del sistema de referencia final.

7.2. FEAVox.

El uso de las técnicas de análisis por *Elementos Finitos* (MEF) reduce significativamente el tiempo de desarrollo de muchos productos en la industria. Estas técnicas presentan ventajas claramente diferenciadas del resto de metodologías siempre y cuando el sistema sea discretizado con la precisión suficiente y se requiera la menor intervención posible por parte del usuario.

FEAVox es un software propio de la *Universidad Politécnica de Valencia*, actualmente en desarrollo y programado en *Matlab*, que fue concebido originalmente para ser una herramienta 3D sumamente eficiente en el análisis estructural por elementos finitos basada en el uso de mallas cartesianas h-adaptativas independientes de la geometría (método *cdFEM*), lo que le confiere la ventaja de que reduce los tiempo de mallado.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

En los últimos tiempos, este programa se está empleando para el desarrollo de una nueva herramienta de análisis que hace uso de los elementos finitos para realizar simulaciones personalizadas de prótesis, con el objetivo principal de determinar si una prótesis es óptima para un paciente. Este es el marco en el que se sitúa este trabajo fin de máster.

Para llevar a cabo estos análisis, se obtiene la morfología del paciente mediante *Tomografía Axial Computerizada (TAC)*. Esta geometría se discretiza mediante una malla cartesiana con información jerárquica que simplifica el refinamiento y acelera los cálculos. Las propiedades del tejido biológico, por su parte, son estimadas en función del color que estos presentan en la imagen médica, y se tienen en cuenta a la hora de generar la malla.

7.2.1. Estructura del Programa.

Antes de conocer los detalles intrínsecos de la programación del análisis modal en *FEAVox* es importante tener una idea clara de la estructura del sistema de archivos. Las subcarpetas se muestran en azul, mientras que los archivos aparecen en color verde. Cabe destacar que la estructura de la *figura 51* representa el esquema más esencial del software.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

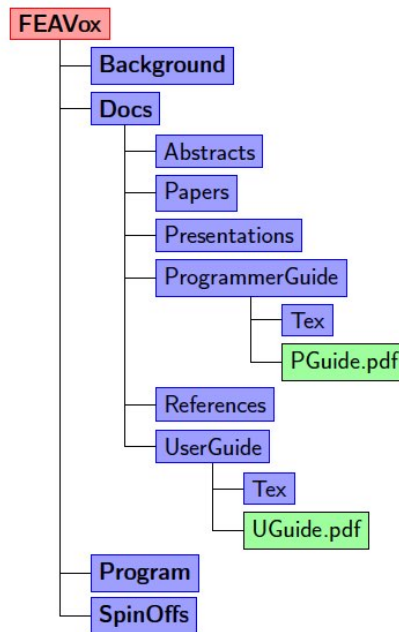


Figura 51.- Estructura de carpetas de *FEAVox*

Los contenidos de las carpetas se detallan a continuación:

- **Background:** En esta carpeta se tiene todo el software desarrollado previamente y que resulta relevante para el desarrollo actual de *FEAVox*.
- **Docs:** Aquí se encuentran subcarpetas en las que se pueden encontrar aquellos documentos relacionados con el proyecto.
- **Program:** Esta carpeta contiene todas las rutinas necesarias para ejecutar *FEAVox*, incluyendo los diversos tipos de problemas y resultados.
- **SpinOffs:** Aquí se guardan versiones antiguas pero estables del programa, con el fin de que los usuarios e investigadores del software puedan realizar pruebas

7.2.3. Estructura de Variables en FEAVox.

Si bien el sistema de archivos en *FEAVox* está bien definido, la forma en que éste guarda y muestra las diferentes variables que intervienen en los algoritmos internos no lo está tanto, por lo que se hace necesario establecer una idea general del funcionamiento del software en este sentido.

Principalmente, los datos en *FEAVox* se almacenan en variables tipo *struct* que agrupan toda aquella información que guarda cierta relación entre sí. De este modo, el acceso a los datos es relativamente más sencillo de comprender y de acceder. Cabe destacar que estas variables genéricas de datos estructurados están globalizadas, lo que permite facilitar el intercambio de información actualizada entre las funciones propias de *FEAVox*.

A continuación se desarrollan aquellas estructuras de datos que juegan un papel importante en el desarrollo del análisis modal que afecta a este documento.

- **Workspace:** Agrupa aquellos datos que tienen una influencia directa en los problemas numéricos de elementos finitos. Así, por ejemplo, se pueden encontrar las matrices $[K]$ o $[M]$, las condiciones de contorno definidas o el número de casos de carga.
- **KMatrix:** Contiene las variables y submatrices involucradas en el cálculo de la matriz de rigidez global del problema. Se pueden encontrar, entre otras cosas, matrices como $[B]$ o los datos propios de las rigideces de una malla de imagen.
- **MMatrix:** Esta estructura es análoga al caso inmediatamente anterior, pero con información asociada al cálculo de la matriz de masa. Es posible ver los datos de masa de los píxeles de una imagen o las funciones de forma empleadas en el cálculo $[N]$.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

- **Parameters:** Como su propio nombre indica, esta variable tipo *struct* engloba todos aquellos datos que determinan las características del problema que se analiza. Un ejemplo pueden ser datos relativos a la malla generada, el tipo de problema que se resuelve o variables de funcionamiento interno del programa.
- **BitMap:** Guarda información sobre los píxeles del problema, en el caso de que sea de tipo imagen. Se guardan datos sobre las propiedades mecánicas interpoladas de los píxeles en función de su color o los mismos datos de imagen analizados.
- **Results:** Se almacenan datos sobre los resultados obtenidos en la resolución del problema. En el caso del análisis modal que aquí se desarrolla, los resultados van a estar almacenados en la estructura *Results.Modal*.
- **Mesh:** Guarda todos los datos relativos a la malla que se ha realizado sobre el modelo. Es especialmente interesante tener en cuenta los datos de las posiciones nodales que aquí quedan almacenados (coordenadas nodales, topología...).

Durante la incorporación del análisis modal en el código fuente de *FEAVox* ha sido necesario leer o escribir datos en estas estructuras. Su comprensión previa puede facilitar la futura modificación por nuevos programadores.

7.2.4. Definición de un Análisis Modal.

El procedimiento de definición de un ejemplo de funcionamiento o un nuevo análisis es análogo para cualquier tipo de problema que involucre imágenes. En este apartado se

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

ilustrará sobre esa cuestión mediante un ejemplo aplicable al caso del análisis modal que concierne a este documento.

Los nuevos ejemplos de análisis deben situarse en dentro de la carpeta *Samples*. Deberá crearse una sub carpeta con la numeración adecuada según el tipo de problema al que hace referencia (en el caso de la figura 53 se corresponde con “41_JawModal”). Aquí se ubicarán tanto las imágenes médicas que contienen la geometría en formato *MAT* como los algoritmos de asignación de condiciones de Dirichlet y de propiedades físicas. Estas rutinas son esenciales en el funcionamiento de *FEAVox*, y deben ser definidas individualmente de manera previa a la ejecución del programa.

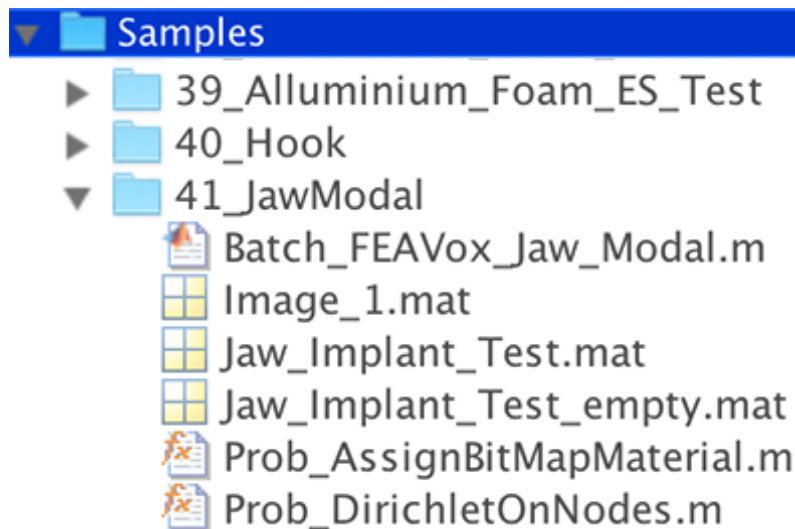


Figura 53.- Archivos necesarios para definir un problema nuevo en *FEAVox*

En la *figura 53* se muestra un ejemplo de un problema de imagen definido para análisis modal. En la carpeta en cuestión, ubicada dentro de *Samples* se pueden ver los archivos que son necesarios para la definición de este ejemplo. Hay varios archivos *MAT*, que contienen las imágenes en formato *DICOM*, y tres rutinas principales, que en el caso del ejemplo de esta figura son:

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

- **Batch_FEAVox_Jaw_Modal.m:** Define, para este problema, los parámetros de funcionamiento y se encarga de la ejecución del programa. Este algoritmo debe situarse dentro de la carpeta *Program*, para que se pueda ejecutar el software de manera correcta, pero se suele guardar una copia en la carpeta de problema. Cabe destacar que el nombre de esta rutina debe adaptarse al nombre de la carpeta de problema. En este *script* se define el nombre de la carpeta, la ruta del fichero a analizar y las opciones de ejecución (algunas se detallan al final de este apartado).
- **Prob_AssignBitMapMaterial.m:** Es la que define la correspondencia entre los colores de los píxeles de imagen y las propiedades físicas del material. Da como salidas los valores interpolados de estas propiedades para los valores Hounsfield detectados. Las interpolaciones se realizan en función de unos vectores dados para cada propiedad. El valor máximo del vector *Hounsfield* debe ser el correspondiente al *titanio*, que es el material del que usualmente se componen las prótesis. En el caso específico del análisis modal, la densidad de todos los materiales que no sean *titanio* deben ser prácticamente nulas, ya que el objetivo es analizar la variación de frecuencias naturales cuando el implante está rodeado por elementos hueso con rigidez pero sin masa.

```
HU= [1, 228, 229, 1500, 2000, 4000];  
Nu= [0.3, 0.3, 0.3, 0.3, 0.3, 0.3];  
E = 1e6*[0.1, 0.1, 500, 1500, 7000, 116000];  
rho= [0.1, 0.1, 0.1, 0.1, 0.1, 4507];
```

Ejemplo 1.- Ejemplo de vectores de referencia que dentro de la rutina *Prob_AssignBitMapMaterial.m*. En ellos se establecen relaciones entre los valores *Hounsfield (HU)*, el coeficiente de Poisson (*Nu*), el módulo de Young (*E*) y la densidad (*rho*).

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

- **Prob_DirichletOnNodes.m:** Define las condiciones de contorno del problema sobre los nodos de la malla. En el caso que ocupa este documento, por tratarse el análisis en vibraciones libres del sistema, no habrá condiciones de contorno, pero sí que habrá que indicarle al programa algo para que éste no de un error numérico. Normalmente basta con darle unas restricciones ficticias.

```
ConstrainedNodes=[1];  
DisplXYZ=zeros(1,3);  
ConstrainedDirection=zeros(1,3);
```

Ejemplo 2.- Condiciones de Dirichlet impuestas en los ejemplos de análisis modal. Las tres variables se corresponden con las tres salidas de la rutina *Prob_DirichletOnNodes.m*

Entre estos algoritmos situados en la carpeta de problema, cobra especial importancia el archivo principal (*Batch_FEAVox_Jaw_Modal.m* en este caso), ya que, al margen de los parámetros propios de control de malla y similares, se deben modificar algunas opciones para asegurar que *FEAVox* analiza el modelo adecuado y de manera correcta. A continuación se muestra un fragmento de código de la rutina de ejemplo, únicamente con las líneas que representan los valores que debe darse a esas opciones.

```
Batch_ProblemParameters(...  
    '41_JawModal',... % Nombre de la carpeta de problema  
    'Medical Modal'),... % Tipo de problema. En el caso modal debe ser obligatoriamente 'Medical Modal'  
Batch_AnalysisParameters(...  
    {'./Samples/41_JawModal/Jaw_Implant_Test.mat'},...%Ruta hasta el archivo de imagen  
    {'Image'},... % Tipo de componente, ('Geometry') o ('Image')  
    188*0.2,... % Dimensión original en X. Producto del número de píxeles en esa dirección por su dimensión  
    en mm.
```

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

180*0.2,... % Dimensión original en Y. Idem que en el caso en X

0.2,...% Slice Thickness. Tamaño del píxel en Z (mm). No se multiplica por el número total de píxeles en Z.

[1,188,1,180,1,221],...% CAT cutout. Píxeles de recorte de imagen. Debe dejarse con el tamaño original

{'Pixel'}, ... % Tipo de integración de imagen 'Pixel' or 'LS'

{'E'}, ... %Tipo de Relación imagen-propiedades del material

0,... %Uso de la interpolación por mínimos cuadrados

Batch_ProgramParameters(...

0,... % Uso de varios núcleos con *matlabpool*

Ejemplo 3. Líneas de código de Batch_FEAVox_Jaw_Modal. En la definición de cualquier problema nuevo de análisis modal de imagen estos parámetros deben tener estos valores, aplicados al campo específico del problema.

Por último, y al margen de los algoritmos esenciales, en este apartado es necesario destacar que los archivos *MAT* presentes en los problemas del tipo *imagen médica* deben contener una matriz de imagen denominada *Matrix*. Si esto no se realiza así, *FEAVox* dará problemas en la lectura de la imagen y se detendrá la ejecución del código. *Implant Viewer* está programado para hacer esto correctamente.

7.2.5. Incorporación del Análisis Modal al código de FEAVox.

En el caso que ocupa a este *Trabajo Fin de Máster*, el análisis modal se requiere para el estudio de imágenes médicas en los que al valor en escala *Hounsfield* de los píxeles se les asocian unas determinadas propiedades físicas. Por este motivo, la estimación de la matriz de masa se ha añadido simplemente para el caso de problemas de tipo imagen, dejando para un futuro la incorporación para otro tipo de problemas ejecutables en *FEAVox*.

Los análisis dinámicos, tal y como se detalló en la *sección 5.5.1*, requieren al menos del valor de la masa del modelo para poder realizar una estimación de frecuencias naturales. En el caso del *Método de los Elementos Finitos*, la masa se engloba dentro de

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

una matriz que relaciona los grados de libertad de los elementos con su estimación de masa correspondiente. La matriz de masa resultante es una discretización de la masa continua que existiría en el modelo real. En este apartado se detallarán de manera simplificada las pautas seguidas para la incorporación del análisis modal (basado en la implementación del cálculo de la matriz de masa) en el código fuente de *FEAVox*, partiendo de la base de que ya se ha definido un ejemplo para analizar, de acuerdo con lo descrito en la sección 7.2.4.

El proceso de evaluación de la matriz de masa $[M]$ guarda similitudes con la obtención de la matriz de rigidez $[K]$. En ambos casos, de cara a la resolución del problema, se deben calcular las matrices individuales de cada elemento de la malla, para luego ensamblarlos en una matriz global que represente fielmente el problema que se analiza.

El proceso de cálculo de las matrices elementales y su integración en el sistema global se lleva a cabo de forma parecida al caso de la matriz de rigidez. No obstante, ha de tenerse en cuenta que el cálculo de $[M]$ difiere ligeramente de la propia para $[K]$ para un mismo elemento.

$$[M^e] = \int_{V_e} \rho \cdot [N]^T \cdot [N] \cdot dV = \sum_{k=1}^{N_{int}} (H\rho [N]^T [N] \det [J])_k$$

La evaluación de la masa de elemento se realiza mediante las funciones de forma del elemento en cuestión, su densidad asociada y el volumen que engloba dicho elemento. Esto, en términos de integración numérica puede expresarse como una suma de los pesos de los puntos de Gauss (H) por la densidad (ρ), las funciones de forma $[N]$ y el determinante del Jacobiano $[J]$, que se encarga de las transformaciones de coordenadas locales a globales.

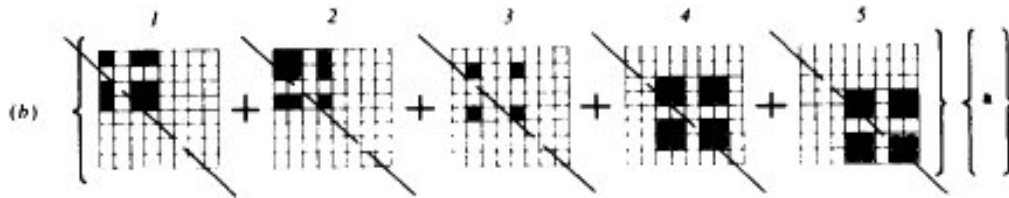


Figura 54.- Ensamblado de la matriz K , que es similar al correspondiente a la matriz M .

Sabiendo ahora cómo se calculan las matrices de masa elementales, el procedimiento de incorporación del análisis dinámico en el código fuente de *FEAVox* se basa en aplicarle a esas matrices elementales el mismo procedimiento. Las rutinas que se han modificado para calcular la matriz $[M]$ son:

- ***Intg_KMatrixBmpElements***: Éste es el algoritmo donde se calculan las matrices de rigidez elementales para los elementos de malla que representan vóxels. Al inicio del se código ha creado la estructura de datos *MMatrix*, para guardar los datos calculados relativos a la matriz de masa. Posteriormente, el proceso de estimación sigue los pasos propios del cálculo de $[K^e]$. Es destacable el hecho de que, por temas de optimización del código, las matrices de masa elementales se guardan todas en una matriz tridimensional (*MMatrix.BmpElements*) cuyas capas representan la matriz de masa de cada elemento.
- ***Sol_InitialAssembly***: En esta parte del código es donde se comienza con la integración de las matrices elementales para formar las globales que representan la totalidad del problema de análisis. En este punto, esencialmente se sigue el procedimiento de integración de $[K]$: se calculan la matriz simétrica global *MBasic* y se le aplican las restricciones multi-punto (*C_MPC*) asociadas a la malla h-adaptada

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

sobre el modelo. Cabe destacar en este punto que, pese a que esta opción estará desactivada por defecto, se ha habilitado en esta rutina la posibilidad de obtener una matriz concentrada de masa, en lugar de la consistente.

- **Sol_SystemAssembly:** En este algoritmo de *FEAVox* se culmina con el proceso de integración de *MBasic* y se calcula la matriz $[M]$, almacenada dentro de la estructura *Workspace.M*, análogamente de lo que pasa con $[K]$.
- **Sol_Resolution:** Esta rutina engloba la resolución del método de los elementos finitos habitual, pero dado que es la que escribe los resultados en las estructuras pertinentes contiene la resolución del problema de autovalores y autovectores que se corresponde con el análisis modal (véase la siguiente expresión). Cabe destacar que *FEAVox* solamente resolverá este apartado si el tipo de problema se define como *Medical Modal*.

$$([K] + \omega^2 [M])\{\bar{U}\} = \{0\}$$

7.2.6. Resultados y Representación de Modos de Vibración.

Los resultados del problema dinámico son dos. Por un lado se tiene un vector de longitud n y que representa el cuadrado de las pulsaciones (ω), y por otro una matriz con n vectores columna unitarios $\{\bar{U}\}$ que representan los modos de vibración, es decir, los desplazamientos asociados a cada frecuencia natural. Tanto las frecuencias naturales (valores positivos de las raíces cuadradas de los valores ω en Hz) como los modos de vibración se almacenan en la estructura de *FEAVox* denominada *Results.Modal*.

En lo referente a la representación sobre la malla de los modos de vibración obtenidos, se ha desarrollado, dentro de la rutina *Sol_Resolution*, una llamada a una función de ploteo denominada *Plot_Modal*, a la que se le especifica el número de modo a dibujar y

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

un factor de escala para poder controlar la amplitud de los movimientos de la malla. De esta manera es más sencillo comprobar la forma de los modos obtenidos

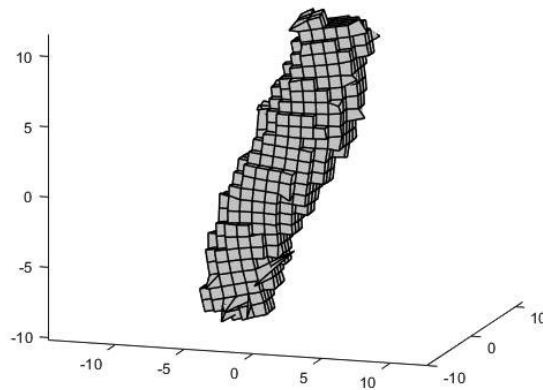


Figura 55.- Deformada típica obtenida con la función *Plot_Modal.m*

La función *Plot_Modal*, a su vez, lo que hace es sumar a las posiciones de los nodos de la malla el modo correspondiente multiplicado por el factor de escala, representarlo con la función *Plot_BitMapMesh* y volver a restar a las posiciones el modo en cuestión para que se pueda representar otro. Dentro de *Plot_BitMapMesh*, que es donde realmente se dibuja la malla, se puede controlar el dominio a representar, y por lo tanto, cortar la malla.

8. Ejemplos.

En esta sección se ilustran algunos ejemplos que muestran el proceso por el cual se puede determinar la viabilidad del método. El procedimiento que será mostrado en este apartado obedece a la utilización de los dos programas principales que se sitúan en el marco de este TFM: *Implant Viewer* y *FEAVox*.

La pauta general del método consiste en calcular las frecuencias naturales y modos de vibración del implante en la misma posición en la que estaría ubicado en la imagen, analizar la forma de los modos resultantes, y ver cómo cambian estos al analizar la unión prótesis-hueso. Se supondrá que la adhesión hueso-implante es perfecta.

8.1. Implante Dental.

Para ilustrar sobre el método desarrollado a lo largo de todo este documento, se plantea un caso práctico en el que se debe colocar una prótesis dental sobre la mandíbula de un paciente determinado. El objetivo principal sería el de evaluar, de manera previa a la intervención médica, el grado de sujeción esperado del implante completamente osteointegrado, o lo que es lo mismo, la viabilidad de la operación en función de las características del hueso del paciente.



Figura 56.- Ubicación del implante sobre la mandíbula del paciente.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

Se dispone de un *TAC* en formato *DICOM* de la mandíbula del paciente de 451x451x451 píxeles de 0,2 mm de lado en cada eje cartesiano, en la que se pretende simular un implante comercial.

8.1.1. Propiedades del Material asociadas a la Imagen.

Según la bibliografía basada en trabajos previos sobre este tipo de huesos humanos [13], se han establecido una serie de vectores con puntos de referencia que relacionan las propiedades del hueso, en función de su valor en la escala *Hounsfield (HU)*, con su coeficiente de *Poisson*, su módulo de *Young (E)* y su densidad (*rho*). Los vectores de datos son:

$$HU=[1,1150,1275,1400,1525,1650,1775,1900,5001];$$

$$Nu=[0.32,0.32,0.32,0.32,0.32,0.32,0.32,0.32,0.32];$$

$$E=1e9*[0.000001, 1.05, 2.37, 3.42, 5.04, 6.84, 9.21, 12.1,96]; \%Pa$$

$$rho=[0,0,0,0,0,0,0,0,4620]; \%Kg/m^3$$

Los últimos valores de estos vectores son los correspondientes al titanio, material plenamente biocompatible del que está hecho el implante. Cabe destacar también que se ha elegido despreciar la masa de los tejidos biológicos para evaluar únicamente el efecto de rigidización de la osteointegración sobre las frecuencias propias del implante.

Nota: Es importante destacar que el valor de la escala *Hounsfield* asociado al titanio debe ser una unidad mayor que el valor que tiene este material en la imagen. De lo contrario los resultados arrojados por *FEAVox* serán erróneos.

8.1.2. Discretización y Ubicación del Implante.

Una vez conseguido el modelo *CAD* de la prótesis, se le pasa a *GiD*, con el objetivo de discretizarlo mediante una malla cartesiana de dimensiones iguales a las del vóxel tipo de la imagen médica. Por las características del formato de los ejes de referencia en archivos *DICOM* (véase el apartado 5.4), es recomendable (aunque no necesario), girar el implante de manera previa al mallado, para que éste quede alineado en sentido negativo del eje *Z* que representa *GiD*.



Figura 57.- Modelo *CAD* del implante (izquierda) y malla cartesiana del mismo (derecha)

Una vez mallado el modelo tridimensional del implante, se debe generar con *GiD* un archivo **.msh* que refleje la topología y características propias de la discretización realizada. Esto se realiza de manera simple mediante el menú *File->Export->GiD mesh* disponible en el programa (para más información véase el apartado 6.3 de este documento).

Con el archivo de malla y la imagen médica del paciente, se inicia *Implant Viewer* y se comienza el proceso de posicionamiento y orientación del modelo sobre el hueso (para saber cómo realizar este proceso consulte el ANEXO 1 *Manual de Usuario de Implant Viewer*). En este caso, el implante se coloca en la zona inferior de la mandíbula.

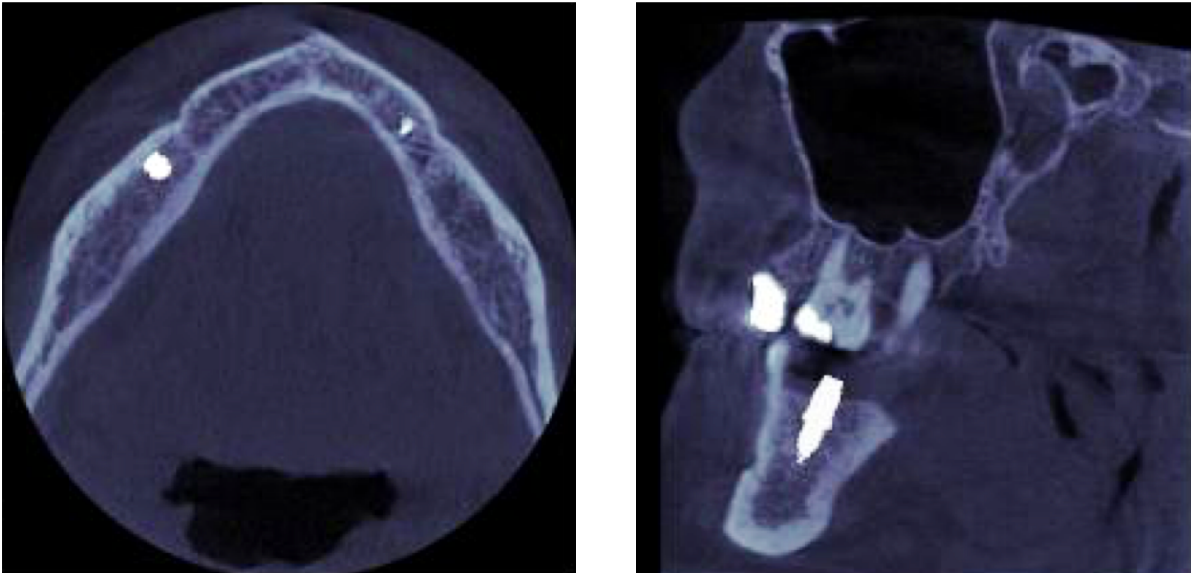


Figura 58.- Ubicación del implante sobre la mandíbula en la imagen médica.

De este proceso se obtiene como resultado un fichero **.mat* con la información de los valores en escala *Hounsfield* de cada uno de los vóxeles de la imagen, incluidos aquellos que han sido modificados para ubicar el implante.

8.1.3. Análisis Modal del Implante.

Un paso previo para determinar el grado de fijación del implante al tejido biológico del paciente es conocer de antemano cuáles son las frecuencias naturales y modos de vibración de la prótesis libre, sin implantar. De este modo, puede determinarse el margen de variación de los resultados obtenidos cuando se aplique el mismo análisis al conjunto hueso-implante. El análisis arroja los siguientes resultados:

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Modo	ω_n (Hz) FEAVox	ω_n (Hz) ANSYS
1	0	0
2	0	$2,989 \cdot 10^{-2}$
3	0	$4,255 \cdot 10^{-2}$
4	0	$5,183 \cdot 10^{-2}$
5	0	$5,411 \cdot 10^{-2}$
6	$2,901 \cdot 10^{-3}$	$7,621 \cdot 10^{-2}$
7	$4,12 \cdot 10^4$	$3,54 \cdot 10^4$
8	$4,273 \cdot 10^4$	$3,591 \cdot 10^4$
9	$7,876 \cdot 10^4$	$7,425 \cdot 10^4$
10	$8,683 \cdot 10^4$	$7,74 \cdot 10^4$
11	$9,234 \cdot 10^4$	$7,95 \cdot 10^4$
12	$1,312 \cdot 10^5$	$1,174 \cdot 10^5$

Tabla 3.- Frecuencias naturales obtenidas del implante

Como puede apreciarse, los seis primeros modos de vibración se corresponden con frecuencias naturales nulas, pues esos se relacionan con los modos de sólido rígido (tres desplazamientos y tres giros en el espacio). Cabe destacar que la forma de los resultados obtenidos se corresponde con lo que cabría esperar para un análisis en vibraciones libres, con lo que se valida el algoritmo de análisis modal incorporado en *FEAVox*.

Como validación de los resultados obtenidos a partir del procedimiento incorporado en *FEAVox*, se muestran en la *tabla 3* los resultados obtenidos en *ANSYS* a partir de la malla de elementos finitos realizada en *GiD*. Los valores obtenidos mediante ambos programas son del mismo orden de magnitud y muy parecidos. La diferencia entre los valores está en que en el caso de *ANSYS*, la malla del análisis está bastante más refinada. Más concretamente, la malla realizada con *FEAVox* coincide con la de *ANSYS* (ya que es la

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

misma que la de aproximación en *ANSYS*), pero la malla de aproximación de *ANSYS* es mucho más refinada que la de *FEAVox*.

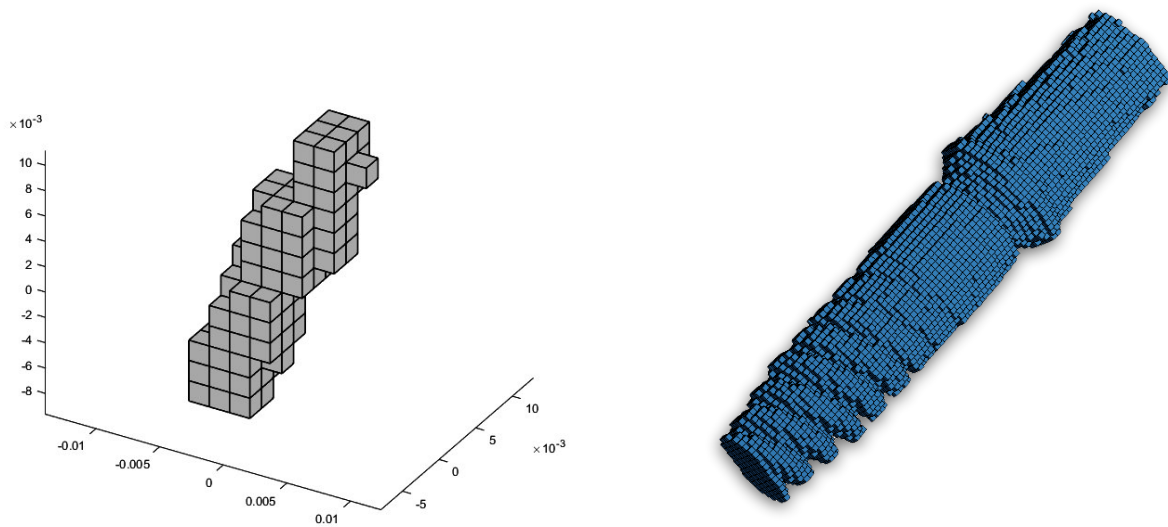
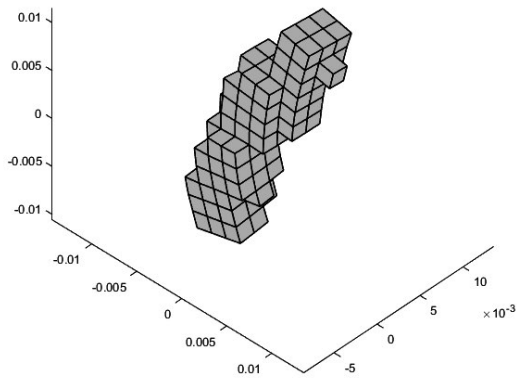


Figura 59.- Malla de discretización del implante en *FEAVox* (izquierda) y malla de integración introducida en *ANSYS* para el análisis

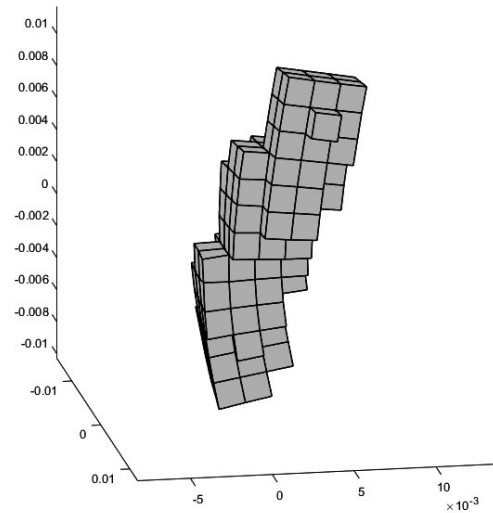
De cara a posteriores comparaciones con los resultados de la unión hueso-implante, se mostrará a continuación el dibujo de los modos de vibración asociados a las frecuencias naturales destacadas en negrita en la *tabla 3* y que no se identifican con los propios del sólido rígido. Reconocer estos modos facilitará la tarea de cuantificar el grado de sujeción del implante al tejido biológico del paciente.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

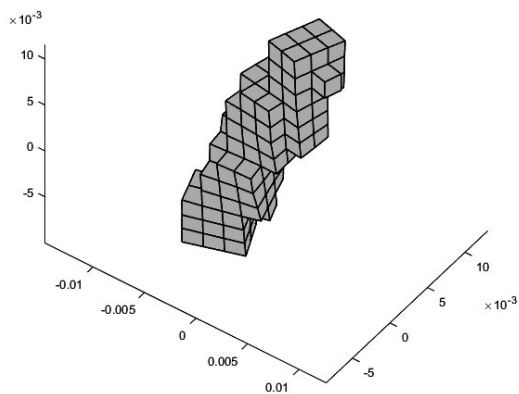
Modo 7: $4,12 \cdot 10^4$ Hz (Flexión)



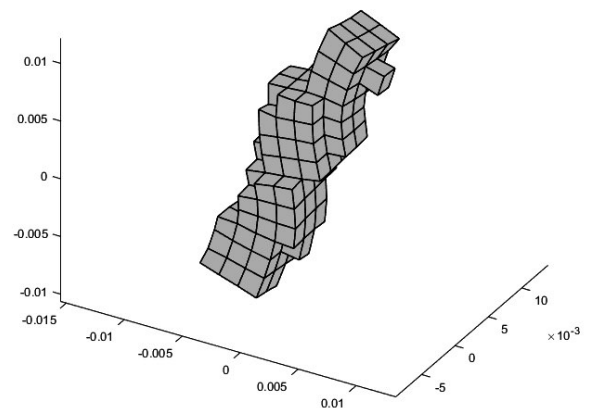
Modo 8: $4,27 \cdot 10^4$ Hz (Flexión)



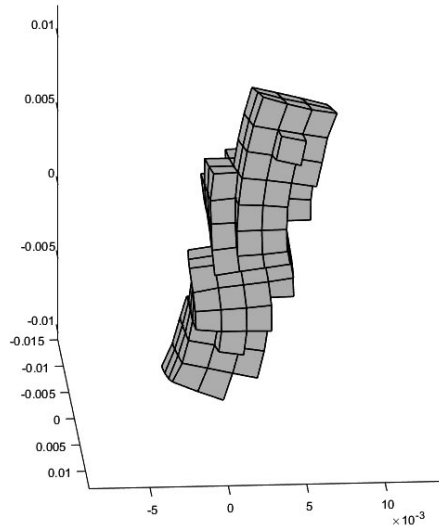
Modo 9: $7,87 \cdot 10^4$ Hz (Torsión)



Modo 10: $8,68 \cdot 10^4$ Hz (Flexión)



Modo 11: $9,23 \cdot 10^4$ Hz (Flexión)



8.1.4. Análisis de la Unión Implante-Hueso.

Una vez conocidos y validados los resultados del análisis modal en el caso del implante por separado, se exponen los correspondientes a la unión hueso-implante. Se muestran los datos relativos a las primeras 12 frecuencias naturales obtenidas como resultado del proceso en *FEAVox*.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Modo	ω_n (Hz)
1	0
2	0
3	0
4	0
5	0
6	0
7	$7,26 \cdot 10^4$
8	$7,28 \cdot 10^4$
9	$1,26 \cdot 10^5$
10	$1,28 \cdot 10^5$
11	$1,3 \cdot 10^5$
12	$1,36 \cdot 10^5$

Tabla 4.- Frecuencias naturales asociadas al conjunto hueso-prótesis

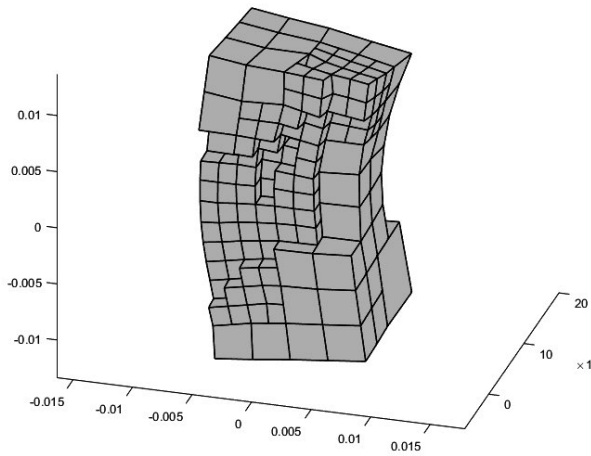
Es necesario recordar que el método con el que se resuelve el problema principal (objeto de este *TFM*) se basa en determinar el grado de fijación del implante al tejido biológico del paciente observando cómo cambia el valor de las frecuencias naturales asociadas a ciertos modos de vibración.

En este caso, los modos de vibración del conjunto que se han analizado se corresponden con los mismos que se han mostrado para el caso de la prótesis por separado. Cabe destacar que el hecho de que ahora se considere el hueso que rodea la prótesis introduce en el problema nuevos modos de vibración y frecuencias naturales, por lo que es necesario dibujar uno a uno los modos y determinar si son o no los mismos que en el caso anterior.

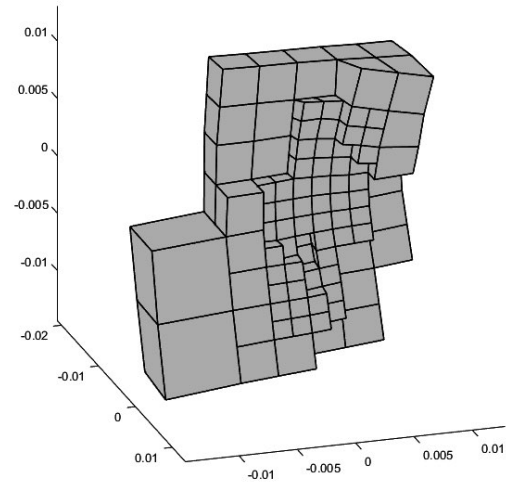
MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Como resultado de ese proceso, se han extraído los siguientes modos, en orden de correspondencia con los mostrados para el caso en que se analizó el implante por separado:

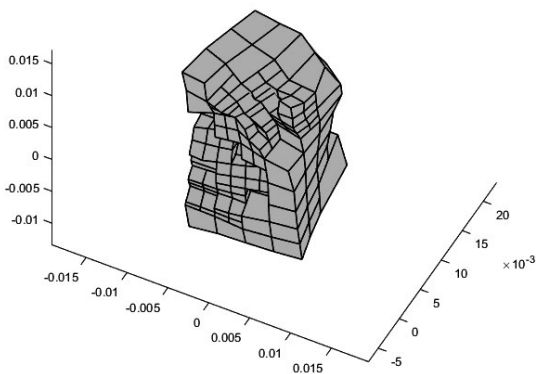
Modo 7: $7,26 \cdot 10^4$ Hz (Flexión)



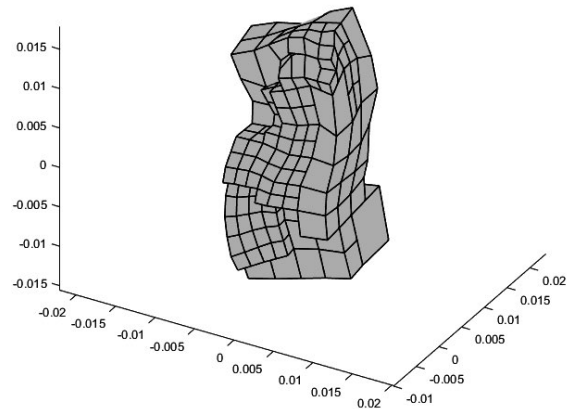
Modo 8: $7,28 \cdot 10^4$ Hz (Flexión)



Modo 9: $1,26 \cdot 10^5$ Hz (Torsión)

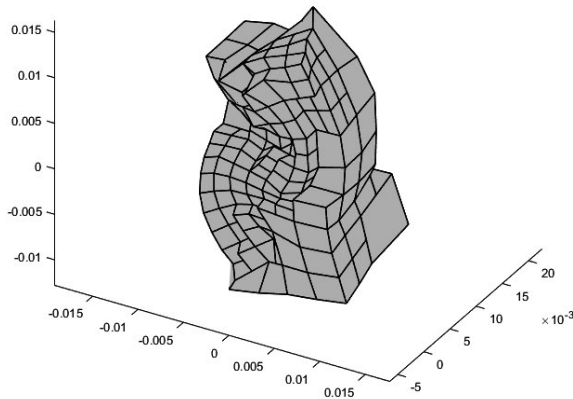


Modo 10: $1,28 \cdot 10^5$ Hz (Flexión)



MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Modo 11: $1,3 \cdot 10^5$ Hz (Flexión)



Las deformaciones asociadas a estos modos están representadas como una sección sobre la malla total realizada sobre el modelo. Esto se realiza así como método de apreciación de la deformada del implante, que es la parte más refinada que se ubica en el interior de la malla.

8.1.5. Comparación de Resultados.

Una vez se tienen todos los modos identificados y se han anotado todas las frecuencias naturales asociadas a estos modos, es preciso comparar los resultados obtenidos para evaluar el grado de fijación del objeto. La *tabla 5* muestra un resumen de los resultados obtenidos en ambos casos para los mismos modos.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Implante (Hz)	Implante-Hueso (Hz)	Variación
$4,120 \cdot 10^4$	$7,26 \cdot 10^4$	$3,14 \cdot 10^4$
$4,273 \cdot 10^4$	$7,28 \cdot 10^4$	$2,987 \cdot 10^4$
$7,876 \cdot 10^4$	$1,26 \cdot 10^5$	$4,724 \cdot 10^4$
$8,683 \cdot 10^4$	$1,28 \cdot 10^5$	$4,117 \cdot 10^4$
$9,234 \cdot 10^4$	$1,30 \cdot 10^5$	$3,766 \cdot 10^4$
	Variación Media (Hz)	$3,7468 \cdot 10^4$

Tabla 5.- Comparación del cambio en las frecuencias naturales para el mismo modo de vibración en los dos casos.

Se aprecia, como tónica general, un aumento notable de las frecuencias asociadas a estos modos en el caso del conjunto Implante-Hueso. Esto sería indicativo de que el hueso aporta una rigidez adecuada al implante, lo que representa un pronóstico favorable de éxito de la intervención. [12]

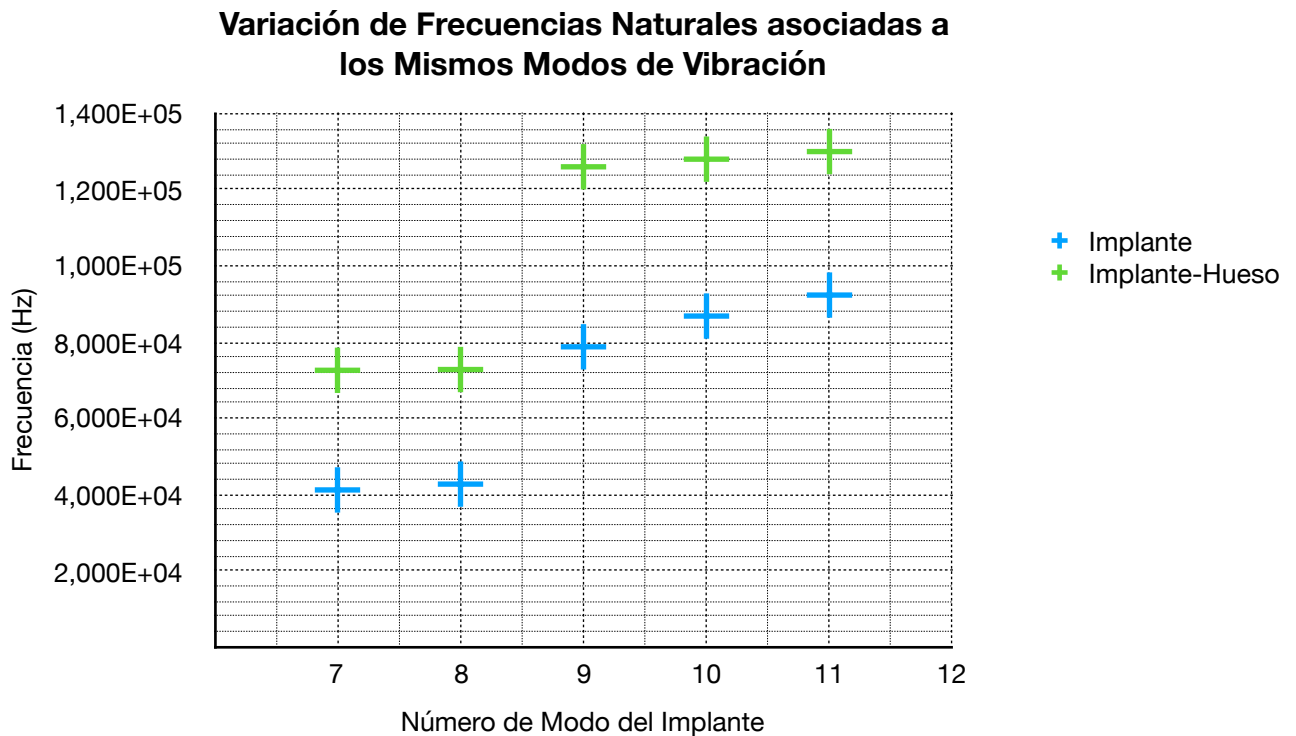


Figura 60.- Representación del cambio en las frecuencias naturales asociadas a los modos de vibración seleccionados para el caso del implante solo.

En la *figura 60* se muestra una representación simplificada del cambio en los valores de las frecuencias naturales asociadas a los mismos modos de vibración que se habían seleccionado para el implante. Un análisis estadístico refleja que se produce un aumento de estos valores en un 58,9%, en términos promediados..

8.1.6. Análisis de Sensibilidades.

Dentro del estudio de la variación de las frecuencias naturales del sistema asociadas a las mismas deformadas, cobra especial interés saber cuánto pueden variar esos valores en función de la rigidez asociada al tejido que rodea al implante. En este sentido, se han realizado varios análisis en los que el módulo de *Young* asociado a la parte biológica se multiplicaba por un factor *k* que reducía o aumentaba sus valores característicos. Con esto

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

se consigue tener una primera idea de la sensibilidad al cambio de las frecuencias naturales.

Modo/k	0,5 (Hz)	0,75 (Hz)	1 (Hz)	1,25 (Hz)
7	$6,034 \cdot 10^4$	$6,697 \cdot 10^4$	$7,257 \cdot 10^4$	$7,748 \cdot 10^4$
8	$6,046 \cdot 10^4$	$6,715 \cdot 10^4$	$7,284 \cdot 10^4$	$7,786 \cdot 10^4$
9	$1,113 \cdot 10^5$	$1,206 \cdot 10^5$	$1,263 \cdot 10^5$	$1,289 \cdot 10^5$
10	$1,144 \cdot 10^5$	$1,228 \cdot 10^5$	$1,281 \cdot 10^5$	$1,343 \cdot 10^5$
11	$1,161 \cdot 10^5$	$1,24 \cdot 10^5$	$1,302 \cdot 10^5$	$1,367 \cdot 10^5$

Tabla 6.-Valores obtenidos para las frecuencias naturales asociadas a los mismos modos de vibración que se estudian en este ejemplo. Cada columna representa los resultados obtenidos empleando el factor k mostrado en la primera celda.

Hay que resaltar que los valores de las frecuencias naturales que se muestran en la *tabla 6* son los correspondientes a los mismos modos de vibración, que se han numerado de acuerdo a los que se mostraron anteriormente para el caso del implante por separado.

Evolución de las Frecuencias Naturales en Función de la Rigidez del Tejido Biológico

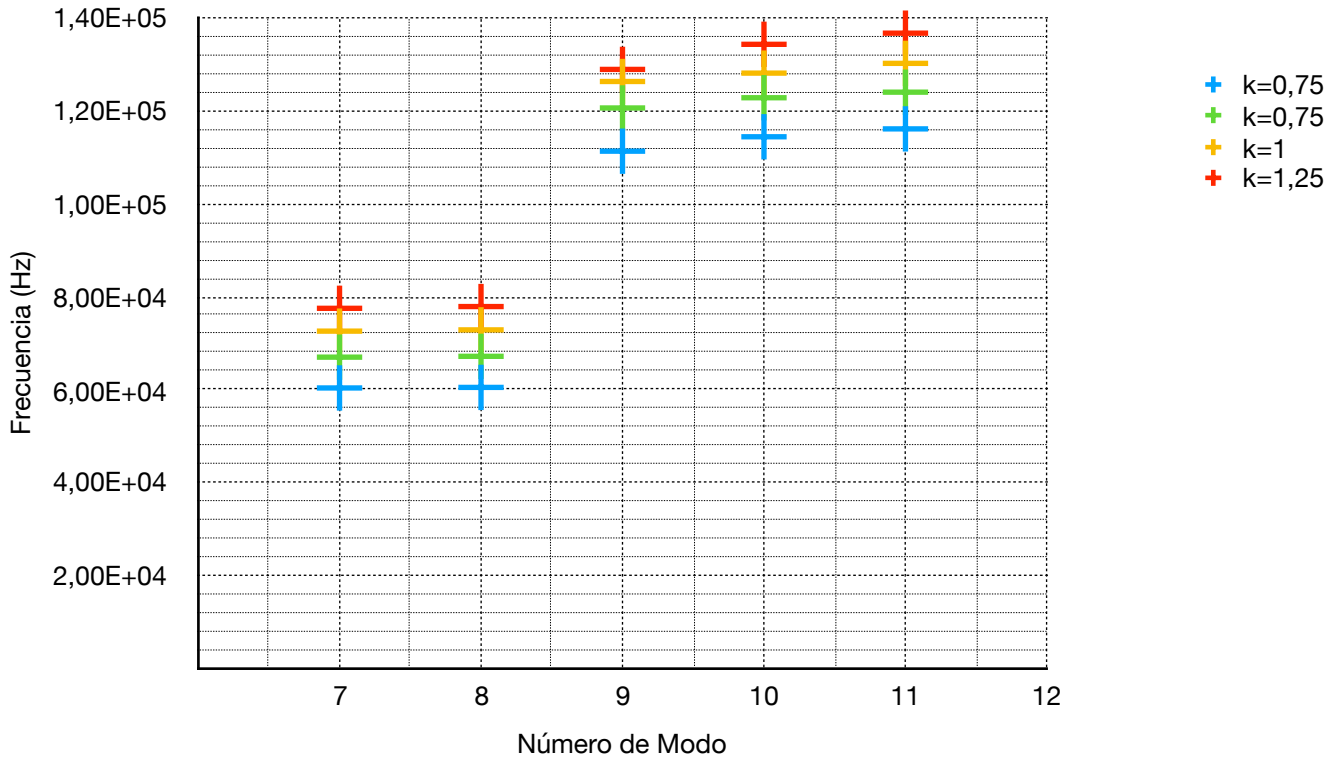


Figura 61.- Cambio en las frecuencias naturales en función de la rigidez del tejido biológico.

En la *figura 61* se puede apreciar que a mayor factor k mayores valores de frecuencia, es decir, a mayor rigidez del hueso, mayor será la fijación. En término medio se ha estimado que un cambio de un 25% en la rigidez del tejido biológico implica una variación del 6,73% en los valores de las frecuencias naturales asociadas a estos modos.

8.2. Implante de Cadera.

De manera análoga al procedimiento realizado en el *apartado 8.1* para el caso de un implante dental, en esta ocasión, se presenta de manera simplificada un análisis modal de una prótesis de cadera de titanio ubicada en la parte superior de un fémur tipo



Figura 62.- Ubicación del implante de cadera sobre la imagen médica.

En ese caso, la imagen de partida tiene dimensiones de 48x73x249 vóxels con lados desiguales, siendo las dimensiones de éstos 0.82x0.82x0.625 mm.

8.2.1. Propiedades del Material asociadas a la Imagen.

En este caso, el análisis de la bibliografía disponible [14] sobre las propiedades mecánicas del fémur humano refleja que propiedades tales como el módulo de *Young* deben estimarse a partir de fórmulas experimentales que se basan en la densidad de las cenizas que se tienen cuando se quema el hueso. Por este motivo, las propiedades que en este caso se aplican son vectores con puntos de referencia (en el sistema internacional) como éstos:

$$HU=[1,1150,1275,1400,1525,1650,1775,1900,2858,4001];$$

$$Nu=[0.32,0.32,0.32,0.32,0.32,0.32,0.32,0.32,0.32,0.32];$$

$$E=1e6*[14,1204.77,1390.17,1583.59,1784.64,1992.98,2208.32,2430.38,2430.38,96000];$$

$$\rho=[0,0,0,0,0,0,0,0,0,4620];$$

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

De nuevo la masa asociada al mapa de bits del tejido biológico se iguala a cero, pues se pretende simular un implante rodeado por un tejido que le aporta rigidez sin añadir masa, con el fin de simplificar el análisis. El material de la prótesis es titanio, y se identifica con las últimas posiciones de estos vectores.

8.2.2. Discretización y Ubicación del Implante.

El proceso de mallado y obtención del archivo de malla en *GiD* es idéntico al realizado para el ejemplo del implante dental. Simplemente debe tenerse en cuenta que la imagen, y por tanto la malla, no son isométricas, en este caso más alargadas en dirección Z.



Figura 63.- Implante mallado (izquierda) y colocado sobre el fémur (derecha)

8.2.3. Análisis Modal del Implante.

Para valorar el grado de fijación del implante al tejido óseo del paciente, se deben estudiar las frecuencias y modos de vibración. En primer lugar, se obtienen las correspondientes al implante por separado. En la tabla siguiente se exponen los resultados.

Modo	ω_n (Hz) FEAVox	ω_n (Hz) ANSYS
1	0	0
2	0	0
3	0	0
4	0	$9,08 \cdot 10^{-3}$
5	0	$2,37 \cdot 10^{-2}$
6	0	$3,88 \cdot 10^2$
7	$2,668 \cdot 10^3$	$2,437 \cdot 10^3$
8	$2,747 \cdot 10^3$	$2,708 \cdot 10^3$
9	$5,064 \cdot 10^3$	$5,272 \cdot 10^3$
10	$5,903 \cdot 10^3$	$5,5 \cdot 10^3$
11	$6,28 \cdot 10^3$	$7,297 \cdot 10^3$
12	$8,281 \cdot 10^3$	$8,423 \cdot 10^3$

Tabla 7.- Frecuencias naturales del implante

En este caso, los resultados obtenidos con *FEAVox* y *ANSYS* concuerdan aproximadamente tanto en orden de magnitud como numéricamente. En cuanto a la lógica de los resultados, simplemente cabe destacar que las seis primeras frecuencias naturales son nulas, lo que encaja con el tipo de análisis en vibraciones libres realizado. De nuevo, las

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

escasas diferencias entre los resultados de *ANSYS* y *FEAVox* se deben a la diferencia entre sus mallas.

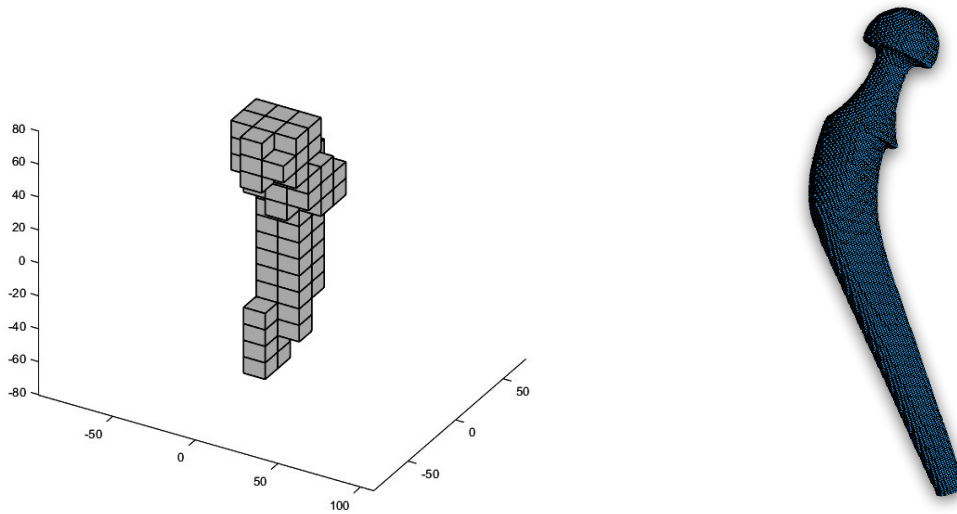
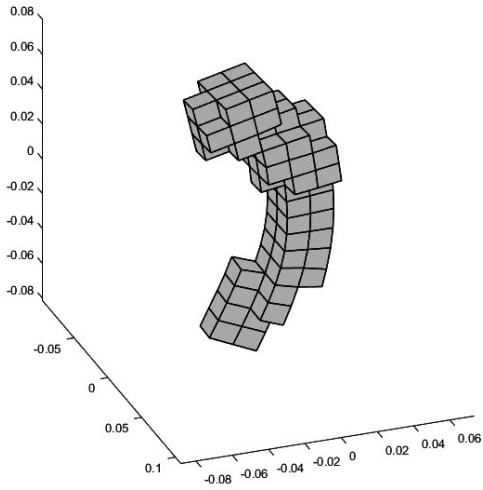


Figura 64.- Malla del implante en *FEAVox* (izquierda) y en *ANSYS* (derecha)

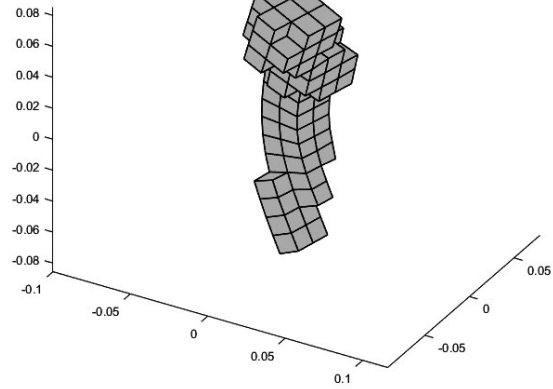
En la *tabla 7* se muestran en **negrita** las frecuencias naturales asociadas a los modos que se estudiarán en el conjunto hueso-implante posterior. Se representan a continuación los modos de vibración:

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

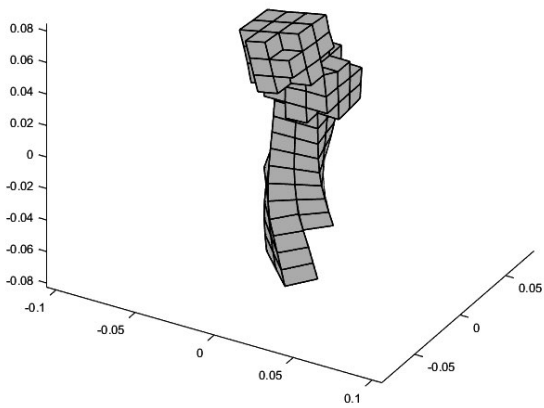
Modo 7: $2,66 \cdot 10^3$ Hz (Flexión)



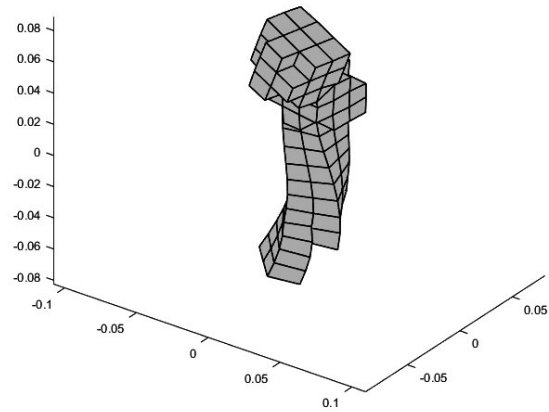
Modo 8: $2,74 \cdot 10^3$ Hz (Flexión)



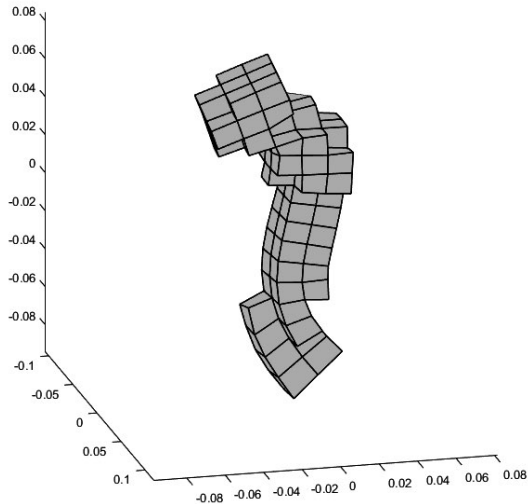
Modo 9: $5,06 \cdot 10^3$ Hz (Flexo-Torsión)



Modo 10: $5,9 \cdot 10^3$ Hz (Flexo-Torsión)



Modo 11: $6,3 \cdot 10^3$ Hz (Flexión)



8.2.4. Análisis de la Unión Implante-Hueso.

De nuevo, la valoración del cambio en las frecuencias naturales del conjunto hueso-implante va a caracterizar el nivel ideal de fijación existente entre ambos. En este sentido, y tras el análisis modal realizado, se muestran los resultados:

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

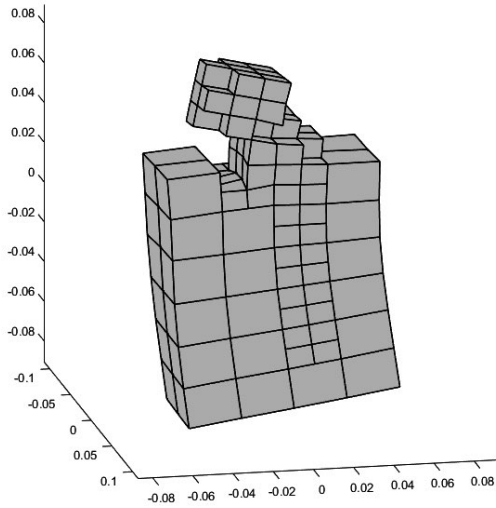
Modo	ω_n (Hz)
1	0
2	0
3	0
4	0
5	0
6	$4,229 \cdot 10^{-4}$
7	$4,343 \cdot 10^3$
8	$4,51 \cdot 10^3$
9	$6,777 \cdot 10^3$
10	$9,314 \cdot 10^3$
11	$9,387 \cdot 10^3$
12	$1,217 \cdot 10^4$
13	$1,447 \cdot 10^4$
14	$1,511 \cdot 10^4$
15	$1,873E \cdot 10^4$

Tabla 8.- Frecuencias naturales de la unión implante-hueso

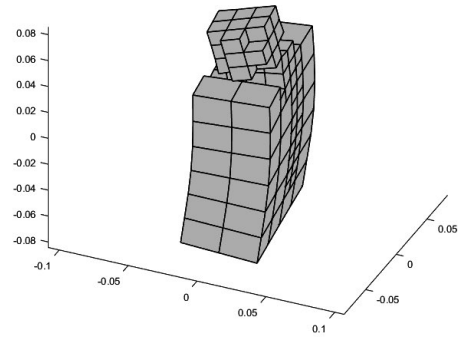
Los modos de vibración del implante asociados a las frecuencias destacadas en la *tabla 8* son los identificados anteriormente en el análisis de la prótesis por separado, solo que aparecen más modos y los que se están estudiando en este ejemplo se separan más. A continuación se muestran las deformadas asociadas a estas frecuencias, en orden de correspondencia con los resultados del anterior análisis.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

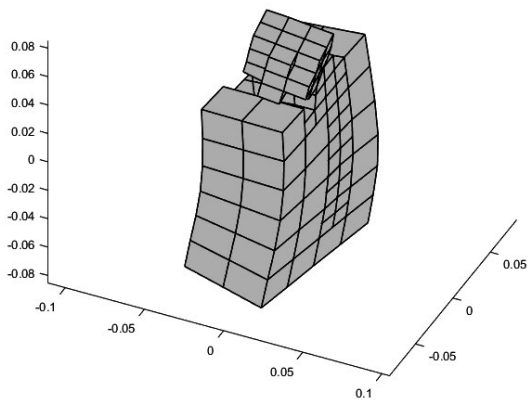
Modo 7: $4,34 \cdot 10^3$ Hz (Flexión)



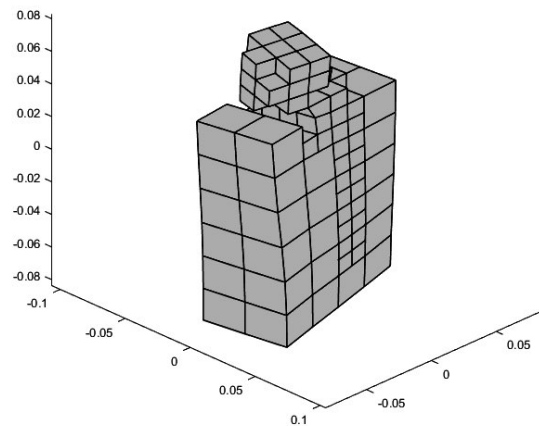
Modo 8: $4,5 \cdot 10^3$ Hz (Flexión)



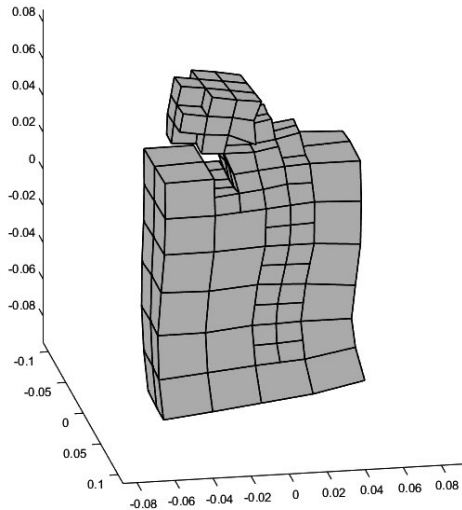
Modo 9: $6,77 \cdot 10^3$ Hz (Flexo-Torsión)



Modo 11: $9,78 \cdot 10^3$ Hz (Flexo-Torsión)



Modo 15: $1,87 \cdot 10^4$ Hz (Flexión)



8.2.5. Comparación de Resultados.

Para facilitar las tareas de comprobación de los resultados, se expone a continuación una tabla con las correspondencias entre las frecuencias naturales asociadas a las mismas deformaciones.

Implante (Hz)	Implante-Hueso (Hz)	Variación
$2,668 \cdot 10^3$	$4,343 \cdot 10^3$	$1,675 \cdot 10^3$
$2,747 \cdot 10^3$	$4,510 \cdot 10^3$	$1,763 \cdot 10^3$
$5,064 \cdot 10^3$	$6,777 \cdot 10^3$	$1,713 \cdot 10^3$
$5,903 \cdot 10^3$	$9,387 \cdot 10^3$	$3,484 \cdot 10^3$
$6,280 \cdot 10^3$	$1,87 \cdot 10^4$	$1,242 \cdot 10^4$
	Variación Media (Hz)	$4,211 \cdot 10^3$

Tabla 9.- Comparación de la variación de las frecuencias naturales asociadas a los mismos modos de deformación

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

En términos generales las frecuencias aumentan para el mismo modo de vibración. La elevada variación media del valor de estas frecuencias es un gran indicativo del grado de fijación existente entre la prótesis de titanio y el tejido biológico.

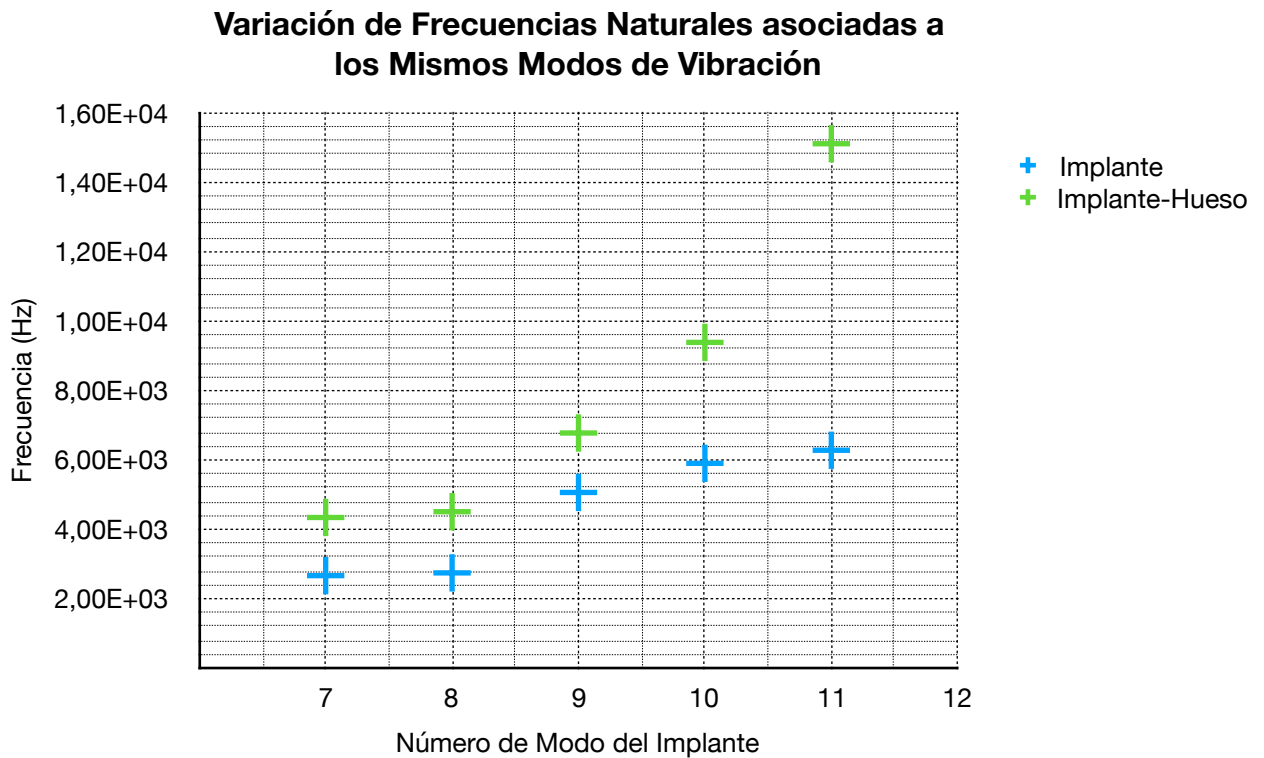


Figura 65.- Representación del cambio en las frecuencias naturales asociadas a los modos de vibración seleccionados para el caso del implante solo.

En la *figura 65* puede verse una representación del aumento de las frecuencias naturales asociadas a los mismos modos de vibración. Para este ejemplo, el conjunto implante-hueso ofrece valores medios de frecuencias naturales un 72% más altas que en el caso del implante por separado.

8.2.6. Análisis de Sensibilidades.

De manera análoga al ejemplo analizado en el apartado 8.1 se muestran los resultados obtenidos al multiplicar la rigidez del tejido biológico por un factor k que reduce o amplifica el módulo de *Young* característico. El objetivo es conocer la variación de las frecuencias naturales asociadas a los mismos modos en función de la rigidez del conjunto biológico.

Modo/k	0,5 (Hz)	0,75 (Hz)	1 (Hz)	1,25 (Hz)
7	3,880E+03	4,144E+03	4,343E+03	4,500E+03
8	3,996E+03	4,293E+03	4,510E+03	4,675E+03
9	6,191E+03	6,495E+03	6,777E+03	7,043E+03
10	8,579E+03	9,065E+03	9,387E+03	9,766E+03
11	1,770E+04	1,826E+04	1,873E+04	1,915E+04

Tabla 10.-Valores obtenidos para las frecuencias naturales asociadas a los mismos modos de vibración que se estudian en este ejemplo. Cada columna representa los resultados obtenidos empleando el factor k mostrado en la primera celda.

En la figura 66 se aprecia de nuevo que, a mayor valor del factor k , mayores son los valores de las frecuencias naturales de estudio. Además, un estudio estadístico de estos resultados refleja que, de media, un cambio de un 25% en la rigidez del hueso implica una variación del 4,38% en los valores de las frecuencias naturales asociadas a las mismas deformadas.

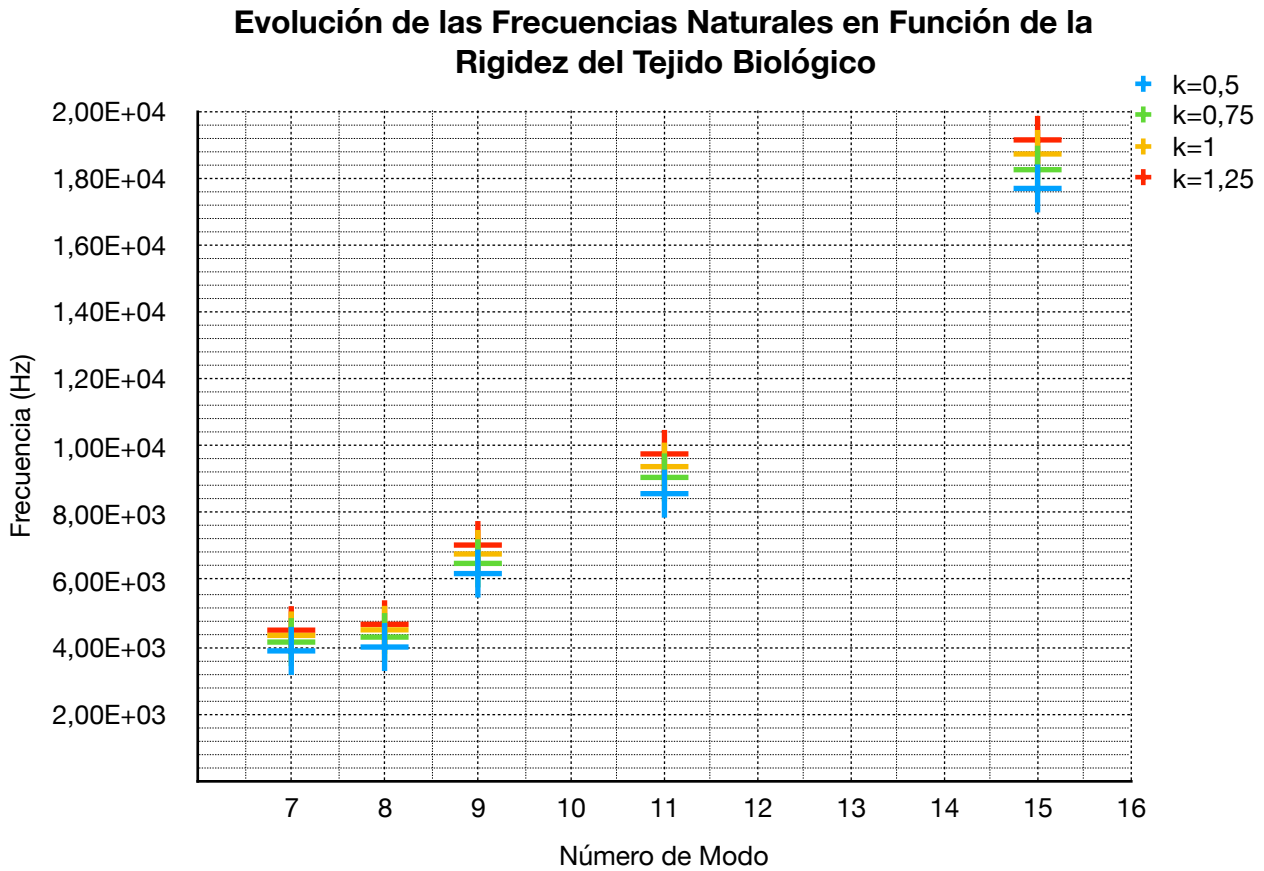


Figura 66.- Cambio en las frecuencias naturales en función de la rigidez del tejido biológico.

9. Conclusiones.

Se ha presentado una metodología que permite estimar la variación de las frecuencias propias de un implante debido a la sujeción por parte del hueso, simulando un caso ideal de perfecta osteointegración del implante con el hueso tal y como está caracterizado en los TACs preoperatorios. Esta configuración representa un caso ideal que, aún no dándose en la realidad, puede ayudar a desarrollar una técnica de evaluación de la sujeción del implante basada en *GiD*, *Implant Viewer* y *FEAVox*.

La incorporación al código interno de *FEAVox* del análisis modal ha sido plenamente satisfactoria, pues se han obtenido resultados lógicos cuya estructura refleja lo que cabría

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

esperar en un análisis habitual en vibraciones libres (primeras seis frecuencias naturales nulas, etc.).

Al margen de todo esto y de las dificultades asociadas a la realización de este *TFM*, es posible establecer algunas conclusiones principales en base a los resultados y a la metodología seguida:

- *Implant Viewer* permite trabajar en la ubicación y orientación del implante con gran precisión y rapidez, siendo también casi totalmente independiente en relación a otros programas. Su interfaz de usuario facilita mucho las operaciones asociadas a este tipo de tareas.
- *FEAVox* constituye un código de cálculo por *Elementos Finitos* eficiente, y ofrece posibilidades en el ámbito del análisis de imágenes por este método.
- Los resultados obtenidos en este trabajo demuestran que existe relación entre las frecuencias naturales y la calidad del hueso que rodea al implante. Por lo tanto, la variación de las frecuencias naturales del implante, al considerar el hueso que lo rodeará, puede representar un biomarcador del grado de sujeción del implante y, por consiguiente, del éxito de la operación.

10. Líneas de Mejora.

Pese a que el objetivo principal que persigue este trabajo fin de master ha sido conseguido, es posible establecer algunas pautas de mejora que sirvan como guía a la hora de continuar con los desarrollos de los conceptos que aquí se han planteado. En este sentido, resulta útil tener en cuenta algunas de estas propuestas:

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

- Determinar con precisión el criterio mediante el cual *GiD* determina cuándo se coloca un elemento y cuándo no en la malla cartesiana, con el fin de mejorar la eficiencia del proceso.
- Incluir en *Implant Viewer* un módulo que le permita mallar las geometrías de los implantes, de tal manera que no dependa de programas externos de preprocesado.
- Implementar mejoras en el ámbito de la velocidad de cálculo, mediante paralelización con *GPU* o con nuevas técnicas de optimización de código.
- Mejorar el rendimiento global de *Implant Viewer* y *FEAVox* cuando éste debe modificar datos de gran tamaño, esto es, cuando las imágenes médicas tienen grandes dimensiones.
- Incluir en *FEAVox* el cálculo de las matrices de masa para otros tipos de problemas, no solo en los referentes al análisis de imágenes médicas.
- Automatizar el proceso de obtención de los modos de vibración en *FEAVox*, pues hasta ahora su dibujo requiere de intervención por parte del usuario.
- Cuantificar estadísticamente el grado de fijación habitual entre el conjunto hueso-implante para los casos habituales de operaciones quirúrgicas, con el objetivo de obtener cuantitativamente una estimación de los resultados que se obtendrán en función de las características del tejido del paciente.

11. Bibliografía.

[1] Willams, D.F., *Definiciones de Biomateriales. Avances en Ingeniería Biomédica*, Amsterdam: European Society for Biomaterials, 1987.

[2] Smith, W. F. y Hashemi, J. *Fundamentos de la Ciencia e Ingeniería de Materiales*, Mexico: McGRAW-HILL, 2006, pp. 448-454.

[3] DICOM - *Digital Imaging and Communications in Medicine*
<http://dicom.nema.org>

[4] Zienkiewicz, O. C. *El Método de los Elementos Finitos Vol. 1*. Reverte: 1981

[5] Besa González, A.J. y Carballeira Morado, J. *Diagnóstico y Corrección de Fallos de Componentes Mecánicos*. Universidad Politécnica de Valencia, 1ª edición: 2015.

[6] Bushong, S. C. *Manual de Radiología para Técnicos. Física, Biología y Protección Radiológica*. Elsevier España, 9ª edición: 2010.

[7] CIMNE *GiD Reference Manual*
<http://www.gidhome.com/documents/referencemanual>, 2016

[8] The Mathworks Inc. *MATLAB Documentation Center*.
<http://www.mathworks.com/help/matlab/>, 1994-2014.

[9] Navarro Jimenez, J. M. y Ródenas García, J.J. *Optimization of a finite element code implemented in MATLAB. On the use of GPUs for High Performance Computing. Master Thesis. ETSII-UPV. 2014*

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

- [10] Lorensen, W. E, Cline and H. E. *Marching Cubes: A high resolution 3D surface construction algorithm*. In: Computer Graphics, Vol. 21, Nr. 4 1987
- [11] Millán Rodríguez, I y Ródenas García, J.J. *Interface for the integration of a 3D Cartesian finite element code and a commercial post processing software. Master Thesis, ETSII-UPV. 2015.*
- [12] Wille, H., Rank, E. y Yosibash, Z. *Prediction of the mechanical response of the femur with uncertain elastic properties*. Elsevier: 2012.
- [13] Hellmich, C., Kober, C., y Erdmann, B. *Micromechanics-Based Conversion of CT Data into Anisotropic Elasticity Tensors, Applied to FE Simulations of a Mandible: 2008.*
- [14] Yosibash, Z., Travelsi, N. y Milgrom, C. *Reliable simulations of the human proximal femur by high-order finite element analysis validated by experimental observations*. Elsevier: 2007.



Implant Viewer

ANEXO I. Manual de Usuario

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

ÍNDICE

1. ¿Qué es Implant Viewer?.....	5
2. Requisitos Mínimos.....	5
2.1.Windows.....	6
2.2.MacOS.....	6
2.3.Linux.....	7
3. Sistema de Archivos e Instalación.....	7
3.1.Sistema de Archivos.....	8
3.2.Instalación.....	10
4. Interfaz Gráfica.....	10
4.1.Ventana de Control.....	11
4.2.Ventanas de Vistas.....	13
5. Nociones sobre Implant Viewer.....	15
5.1.Archivos en Implant Viewer.....	15
5.2.Funciones Principales.....	16
5.3.Controles de Dibujo.....	17
5.4.Movimiento del Implante.....	19
5.4.1.Traslación del Implante.....	20
5.4.2.Rotación del Implante.....	21
5.5.Otras Funciones.....	23
5.5.1Corte de la Imagen Médica.....	23
5.5.2.Reparación del Implante.....	24
5.5.3.Renderizado.....	26
5.5.4.Ángulos de Euler (ZYZ).....	27
5.6.Menú Principal.....	28
5.6.1.Pestaña File.....	28
5.6.2.Pestaña Edit.....	29
5.6.3.Pestaña DICOM.....	30
5.7.Panel de Preferencias.....	31
5.7.1.Plot Options.....	31
5.7.2.Others.....	32
6. Atajos de Teclado.....	34

1. ¿Qué es Implant Viewer?

En el entorno médico, la necesidad y viabilidad de operar a un paciente para colocarle un implante de cualquier tipo puede requerir del conocimiento específico de las circunstancias propias de la zona corporal donde debe ubicarse dicho objeto. En la práctica habitual, esto se realiza empíricamente, pero existen otras alternativas útiles a valorar que no requieren de un seguimiento físico del paciente. *Implant Viewer* constituye una de esas alternativas.



Este software, basado en el lenguaje *Matlab*, permite al usuario interactuar con el modelo tridimensional de una imagen médica para colocar sobre ella un implante, definido como un modelo de vóxels creado a partir de un modelo *CAD*, en la ubicación necesaria, con el objetivo de poder analizar principalmente la viabilidad del mismo en base a la calidad ósea de la zona circundante. Para conseguir esto, el usuario dispone de una interfaz sencilla que permite una colocación rápida e intuitiva del objeto, con múltiples controles y posibilidades de modificación y control sobre lo que se está haciendo.

La posibilidad de integración efectiva de *Implant Viewer* con otras herramientas de análisis de imagen por elementos finitos, tales como *FEAVox*, hacen de este programa una herramienta extremadamente útil en campos de la salud.

2. Requisitos Mínimos.

Los requisitos mínimos de hardware necesarios para ejecutar el programa se identifican con necesarios para ejecutar *Matlab*. Se detallan a continuación en función del sistema operativo.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

2.1. Windows.

Para sistemas de 32 bit los requerimientos de hardware son:

- **Procesador:** Intel Pentium (Pentium IV y posterior), Intel celeron, Intel Xeon, Intel Core, AMD Athlon, AMD Optaron.
- **Disco duro:** se requieren al menos 500MB de espacio disponible.
- **Memoria RAM:** 512 MB.
- **Video:** Tarjeta gráfica de 16, 24 o 32 bits con OpenGL, DirectX 9.

Para sistemas de 64 bit:

- **Procesador:** Intel Pentium (Pentium IV y posterior), Intel celeron, Intel Xeon, Intel Core, AMD64.
- **Disco duro:** se requieren al menos 500 MB de espacio disponible.
- **Memoria RAM:** 512 MB.
- **Video:** Tarjeta gráfica de 16, 24 o 32 bits con OpenGL, DirectX 9.

2.2. MacOS.

Para sistemas MAC (Mac OS X 10.4.7, Mac OS X 10.4.8) los requerimientos de hardware se detallan a continuación:

- **Procesador:** PowerPC (G4), Intel
- **Disco duro:** se requieren al menos 500MB de espacio disponible.
- **Memoria RAM:** 512 MB.
- **Video:** 11X Server.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

2.3. Linux.

Para sistemas de 32bit los requerimientos de hardware son:

- **Procesador:** Intel Pentium (Pentium IV y superiores), Intel Celeron, Intel Xeon, Intel Core, AMD Athlon, AMD Opteron.
- **Disco duro:** se requieren al menos 500MB de espacio disponible.
- **Memoria RAM:** 512 MB.
- **Video:** Tarjeta gráfica de 16, 24 o 32 bits con OpenGL, DirectX 9.

Para sistemas 64bit:

- **Procesador:** Intel Pentium (Pentium IV y posteriores), Intel Celeron , Intel Xeon, Intel Core, AMD64.
- **Disco duro:** se requieren al menos 500MB de espacio disponible.
- **Memoria RAM:** 512 MB.
- **Video:** Tarjeta gráfica de 16, 24 o 32 bits con OpenGL, DirectX 9.

3. Sistema de Archivos e Instalación.

Implant Viewer está constituido esencialmente por varios ficheros y carpetas que determinan el correcto funcionamiento del software. Entender su estructura interna y realizar una instalación efectiva asegurará la ejecución del programa.

3.1. Sistema de Archivos.

Implant Viewer ha sido programado íntegramente de manera modular, es decir, el conjunto del programa está dividido en pequeños scripts de código independientes entre sí. Esos scripts, pese a realizar funciones muy diversas, trabajan coordinadamente para posibilitar la ejecución correcta del software. El conjunto de todos estos archivos, estructurado bajo un sistema ordenado y lógico, debe ubicarse enteramente en un directorio común, pues constituye la columna vertebral del programa.

En la figura 1 se muestra un diagrama básico de la estructura de carpetas en las que están alojadas las diferentes funciones de *Implant Viewer*. La ubicación de los archivos obedece a un orden, determinado por la función específica de cada uno de ellos dentro del desempeño general del programa. De este modo, se encuentran inicialmente, además de los archivos principales de arranque del programa, tres carpetas principales: *Movement* (movimiento), *Plot Controls* (controles de dibujo) y *Tools* (herramientas).

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

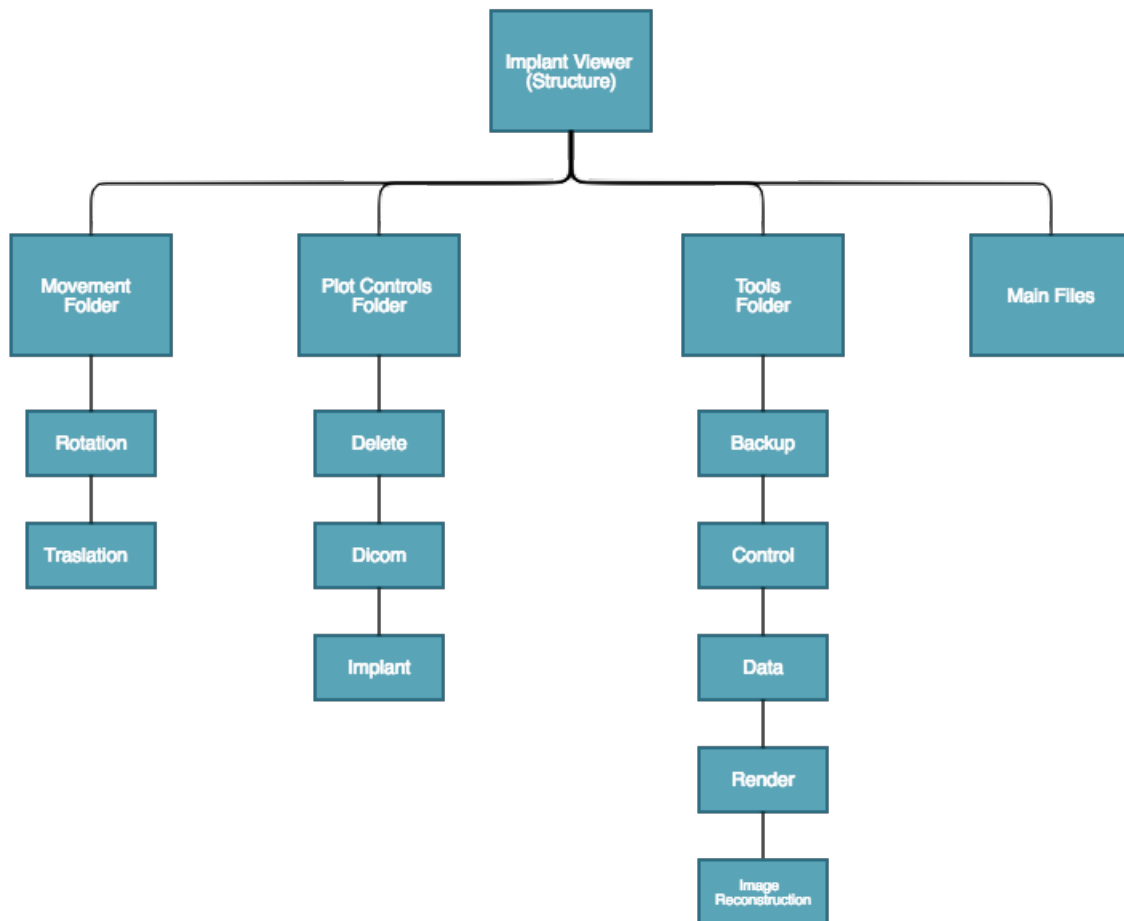


Figura 1.- Estructura de carpetas de *Implant Viewer*

- **Movement**, como su propio nombre indica, contiene todos aquellos scripts relacionados con el movimiento del implante sobre la imagen médica. Se divide en los movimientos relacionados con la traslación y rotación del implante.
- **Plot Controls** hace referencia a las funciones que controlan el dibujo de la información con la que se está trabajando. Así, dentro de esta carpeta pueden encontrarse rutinas de dibujo de la imagen médica (en formato DICOM), del modelo del implante o de borrado de lo dibujado previamente.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

- **Tools** contiene todos aquellos algoritmos que controlan, extraen y/o modifican información de la imagen médica, del implante o del programa. Se encuentran aquí scripts relacionados con copias de seguridad, control del funcionamiento general, generación del renderizado 3D, reparación de píxeles y extracción de datos a partir de la información de entrada (carpeta *Data*).

Además de todas estas carpetas, en la estructura principal se encuentran algunos archivos esenciales para ejecutar el programa, como son *Preferences.m* o *ImplantViewer.m*, siendo este último el script de arranque.

3.2. Instalación.

La instalación del software es relativamente sencilla, aunque hay diferentes procedimientos en función de las necesidades del usuario. Si se dispone de una instalación previa de *Matlab* en el ordenador de destino, simplemente debe copiar la estructura de archivos del programa en un mismo directorio y ejecutar el script llamado *Implant Viewer*.

En caso contrario, es posible instalar el programa de la manera habitual con un asistente de instalación o emplear una copia portable de la aplicación.

Nota: Es posible que se requiera alguna Toolbox adicional de Matlab para la ejecución del programa.

4. Interfaz Gráfica.

La interfaz gráfica de *Implant Viewer* es fundamentalmente modular, ya que se compone de 5 ventanas: una principal (ventana de control) y otras cuatro que muestran las diferentes vistas de la imagen médica (coronal, sagital, axial y tridimensional). Cada una de

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

esas ventanas es independiente de las otras y puede colocarse en cualquier parte de la pantalla.

A continuación se detallan las características fundamentales de los dos tipos de ventanas existentes.

4.1. Ventana de Control.

La ventana de control, o ventana principal, constituye la base de funcionamiento del software, pues en ella se encuentran todas las funciones que permiten al usuario interactuar con el programa a diferentes niveles. En términos generales, esta ventana es la que determina lo que se muestra en las otras por medio de las funciones y controles que incorpora.

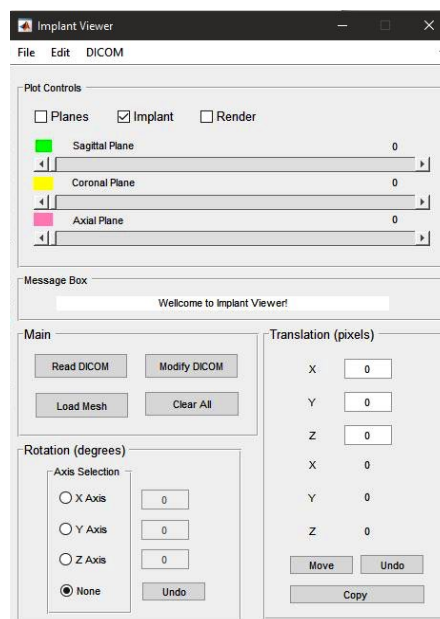


Figura 2.- Ventana de Control

Como puede apreciarse en la figura 2 la ventana de control se divide en secciones que agrupan funciones que sirven para realizar una determinada tarea dentro del programa. De este modo, se tiene: el panel *Main* (Principal), *Plot Controls* (Controles de Dibujo),

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

Traslacion (Traslación), *Rotation* (Rotación) y la *Message Box*. Cabe destacar, además de esto, que la ventana dispone de un menú superior en el que se pueden encontrar algunas de las funciones mostradas a través de los botones de la interfaz, junto con otras opciones útiles para el uso del software.

- **Main:** En este panel se agrupan las cuatro funciones más utilizadas en el programa, como son la lectura/escritura de los archivos y parámetros empleados en *Implant Viewer*.
- **Plot Controls:** Aquí se agrupan varios sliders que controlan los cortes bidimensionales que se realizan sobre la imagen médica y el implante. Además de eso, existen tres opciones que permiten activar y desactivar la visualización de los planos de corte, el implante y el modelo tridimensional, respectivamente (de izquierda a derecha).
- **Traslacion:** Este panel permite la introducción por parte del usuario de las distancias necesarias para mover el objeto a lo largo de los tres ejes. Deben introducirse los tres números a la vez, y en el caso de que no sea necesario trasladar en una dirección se pondrá un cero. Por otro lado, *Implant Viewer* mostrará la distancia total recorrida por el implante en términos de cada eje.
- **Rotation:** Al igual que con las traslaciones, el software permite introducir valores en grados que permitan girar el implante. Sólo se permite girar alrededor de un eje a la vez.
- **Message Box:** En ella se muestran mensajes que permiten interactuar con el usuario.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

Una explicación más detallada de las capacidades de *Implant Viewer* y sus funciones se detalla en apartados sucesivos.

4.2. Ventanas de Vistas.

En la interfaz gráfica que compone el programa se pueden apreciar tres ventanas similares a las que se ven en el la figura 3. Se trata de las ventanas de vistas bidimensionales, en las que se muestran las imágenes que resultan del corte de los planos coronal, sagital y axial de la imagen médica.

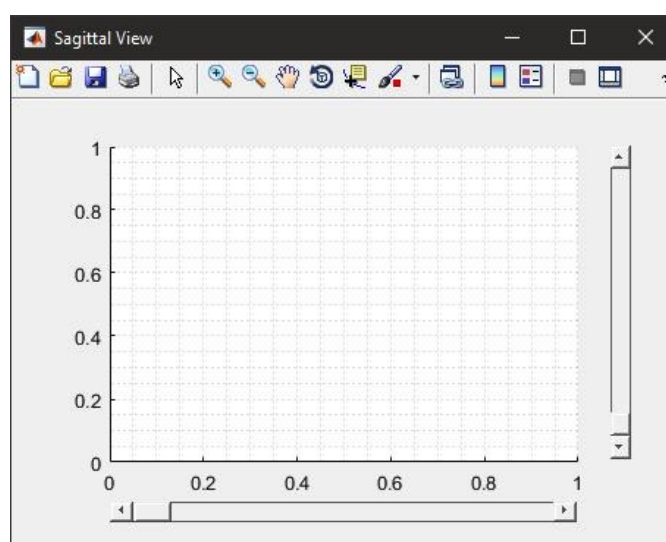


Figura 3.- Una de las ventanas de vistas (plano sagital)

Este tipo de ventanas disponen de dos sliders que permiten mover los otros dos planos que no afectan a la ventana en cuestión. Si éstos se mueven manualmente con el ratón, se puede obtener una previsualización instantánea de la nueva posición de los planos sobre esa misma ventana.

Las tres ventanas de vistas poseen controles simplificados que permiten mejorar la experiencia de usuario. Al tener una de ellas como principal es posible utilizar dichos controles:

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

- **Avanzar en el plano de corte actual:** El plano que se está visualizando en la ventana de vista es modificable moviendo la rueda del ratón. Si se hace en un sentido se avanzará y en el otro se retrocederá en referencia al sistema de coordenadas.
- **Mover el Implante:** Es posible modificar la posición del implante arrastrándolo con el botón izquierdo del botón. Al soltar el botón se modificará la posición del implante.
- **Orientar el Implante:** Se hace click derecho (o *Control + Click Izquierdo*) sobre el punto de giro y se mantiene y mueve al ratón hasta obtener el ángulo deseado.

Además de estas tres ventanas existe otra similar que sirve para ver la representación tridimensional. Por cuestiones propias del rendimiento gráfico mostrará por defecto una caja azul, que representa los límites de la imagen médica, el implante y los tres planos de corte. El objetivo principal de esta representación es orientar al usuario cuando está colocando el objeto. No obstante, si así es deseado, es posible mostrar también un renderizado de la imagen médica en esta ventana (a través de las opciones de la ventana principal).

Nota: El orden de redibujado de la información disponible en las ventanas de vistas cuando se realiza un cambio es siempre el mismo, y se corresponde con los planos coronal, sagital, y axial, y finaliza con la ventana tridimensional.

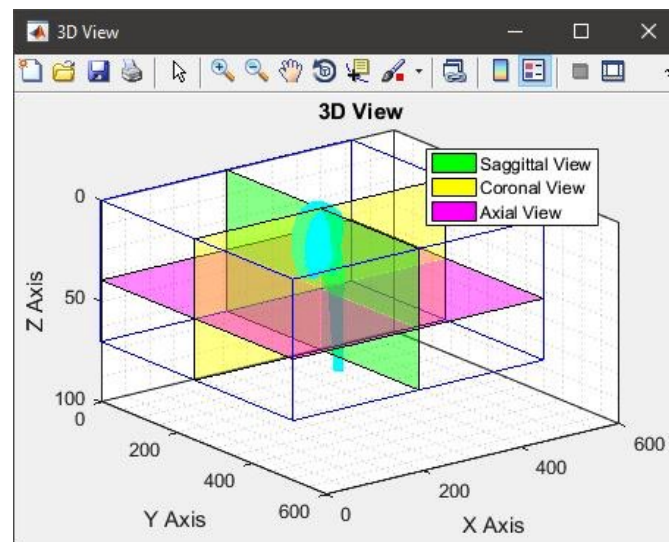


Figura 4.- Ventana de vistas (representación tridimensional)

5. Nociones sobre *Implant Viewer*.

El manejo de este software busca aportar sencillez al proceso de colocación del modelo de implante sobre una imagen médica. En este sentido, *Implant Viewer* dispone de múltiples herramientas pensadas para favorecer la experiencia de usuario, así como algunos atajos de teclado y controles de ratón que agilizan la modificación de la imagen médica.

En los próximos sub-apartados se desarrollan de manera detallada las principales características del programa y los controles asociados a cada una de ellas.

5.1. Archivos en *Implant Viewer*.

Implant Viewer es capaz de trabajar con diferentes tipos de archivos, algunos de los cuales de uso frecuente en el tratamiento de imágenes médicas y sus estándares. La lista de archivos que el programa es capaz de leer se detalla a continuación:

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

- **Archivos DICOM (*.dcm o *.dicom):** Son los archivos que se identifican con la información que aporta la imagen médica. Pueden estar constituidos por un solo archivo o por varios de ellos. El programa es capaz de leerlos (sean uno o varios) y modificarlos para colocar el implante.
- **Archivos MAT (*.mat):** Se trata de los archivos propios de *Matlab*. *Implant Viewer* los emplea para guardar las imágenes médicas modificadas en la ubicación deseada por el usuario. También es posible leer un archivo MAT como imagen médica.
- **Archivos MSH (*.msh):** Son aquellos que se asocian con el modelo tridimensional del implante, pues son los textos ASCII que exporta GiD con la información propia de la malla creada a partir del modelo CAD.
- **Archivos TXT (*.txt):** El software los usa para exportar la información de la orientación del implante con respecto a sus condiciones iniciales.

5.2. Funciones Principales.

Para comenzar con el proceso de colocación en *Implant Viewer* es necesario introducir en el programa la información de la imagen médica y del implante que se pretende ubicar. Además de esto, es especialmente importante poder modificar los datos una vez se ha concluido esta tarea. Se trata de funciones básicas en el funcionamiento del programa, y se ubican dentro del cuadro *Main* situado en la ventana de control.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA



Figura 5.- Panel de control con las funciones principales

En términos específicos, las funciones principales son:

- Lectura de archivos *DICOM* (read DICOM).
- Escritura de la imagen médica (modify DICOM).
- Lectura de los datos de malla o carga del implante (load mesh).
- Borrado de los datos de sesión (clear all).

Además de estos botones, es posible acceder a todas estas funciones a partir del menú principal de la ventana de control (véase apartado 5.5).

Nota: El programa requiere que primero se lea la imagen médica y luego se cargue la malla del implante.

5.3. Controles de Dibujo.

Los controles de dibujo están relacionados con aquellos algoritmos del programa que se encargan de dibujar en las ventanas de vistas. Tienen que ver tanto con la posibilidad de dibujar la información como con el modo en que ésta se visualiza.

Implant Viewer dispone de múltiples controles que permiten la visualización de la información y ofrecen la posibilidad de customizar el aspecto de la misma. Los más importantes se ubican en la sección *Plot Controls* de la ventana de control y en las ventanas de vistas.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

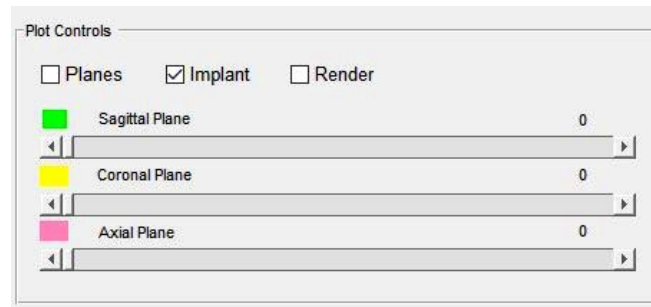


Figura 6.- Panel con los controles sobre los dibujos en las ventanas de vistas.

El panel de control se compone de los tres sliders principales, que controlan la posición de los planos de vistas, y de varios checkbox que controlan qué información se desea mostrar u ocultar (visualización de planos, del implante o del renderizado tridimensional). Por defecto, la visualización del implante estará activada, y la de los planos se activará automáticamente al cargar la información de la malla. Por otra parte, la visualización del renderizado en la ventana tridimensional aparecerá desactivada, ya que reduce sensiblemente el rendimiento general del programa.

Si se mueve manualmente alguno de los sliders de la ventana de control, manteniendo el botón izquierdo del botón pulsado, se podrá ver en tiempo real el movimiento de los planos correspondientes en la ventana de vista tridimensional. Al terminar el movimiento, y cuando se libera el botón presionado, la información se actualiza de manera general.

Las ventanas de vistas, en las que se muestran los cortes de la imagen médica determinados por los diferentes planos, disponen también de controles de dibujo. Se trata de dos sliders por ventana, y controlan la posición los dos planos que no afectan a la ventana en cuestión. De nuevo, el movimiento manual de éstos mostrará una previsualización de la posición de los planos en cada instante, pero solo en esa ventana. Cuando se termine de cambiar la posición de éstos se apreciarán los cambios en el resto del programa.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

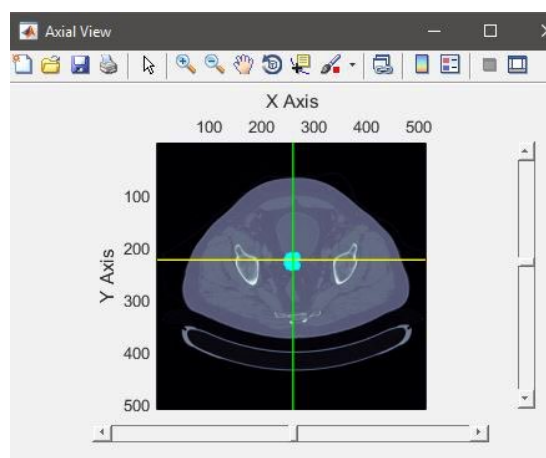


Figura 7.- Ventana de vista. Se muestra la ubicación de los dos sliders y de los planos de corte.

Los controles de dibujo explicados en este apartado son los más empleados en el uso normal de *Implant Viewer*. Sin embargo, en el panel de preferencias del programa (apartado 5.7) existen algunos parámetros adicionales que sirven para modificar la información que se muestra gráficamente por pantalla. Su comprensión permitirá adecuar la visualización a las características propias de la información que se está tratando.

Nota: En el programa existen dos modos de renderizado: uno general (descrito en el apartado 5.6.3) y otro que muestra la información 3D en la ventana de vistas (es el desarrollado en esta sección). Es posible que el programa vea reducido su rendimiento mientras carga el modelo al renderizar.

5.4. Movimiento del Implante.

El objetivo fundamental de este software requiere de un posicionamiento adecuado del implante sobre la imagen médica. Ese posicionamiento se puede dividir en conjuntos de traslaciones y rotaciones, puesto que la combinación de ambos tipos de movimiento permite conocer la ubicación exacta del objeto en cuestión. Las opciones de que dispone *Implant Viewer* a este respecto son descritas a continuación.

5.4.1. *Traslación del Implante.*

Este tipo de movimiento permite ubicar el objeto en una zona determinada de la imagen médica, sin modificar su orientación espacial. Existen dos alternativas a este respecto en el software, y dependen de si la distancia a mover es conocida o no.

La primera opción está ubicada en la sección *Translation* de la ventana principal del programa, y se suele emplear cuando se conocen las distancias exactas que son necesarias para colocar el implante. Tal y como se muestra en la figura 8, este conjunto de opciones se compone de varios botones y zonas de introducción de texto. Cabe destacar que las traslaciones se miden en píxeles, y no en las unidades habituales de distancia.

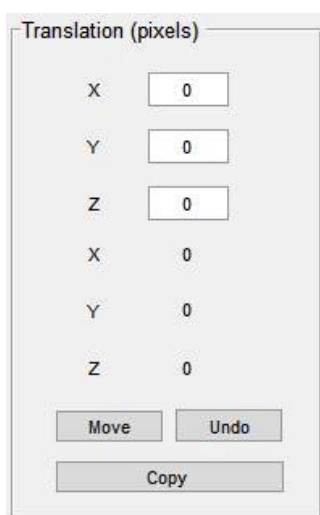


Figura 9.- Panel de controles de traslación

En la parte superior de esta sección se pueden encontrar tres bloques de introducción de datos. Se trata de las casillas de introducción de distancias sobre los ejes *x*, *y* y *z*. En la práctica, estos valores pueden ser considerados como incrementos a lo largo de los ejes, y es muy importante notar que *Implant Viewer* toma los tres valores a la vez para realizar las estimaciones de movimiento, por lo que siempre debe verificarse que se coloca un cero en aquellos ejes sobre los que no se desee realizar movimiento alguno.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

Una vez introducidos los valores correspondientes se deberá presionar el botón *Move* para efectuar el movimiento. Si éste ha sido erróneo puede emplearse el botón *Undo*, que tiene un alcance de hasta cuatro pasos.

Justo en la parte superior de estos botones se ofrece al usuario información sobre los píxeles totales que se ha desplazado el objeto en términos globales. Estos valores no pueden ser modificados por el usuario, y pueden ser copiados en las casillas pulsando sobre el botón *Copy*. Esta función es particularmente útil cuando, una vez colocado el implante, se quiere introducir una nueva malla sin borrar los datos de la sesión.

La segunda opción para mover el implante resulta ser la más sencilla e intuitiva, ya que consiste en arrastrar directamente el objeto a través de las ventanas de vistas. En concreto, se debe hacer click izquierdo sobre la sección de implante de la vista en cuestión hasta que ésta sea de color rojo. Luego, dejando pulsado el botón, se arrastrará hasta la posición final. Cuando se libere el ratón se efectuarán los cambios de movimiento.

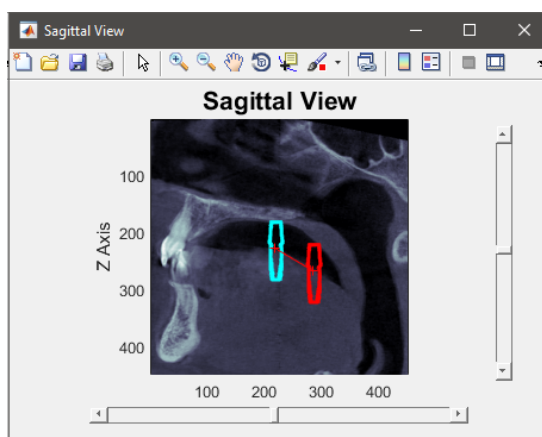


Figura 10.- Proceso de movimiento manual por medio del ratón en las ventanas de vistas.

5.4.2. Rotación del Implante.

El movimiento de rotación, por su parte, modifica la orientación espacial del implante sin afectar a la posición del mismo con respecto al origen del sistema de referencia. El giro

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

del implante se puede realizar sobre cualquiera de los tres planos de vistas y teniendo como referencia cualquier punto deseado. Al igual que en el caso de la traslación, existen dos alternativas para girar el objeto: una que implica conocer el ángulo de giro y otra en la que no es necesario.

La primera opción, de igual manera que en el otro tipo de movimiento, se encuentra en la ventana principal o de control, en el conjunto de opciones que se agrupan bajo el nombre de *Rotation*. Se trata esencialmente de un selector de eje de giro con sendas casillas asociadas a dichos ejes.

Una vez seleccionado el eje de giro, el programa requiere de la introducción del ángulo de giro en grados (valores positivos implican un giro en el sentido del eje positivo) y pedirá en la ventana de vista correspondiente un punto como referencia de giro. Una vez dado todo esto, el giro se efectúa en el resto del software.

Es necesario destacar que en la sección de rotación de la ventana de control también se encuentra un botón *Undo* que llama a la función deshacer principal. Tiene hasta cuatro pasos de retroceso.

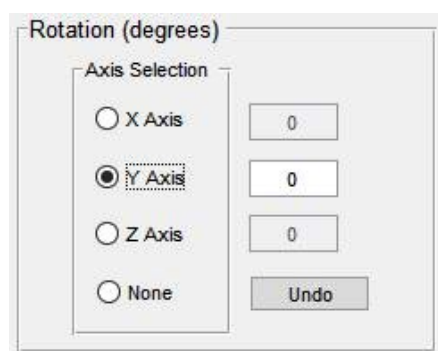


Figura 11.- Paneles de control del giro del implante

La segunda opción es una alternativa más rápida, sencilla e intuitiva. Consiste en seleccionar la ventana de vista que refleja el eje sobre el cual se desea rotar y hacer y mantener el click derecho (o Ctrl + Click izquierdo) hasta que la imagen del objeto aparezca en color rojo. En ese momento, para girar el implante solamente es necesario mover el

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

ratón hasta el ángulo deseado y soltar el botón para que los cambios se realicen. El giro se realiza con respecto al plano de vista de la ventana seleccionada.

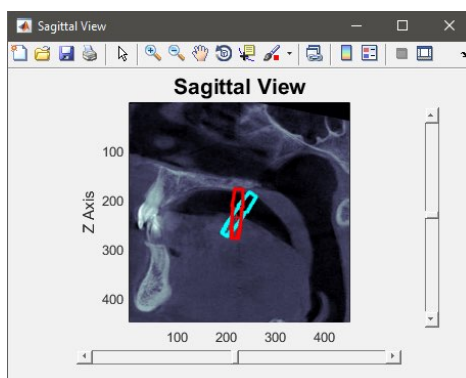


Figura 12.- Proceso de rotación manual por medio del ratón en las ventanas de vistas

5.5. Otras Funciones.

Además de las características principales ya detalladas del programa, existen algunas otras funciones útiles que complementan la tarea de colocar el implante sobre la imagen médica y permiten expandir las posibilidades de *Implant Viewer*.

5.5.1. Corte de la Imagen Médica.

Pese a que el objetivo principal de este programa es la colocación de un objeto sobre una imagen en formato DICOM, es posible que, por las dimensiones de la misma o por el grado de carga de cálculo del programa interese reducir la información. Para ello, *Implant Viewer* dispone de dos modos de corte que se pueden localizar en la pestaña DICOM del menú principal (véase apartado 5.6).

Los dos modos de corte requieren conocer la posición de la nube de puntos que representa al implante, por lo que, si se han modificado ya los píxeles correspondientes a la

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

ubicación del mismo, es importante que no se mueva el objeto. Las dos opciones de corte de la imagen son:

- **bCut:** Esta es una alternativa de corte que no requiere de control por parte del usuario. Su nomenclatura significa “boundary cut” o corte por frontera, y lo que hace es meter el implante en una caja ajustada a sus dimensiones, obviando el resto de la imagen médica.
- **pCut:** pCut o “Plane Cut” emplea los tres planos de vistas para limitar las dimensiones de la imagen que se quiere obtener como resultado. Es capaz de detectar en qué parte de la imagen se encuentra la nube de puntos y elimina las zonas restantes. El resultado es una zona más acotada y que permite un mejor manejo de la información importante.

5.5.2.Reparación del Implante.

Al discretizar el implante en elementos tipo cubo y leer con *Implant Viewer* el archivo resultante, se desencadena un algoritmo que transforma la información bruta de la malla inicial en puntos que representan los centros de elemento. Estos puntos están referenciados respecto al conjunto en unidades de píxel.

Esta apreciación, pese a no ser del todo relevante para el usuario, genera un problema a tener en cuenta a la hora de colocar el objeto sobre la imagen médica. Para que la escala del objeto sea acorde a las dimensiones de la imagen del paciente, al realizar la malla se tienen en cuenta las medidas unitarias del vóxel de ésta. No obstante, en el proceso de conversión a unidades en píxeles se pierde la información referente a las dimensiones reales, por lo que el modelo sería correcto en la orientación original en la que se ha creado la malla, lo que significa que el implante, al ser girado puede dejar hileras de píxeles sin tener en cuenta.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

Si el objeto ha sido girado y posicionado correctamente y se ha modificado la imagen médica sin tener en cuenta esta problemática, será posible apreciar píxeles del entorno del objeto sin la información del mismo. Una vez ha ocurrido esto, el usuario tiene dos alternativas de subsanación: girar el implante en GiD a la orientación correcta (véase la sección 5.5.4) y volverlo a introducir en el programa o utilizar la herramienta *hFix* disponible en la pestaña DICOM del menú de navegación superior.



Figura 13.- Implante en el archivo *DICOM* con "píxeles huecos"

La herramienta *hFix* se emplea después de haber modificado la imagen médica. Se trata de un algoritmo que reconstruye el archivo *DICOM* en función de la relación de aspecto de los píxeles originales, obteniéndose como resultado el implante sin huecos de información, salvo los que ya tuviera inicialmente el objeto. Esto constituye una gran ventaja, al no tener que depender de otros programas (como GiD) para colocar correctamente el implante.



Figura 14.- Implante reparado con la función *hFix*. El hueco central, original del modelo, se ha respetado.

Nota: Para el correcto funcionamiento de *hFix* se recomienda que, aunque se haya modificado el archivo, la nube de puntos se mantenga sobre la ubicación original, para poder acotar el conjunto de vóxeles a corregir de manera más precisa.

5.5.3. Renderizado.

Es notable el hecho de que el programa basa su funcionamiento principal en las vistas bidimensionales que resultan del corte de la imagen médica. No obstante, es posible que en algunas ocasiones sea necesario que el usuario obtenga una visualización tridimensional que le ayude a saber qué es lo que está viendo realmente. *Implant Viewer* contempla esa opción, y está disponible en el menú *Edit*.

Al pulsar sobre esta función, aparecerá en primer lugar un mensaje de advertencia que recuerda al usuario que el proceso de renderizado puede minimizar el rendimiento del programa, por lo que se recomienda bajar la resolución en el panel de preferencias (sección 5.6 de este documento). Si se acepta ese mensaje el programa iniciará el proceso y aparecerá una nueva ventana que muestra una reconstrucción superficial de los volúmenes deseados, en función del valor Hounsfield aproximado dado en el panel de preferencias (véase sección 5.7). Es posible la nueva ventana tarde un poco en aparecer, pues debe cargar mucha información previa.

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

Nota: El renderizado general es diferente del explicado para la ventana de vista tridimensional, que se activa desde la sección de controles de dibujo del panel de control principal.

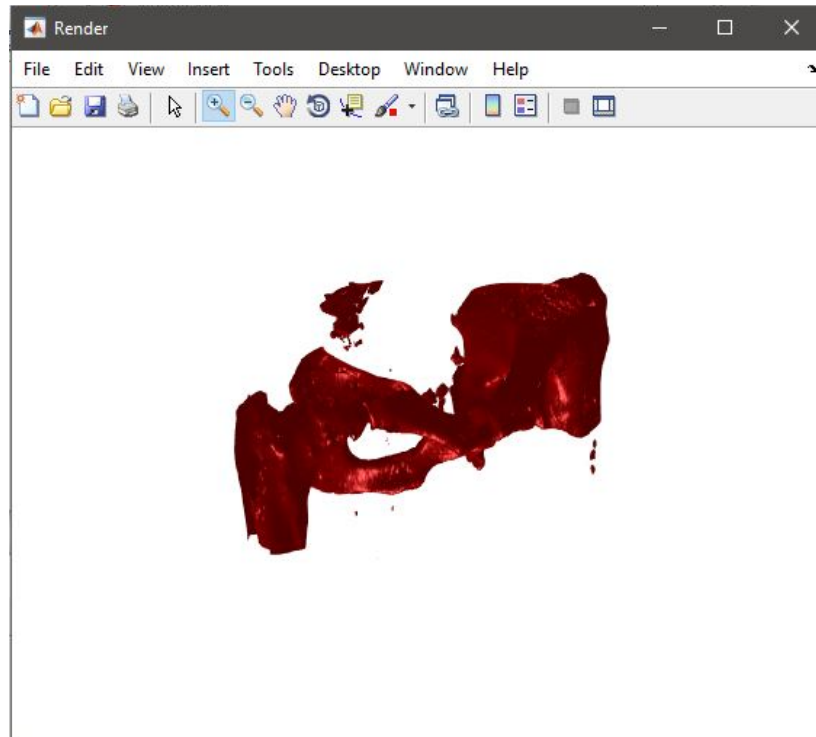


Figura 15.- Muestra del renderizado de un archivo *DICOM*

5.5.4. Ángulos de Euler (ZYZ).

Las dimensiones de los vóxeles de las imágenes médicas tienen son desiguales a lo largo de los ejes de referencia principales. Esto, como ya se ha mencionado en el apartado 5.5.2, genera un problema de pérdida de información al girar el implante.

Pese a que *Implant Viewer* posee algoritmos que corrigen esto por completo (hFix), es posible que el usuario prefiera girar el modelo en otro programa. En previsión de esto, el software está preparado para escribir en un archivo *TXT* la información de orientación y posición del objeto en cada momento. Se puede realizar a través de la opción correspondiente del menú *File*, tanto a través del submenú correspondiente como cuando

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

se pulsa *Save*, pues se recordará al usuario que puede realizar esto antes de guardar el archivo general.

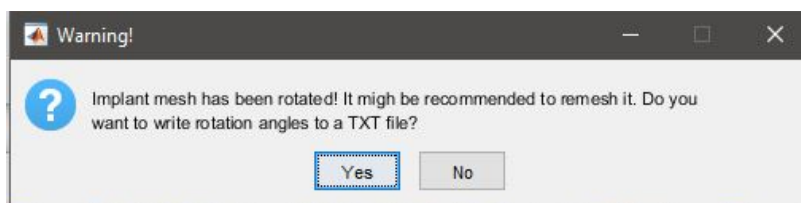


Figura 16.-*Warning* que advierte al usuario de la posibilidad de guardar la información de posición y orientación del implante en un fichero *TXT*

Nota: Es posible que, para ciertas orientaciones, los ángulos de Euler puedan ser erróneos. Esto ocurre muy pocas veces, y se debe a singularidades en la definición de los mencionados ángulos.

5.6. Menú Principal.

En el menú principal pueden encontrarse la mayoría de funciones ya explicadas en las diferentes secciones que hacen referencia a la ventana de control. Además de esto, también dispone de otras funciones que complementan la actividad propia de *Implant Viewer*. En este apartado se muestra una visión general de los contenidos del menú de navegación principal, que se divide en tres partes bien diferenciadas: las pestañas *File*, *Edit* y *DICOM*.

5.6.1. Pestaña *File*.

En este submenú pueden encontrarse todas aquellas opciones importantes para el funcionamiento habitual del software. Incluye el acceso a las preferencias y ajustes que regulan el tratamiento de la información, pero también contiene la lectura de malla, el guardado de la sesión y la función de obtención de los ángulos de Euler (en el orden ZYZ).

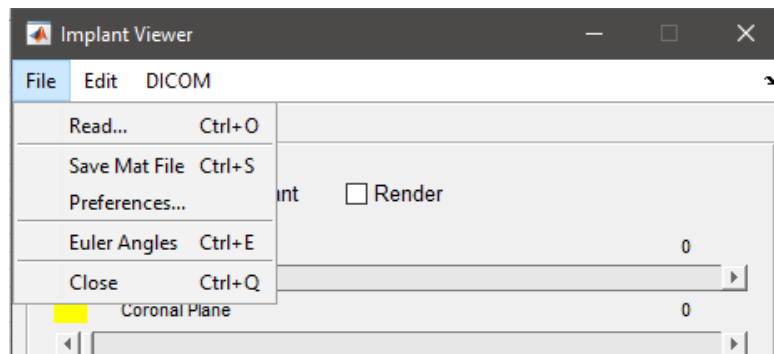
MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA


Figura 17.- Pestaña *File*

5.6.2. Pestaña *Edit*.

En la sección *Edit* están disponibles las opciones que pueden modificar la información que se está mostrando. Hay botones para deshacer el último paso, redibujar, renderizar, eliminar y recortar los datos.

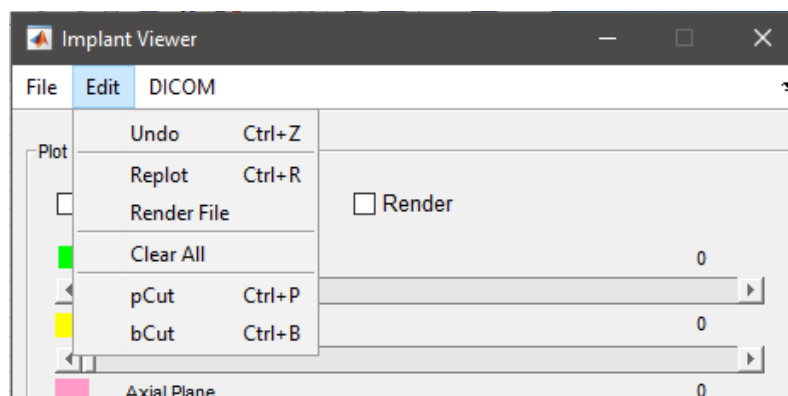


Figura 18.- Pestaña *Edit*

5.6.3. Pestaña *DICOM*.

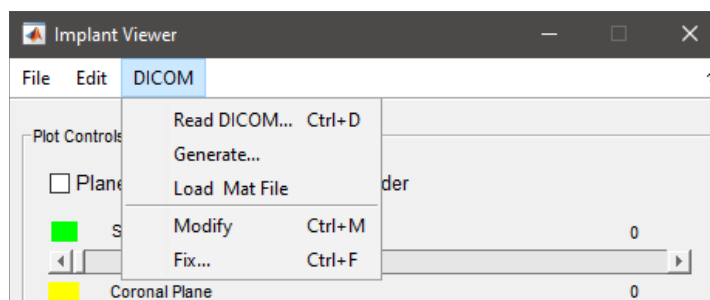
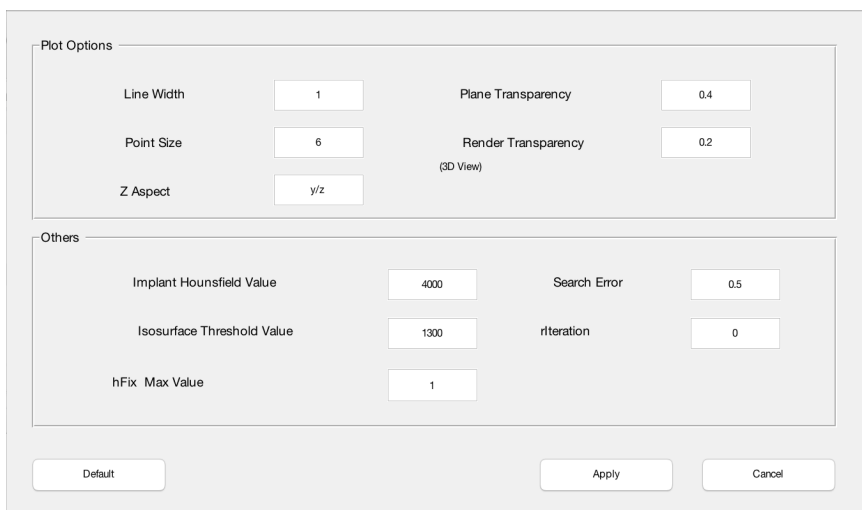


Figura 19.- Pestaña *DICOM*

En esta parte se encuentran las funciones que afectan a la imagen médica propiamente dicha. Las más utilizadas son las mismas que están disponibles en los botones de la ventana principal (lectura y modificación de archivos *DICOM*). No obstante, pueden encontrarse otras funciones tales como *Fix*, *Generate* o *Load Mat File*.

- ***Fix***: Esta opción desencadena el algoritmo interno de reparación del objeto (sección 5.5.2).
- ***Generate***: Origina una imagen médica con ceros de dimensiones 512x512x512 píxeles.
- ***Load Mat File***: Esta función permite al usuario introducir archivos MAT con matrices 3D que hagan las veces de imagen médica. Pueden cargarse incluso archivos *DICOM* que ya hayan sido tratados con *Implant Viewer* previamente.

5.7. Panel de Preferencias.



Plot Options	
Line Width	1
Point Size	6
Z Aspect	y/z
Plane Transparency	0.4
Render Transparency (3D View)	0.2

Others	
Implant Hounsfield Value	4000
Isosurface Threshold Value	1300
hFix Max Value	1
Search Error	0.5
Iteration	0

Buttons: Default, Apply, Cancel

Figura 20.- Panel de preferencias

En esta parte del programa se encuentran los parámetros que controlan de manera elemental el tratamiento de la información con la que se está trabajando en un momento determinado. Mediante la modificación de los parámetros aquí expuestos se pueden controlar aspectos tales como el modo en que se visualiza la información o el tipo de información con la que se va a editar el archivo original.

El panel de ajustes posee tres botones principales, y son el seteo de valores por defecto, la aplicación de cambios y el cierre de ventana o cancelación de cambios. Además, las diferentes variables se agrupan según el aspecto del programa al que afectan, con el fin de minimizar el tiempo de búsqueda.

5.7.1. Plot Options.

En este apartado de las preferencias del programa se pueden localizar aquellos parámetros que se controlan aspectos relativos al dibujo de la información. Las variables disponibles son:

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA

UNIVERSIDAD POLITÉCNICA DE VALENCIA

- **Line Width:** Define el grosor con el que se van a plotear las líneas que representan los planos de corte en las ventanas de vistas bidimensionales.
- **Plane Transparency:** Grado de transparencia de visualización de los planos de corte en la vista de perspectiva.
- **Point Size:** Tamaño de dibujo de los puntos que representan al objeto tridimensional.
- **Render Transparency:** Transparencia del modelo tridimensional de la imagen médica al ser representado en la ventana de vista en perspectiva.
- **Z Aspect:** Define la relación entre la medida de los píxeles en la dirección Y y Z. Por defecto, el programa calcula la relación para establecer una representación realista de la imagen, pero si el usuario lo desea puede introducir manualmente una. Esto es útil principalmente cuando se trabaja con archivos MAT en los que no se conocen las dimensiones de los píxeles. en estos casos el usuario deberá medir sobre la pantalla las dimensiones de los ejes Y y Z e introducir la relación en el panel de preferencias.

5.7.2.Others.

Las opciones de esta sección son de diferente naturaleza, pero en términos generales son las que manejan y modifican la información de la imagen médica y el implante:

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA

- **Implant Hounsfield Value:** Valor en la escala hounsfield por el que se van a sustituir los píxeles que se encuentran en la misma orientación y posición que el objeto.
- **Search Error:** Es el error admisible, medido en píxeles, que se genera al buscar los puntos del implante que coinciden con la posición de los planos de corte.
- **Isosurface Threshold Value:** Valor de isosuperficie a tener en cuenta para la formación de los renderizados tridimensionales de la información médica.
- **rliteration:** Grado de disminución de la resolución al calcular las superficies de los renderizados tridimensionales. A mayor número, menor resolución y mayor rapidez de cálculo.
- **hFix Max Value:** Se trata de un parámetro que controla el número de píxeles de alcance de comprobación para la rutina de reconstrucción hFix. El programa lo determina automáticamente, pero el usuario puede introducirlo si lo desea.

6. Atajos de Teclado.

Algunas de las funciones más interesantes de este software pueden ser accedidas mediante atajos de teclado. Una lista de ellas se detalla a continuación:

Función	Acceso de Teclado
Leer Objeto	Ctrl + O
Guardar en Archivo MAT	Ctrl + S
Ángulos de Euler	Ctrl + E
Cerrar Programa	Ctrl + Q
Deshacer	Ctrl + Z
Redibujar	Ctrl + R
pCut	Ctrl + P
bCut	Ctrl + B
Leer DICOM	Ctrl + D
Modificar DICOM	Ctrl + M
Reconstruir (hFix)	Ctrl + F

MÁSTER UNIVERSITARIO EN INGENIERÍA MECÁNICA
UNIVERSIDAD POLITÉCNICA DE VALENCIA