



UNIVERSIDAD DE JAÉN
Escuela Politécnica Superior (Jaén)

Trabajo Fin de Máster

GENERACIÓN DE CONDICIONES DE CONTORNO PARA LA SIMULACIÓN DEL FLUJO OSCILATORIO DEL LÍQUIDO CEFALORRAQUÍDEO EN EL CANAL ESPINAL

Alumno/a: Cifuentes González, Ángel

Tutor/a: Gutiérrez Montes, Cándido
Dpto.: Ingeniería Mecánica y de Minas

Septiembre, 2019



Universidad de Jaén
Escuela Politécnica Superior de Jaén
Departamento de Informática

Don Cándido Gutierrez Montes , tutor del Proyecto Fin de Carrera titulado: Generación de condiciones de contorno para la simulación del flujo oscilatorio del líquido cefalorraquídeo en el canal espinal, que presenta Ángel Cifuentes González, autoriza su presentación para defensa y evaluación en la Escuela Politécnica Superior de Jaén.

Jaén, Septiembre de 2019

El alumno:

Ángel Cifuentes González

Los tutores:

Cándido Gutiérrez Montes

Índice

1. INTRODUCCIÓN	6
1.1. Importancia del estudio.	6
1.2. Fundamento fisiológico.	7
1.3. Aplicaciones clínicas.	10
1.4. Objetivos del trabajo.....	10
2. MARCO TEÓRICO.....	12
2.1. Ecuaciones que gobiernan la Mecánica de Fluidos.	12
2.1.1. Teorema de Transporte de Reynolds	13
2.1.2. Ecuación de continuidad	14
2.1.3. Ecuación de cantidad de movimiento	15
2.2. Introducción a la Mecánica de Fluidos Computacional (CFD).	16
2.3. Método de las diferencias finitas.	17
2.4. Método de Volúmenes Finitos.	22
2.4.1. Ecuación de transporte escalar	22
2.4.2. Celdilla computacional y nomenclatura.	22
2.4.3. Discretización espacial y temporal.	24
2.4.4. Evaluación de las integrales de volumen:.....	25
2.5. Simulaciones FSI.	29
2.6. Métodos de malla dinámica.....	31
2.6.1. Métodos Smoothing.	32
2.6.2. Método Layering.	35
2.6.3. Método Remeshing.	36
2.6.4. Selección del método de malla dinámica.....	36
2.6.4. Macros para UDFs en simulaciones con métodos de malla dinámica.	37
3. DEFINICIÓN Y OBTENCION DE LOS DOMINIOS COMPUTACIONALES.	40
3.1 Geometría Simplificada.	40
3.2. Geometría real.	44
4. OBTENCIÓN DE LAS UDF PARA MOVIMIENTO DE LA PARED.	47
4.2. Introducción a las UDFs.	47
4.2. Obtención de la UDF con caudal real.....	48
4.1.1. Obtención de la señal de caudal.	49

4.1.1. Amortiguamiento y desfase de la señal.....	51
4.2. Obención de la UDF para los casos canónicos.....	56
5. CONFIGURACIÓN DE LOS CASOS.....	59
6. ANÁLISIS DE LOS RESULTADOS.....	62
6.1. Casos con geometría simplificada.....	62
6.1.2. Resultados casos canónicos.....	62
6.1.1 Caso canónico 1, $\beta=0$	63
6.1.2 Caso canónico 2, $\beta=0.25$	67
6.1.3. Caso canónico 3, $\beta=0,5$	73
6.1.4. Caso canónico 4, $\beta=0,75$	78
6.1.2. Resultados del caso con caudal realista.....	85
6.2. Geometría simplificada con caudal real.....	88
7. CONCLUSIONES.....	91
Bibliografía	94
APÉNDICES.	95
APÉNDICE A. CÓDIGOS DE LAS UDFs	95
APÉNDICE B. CÓDIGOS DE POST PROCESADO EN MATLAB	98

Índice de figuras

Figura 1.1. Estructura de las meninges.	8
Figura 1.2. Sección de la médula espinal.	8
Figura 1.3. Esquema de circulación del CSF [1].	9
Figura 2.1. Ejemplo de mallas computacionales 1D y 2D [2]	18
Figura 2.2. Representación geométrica de la derivada [2]	19
Figura 2.3. Estructura de una matriz para una malla computacional de 3 puntos.	21
Figura 2.4. Representación de una celdilla computacional [3]	23
Figura 2.5. Esquema de los modelos FSI	30
Figura 2.6. Esquema del acoplamiento entre campos sólidos y fluidos en modelos FSI.....	31
Figura 2.7 Ejemplo de malla deformada con el método ‘Smoothing-Spring’ [4]	33
Figura 2.8. Representación de las capas cercanas a un contorno móvil en una malla dinámica. [4].....	35
Figura 2.9 Metodología utilizada en [5] para reproducir un caudal real en una geometría compleja mediante simulaciones con malla móvil.....	39
Figura 3.1. Geometría simplificada en un caso centrado.	41
Figura 3.2. Dimensiones de la geometría simplificada.	41
Figura 3.3. Capturas de la entrada de la malla en: a) caso $\beta=0$, b) caso $\beta=0.25$, c) caso $\beta=0.5$, d) caso $\beta=0.75$	42
Figura 3.4. Representación de las mallas obtenidas para los diferentes valores de β	43
Figura 3.5. Histograma del parámetro de oblicuidad de las mallas para los casos: a) $\beta =$ 0 , b) $\beta = 0.25$,c) $\beta = 0.5$,d) $\beta = 0.75$	43
Figura 3.6. Malla triangular y superficie dividida en sub-superficies.....	45
Figura 3.7. Geometría de la médula espinal suavizada con SolidWorks.....	46
Figura 3.8. Malla computacional de la geometría realista.	46
Figura 4.1 Señales de flujo másico a lo largo del tiempo para diferentes alturas de la médula [5].....	50
Figura 4.2. Curva de caudal en el foramen magnum interpolada a partir de los datos de [5].	51
Figura 4.3 Datos de Stroke Volume y Desfase de la señal a diferentes alturas [6]	52
Figura 4.4 Funciones interpoladas para el coeficiente de amortiguamiento y desfase de la señal de caudal.	53
Figura 4.5 Mapa de flujo volumétrico para la señal de caudal real.	54
Figura 4.6. Mapa de desplazamientos obtenidos de un caudal real.	55
Figura 4.7 Señal de desplazamiento para un caudal real a diferentes alturas.	56
Figura 4.8. Mapa de Flujo volumétrico.....	57
Figura 4.9 Mapa de desplazamientos obtenidos de un caudal senoidal.	58
Figura 6.1. Velocidades longitudinales en el punto medio de cada canal para $\beta = 0$	63
Figura 6.2 Comparación entre los caudales obtenidos mediante simulación y los teóricos para $\beta = 0$	64
Figura 6.3. Posición de los puntos tomados para representar la velocidad.....	65
Figura 6.4 Comparativa entre las velocidades oscilatoria y estacionaria en diferentes puntos para $\beta = 0$	66
Figura 6.5. Perfil de velocidad estacionaria para $\beta = 0$	66
Figura 6.6 Velocidades longitudinales en el punto medio de cada canal para $\beta = 0.25$	67

Figura 6.7 Comparación entre los caudales obtenidos mediante simulación y los teóricos para $\beta = 0.25$	68
Figura 6.8. Posición de los puntos tomados para representar la velocidad.....	69
Figura 6.9. Comparativa entre las velocidades oscilatoria y estacionaria en diferentes puntos del canal ancho para $\beta = 0$	70
Figura 6.10. Comparativa entre las velocidades oscilatoria y estacionaria en diferentes puntos del canal estrecho para $\beta = 0.25$	71
Figura 6.11. Perfil de velocidades estacionarias para el canal ancho en el caso $\beta = 0.25$...	72
Figura 6.12. Perfil de velocidades estacionarias para el canal estrecho en el caso $\beta = 0.25$...	72
Figura 6.13 Velocidades longitudinales en el punto medio de cada canal para $\beta = 0.5$	73
Figura 6.14 Comparación entre los caudales obtenidos mediante simulación y los teóricos para $\beta = 0.5$	74
Figura 6.15. Posición de los puntos tomados para representar las velocidades estacionarias.	74
Figura 6.16. Comparativa entre las velocidades oscilatoria y estacionaria en diferentes puntos del canal ancho para $\beta = 0.5$	76
Figura 6.17. Comparativa entre las velocidades oscilatoria y estacionaria en diferentes puntos del canal estrecho para $\beta = 0.5$	76
Figura 6.18. Perfiles de velocidad estacionaria en el canal ancho para $\beta = 0.5$	77
Figura 6.19. Perfiles de velocidad estacionaria en el canal estrecho para $\beta = 0.5$	78
Figura 6.20 Comparación entre los caudales obtenidos mediante simulación y los teóricos para $\beta = 0.75$	79
Figura 6.21. Velocidades longitudinales en el punto medio de cada canal para $\beta = 0.75$	80
Figura 6.22. Puntos tomados para representar la velocidad estacionaria.	81
Figura 6.23. Comparativa entre las velocidades oscilatoria y estacionaria en diferentes puntos del canal ancho para $\beta = 0.75$	82
Figura 6.24 Comparativa entre las velocidades oscilatoria y estacionaria en diferentes puntos del canal estrecho para $\beta = 0.75$	82
Figura 6.25. Perfiles de velocidad estacionaria en el canal ancho para el caso $\beta = 0.75$	83
Figura 6.26 . Perfil de velocidades estacionarias en el canal estrecho para el caso de $\beta = 0.75$	84
Figura 6.27. Contorno de velocidades axiales para $\beta = 0.75$	84
Figura 6.28. Comparación de los caudales impuestos y obtenidos de la simulación.	85
Figura 6.29. Velocidades longitudinales en el punto medio de cada canal para el caudal real.	86
Figura 6.30. Perfil de velocidades en el canal ancho para la simulación con caudal real.....	87
Figura 6.31. Perfil de velocidades en el canal estrecho para la simulación con caudal real..	87
Figura 6.32. Comparación de los caudales obtenidos de simulación e impuestos en una geometría realista y caudal real.....	89
Figura 6.33. Velocidades longitudinales en el punto medio de cada canal para un caudal real y geometría realista.	90
Figura 7.1.....	92

1. INTRODUCCIÓN

El objeto del trabajo de fin de máster es la generación de las condiciones de contorno para la simulación del flujo oscilatorio del líquido cefalorraquídeo en el canal espinal mediante técnicas CFD, para ello se emplearán softwares comerciales tanto para la generación del código que permitirá dicho movimiento como para las simulaciones realizadas para validar que el movimiento implementado es el deseado.

Debido a la complejidad del caso, se generará una geometría sencilla y se simulará en la misma, ya que será necesario menos potencial de cálculo. Finalmente se generará la geometría realista, la cual será simulada con el movimiento implementado, en función de la señal de caudal de flujo cefalorraquídeo deseado.

1.1. Importancia del estudio.

El líquido cefalorraquídeo o cerebroespinal (CSF, 'Cerebrospinal fluid') tiene una importancia vital en nuestro organismo ya que dicho flujo se encuentra entre las meninges, que cubren todo el Sistema nervioso central (CNS, 'Central nervous system'), como se muestra en la Figura 1.1, y actúa dando apoyo al sistema inmunológico, como protección estructural, y como apoyo a la homeostásis metabólica del CNS.

Estudios recientes han mostrado la posibilidad de conducir fármacos mediante el CSF para tratar problemas neuronales y del CNS, lo cual ha aumentado el interés en estudiar el comportamiento del mismo. La importancia de la dinámica es tal que ha sido investigada en estudios sobre enfermedades del CNS como la siringomielia, Alzheimer, hidrocefalia, etc. En cuanto al transporte de fármacos, se ha podido observar, en determinados estudios de seguimiento, que los fármacos transportados dentro del CSF se transportan dentro y fuera del tejido cerebral a través de caminos perivasculares o leptomeníngeos.

A pesar de la visible importancia del papel del CSF la información sobre la biomecánica del fluido es escasa. Sin embargo, el CSF es una prometedora ruta de transporte de fármacos ya que permite una focalización directa del CNS permitiendo

de ésta forma minimizar los efectos que provocan los tratamientos intravenosos y orales convencionales, además de permitir el transporte de moléculas con un tamaño que no podía ser transportado a través de la barrera hematoencefálica.

La importancia de muchos de dichos tratamientos dependerá en parte del transporte dentro del espacio sub aracnoidé (SAS, 'subarachnoid space'), ya que las estructuras afectan de manera directa al flujo que circula por el canal espinal, motivo por el que se estudia el efecto del mismo, habiéndose demostrado que el hecho de considerar las raíces nerviosas (NR, 'Nerve roots') en un modelo aumentan el tiempo de transporte con respecto a modelos simplificados, por lo que para éste trabajo se utilizará un modelo simplificado en el que no se tienen en cuenta los nervios.

Por otro lado, diferentes estudios, como los de Di Chiro en 1964, demostraron la rápida migración de compuestos inyectados en los ventrículos cerebrales hacia el canal espinal, lo cual ha sido corroborado con numerosos estudios radiológicos, dando apoyo a la hipótesis de que existen, aparte de los flujos bifásicos, un mecanismo de circulación activa relacionado con el transporte de CSF en la médula espinal. Posteriores estudios han confirmado que en el caso de adultos, partículas trazadoras inyectadas en la región lumbar se pueden detectar en las cisternas basales y la zona craneal en aproximadamente 15-20 minutos tras la inyección, lo que sugiere que dichas velocidades son del orden del centímetro por minuto.

Por tanto, uno de los principales objetivos de dicho estudio es generar unas condiciones de contorno tales que permitan captar dichos fenómenos para poder desarrollar un modelo que permita mejorar la comprensión del mismo.

1.2. Fundamento fisiológico.

Como se ha mencionado, el líquido cefalorraquídeo se encuentra circulando por el interior del canal espinal, entre las meninges, unas membranas de tejido conectivo que cubren todo el sistema nervioso central y que en mamíferos presentan tres capas con dos espacios intermedios como se muestra en la Figura 1.2, que son:

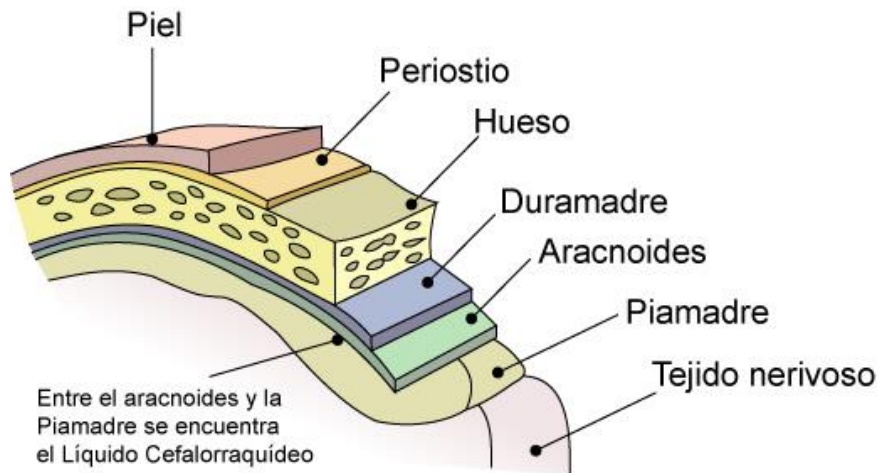


Figura 1.1. Estructura de las meninges.

- **Piamadre:** es la capa más interna de las meninges y por tanto, la que se encuentra en contacto con el tejido nervioso, el tejido cerebral y con el encéfalo. Es una capa muy vascularizada y contiene fibroblastos.
- **Espacio subaracnoideo, SAS:** es el espacio por el que circula el CSF y amortigua golpes, reduciendo la posibilidad de traumatismos.
- **Aracnoides:** es una capa vascular atravesada por vasos sanguíneos hacia la piamadre y se denomina leptomeninge al conjunto formado por el aracnoides y la piamadre.
- **Espacio subdural:** es el espacio que separa el aracnoides y la duramadre y por él circula algo de líquido cefalorraquídeo.
- **Duramadre:** es la capa exterior de las meninges, a la cual le impondremos un movimiento para realizar las simulaciones.

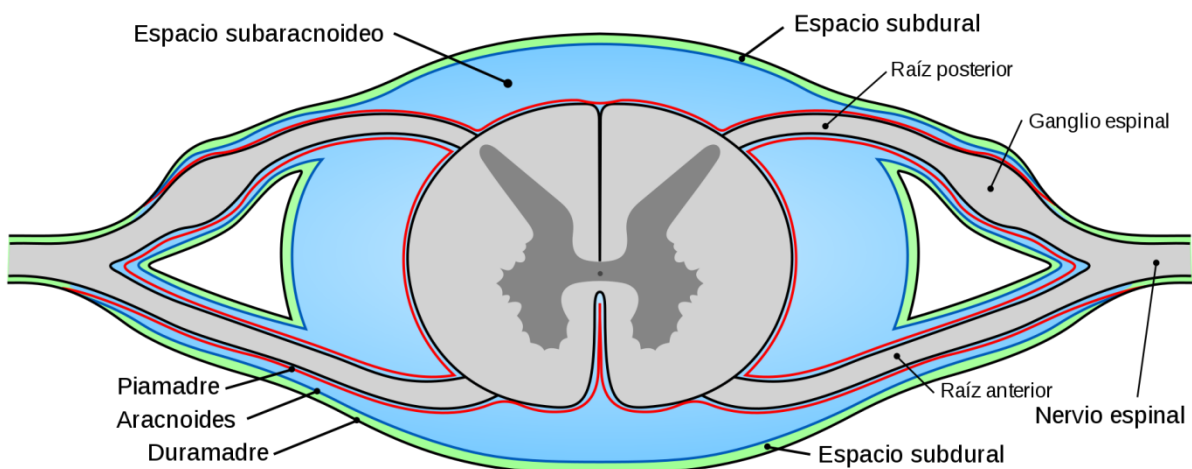


Figura 1.2. Sección de la médula espinal.

Como se ha comentado, las funciones de las meninges son las de proteger frente a ataques químicos ya que actúa como una barrera impidiendo la entrada de sustancias perjudiciales para el sistema nervioso que pueden causar infecciones como la meningitis, y aportar protección mecánica ya que el CSF amortigua los golpes, lubrica y nutre a los haces de mielina que recubren el CNS.

El CSF es uno de los tres principales elementos que contiene el cráneo y ocupa un 10% del volumen intracraneal. El volumen de CSF en adultos varía entre 75 y 270 ml, de los cuales aproximadamente entre el 20 y el 50% se encuentra en el canal espinal. Su circulación se produce libremente entre el SSS encefálico y espinal. Los factores que generan el movimiento del CSF son:

- Impulsos desde las áreas desde las que se produce el fluido hasta donde éste se absorbe.
- Oscilación continua a lo largo del canal espinal, presentando una mayor amplitud a medida que se aproxima al cuarto ventrículo del cerebro, cavidad situada por encima de la médula espinal y que presenta forma triangular. La amplitud es aproximadamente un 60% menor en la zona lumbar que en la zona de los ventrículos.
- Impulsos por sístole cardíaca o inhalación respiratoria.

En la se muestra de manera gráfica el sentido del flujo de CSF a lo largo del canal espinal.

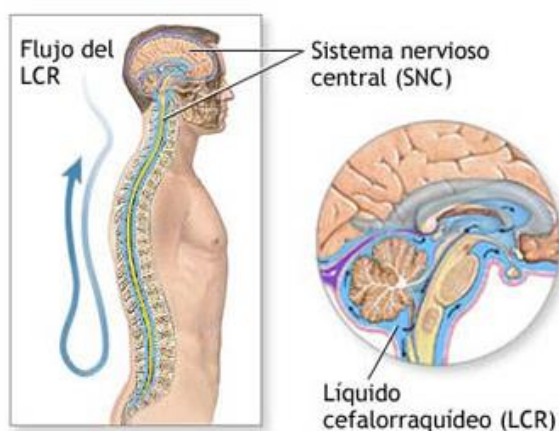


Figura 1.3. Esquema de circulación del CSF [1].

1.3. Aplicaciones clínicas.

Más allá de la ampliación del conocimiento sobre el comportamiento del CSF, los estudios pueden aportar grandes beneficios para la medicina como pueden ser: desarrollo nuevos métodos de predicción de fenómenos a través del procesado de imágenes; mejora de la inyección intratecal, esto es administración de fármacos directamente en el SSS para evitar la barrera hematoencefálica, consiguiendo una aplicación más efectiva de dicho medicamento; y diseño de nuevos dispositivos biomédicos.

1.4. Objetivos del trabajo

Como se ha mencionado, el objetivo principal del trabajo es generar las condiciones de contorno adecuadas para simular el movimiento de las paredes flexibles del canal espinal, provocado por la entrada y salida periódica del flujo de líquido cefalorraquídeo en el canal espinal. Para ello se utilizarán los softwares ANSYS Fluent para realizar las simulaciones, CFD-Post para el post procesado de los resultados, junto con Matlab, el cual será utilizado tanto para el post procesado como para implementar el código que nos permitirá obtener el movimiento deseado de las paredes del canal espinal. Por otro lado, para realizar el mallado de la geometría se utilizará ANSYS Meshing.

Debido a la complejidad del caso, se tendrá en cuenta una serie de simplificaciones del problema, como se comentará en las siguientes secciones. Del mismo modo, se comenzará con simulaciones con geometrías simplificadas que faciliten la comprensión de los fenómenos que tienen lugar en dicho caso, y posteriormente se realizarán simulaciones sobre una geometría realista la cual ha sido ‘suavizada’ para reducir el tiempo de cálculo.

La metodología a seguir a lo largo del trabajo será:

- Realizar un estudio bibliográfico con el objetivo de recopilar la información necesaria para poder abordar el problema.

- Obtención de las geometrías y dominios computacionales simplificados, para casos en los que la piamadre y la duramadre serán concéntricas y otros en los que aparecerá una excentricidad.
- Generar los códigos de las UDF necesarias para cada una de las simulaciones, tanto para un caudal senoidal puro como para un caudal realista.
- Generar el dominio computacional partiendo de la geometría de la médula espinal proporcionada por [citar fuente].
- Simular los modelos simplificados, con las diferentes UDFs, con el fin de estudiar la velocidad estacionaria de las partículas frente a la oscilatoria, lo cual ayudará a la comprensión de la difusión intratecal de fármacos en la médula.
- Simular el caso realista con las diferentes UDFs y obtener una comparación entre un caso canónico (caudal realista) y un caso real.
- Generar el código necesario para realizar de manera correcta el post-procesado de las simulaciones, para lo que se utilizarán las herramientas CFD-Post y Matlab.

2. MARCO TEÓRICO

En esta sección se introducirán los fundamentos teóricos necesarios para la comprensión del trabajo, así como los diferentes métodos y modelos utilizados durante las simulaciones.

2.1. Ecuaciones que gobiernan la Mecánica de Fluidos.

La mecánica de fluidos está gobernada por las ecuaciones de Navier-Stokes. Para poder hacer uso de dichas ecuaciones es necesario emplear los teoremas de divergencia y de transporte para transformar las ecuaciones de forma integral a forma derivativa o diferencial, las cuales se utilizarán en este trabajo, a excepción de la ecuación de la energía, para implementarlas con el Método de los Volúmenes Finitos.

Las leyes de la mecánica de aplicadas a volúmenes fluidos se obtienen directamente al aplicar las ecuaciones de conservación de la masa, cantidad de movimiento y primer principio de la termodinámica a un volumen fluido.

La realidad es que, los ingenieros no tienen toda la información necesaria a la hora de abordar los problemas, es decir, conocen tan solo la cantidad de fluido que pasa por una cierta región o la energía necesaria para mover una masa fluida, y no las fuerzas que actúan sobre él o la cantidad de energía que conlleva dicho movimiento. Por ello, se transformarán las ecuaciones comentadas para poder aplicarlas sobre volúmenes fluidos, y esto se realizará aplicando el Teorema de Transporte de Reynolds.

Para hablar de las ecuaciones de Navier-Stokes se introducirá el término de flujo convectivo $\Phi(x,t)$ de una magnitud fluida, esto es, la cantidad de magnitud que atraviesa una determinada superficie en la unidad de tiempo por el movimiento del fluido. De modo que, si $\Phi=1$ se estará representando el volumen de fluido que atraviesa la superficie por unidad de tiempo, si $\Phi=\rho$ representará la cantidad de masa por unidad de tiempo, si $\Phi=\rho v$ representará la cantidad de movimiento, y si $\Phi=\rho(e+v^2/2)$ representará la energía por unidad de volumen.

Así, en una superficie fija Σ_0 , la cantidad de magnitud fluida que cruza dicho elemento de superficie se puede calcular mediante la integral

$$\int_{\Sigma_0} \phi (\vec{v} \cdot \vec{n}) d\sigma \quad \text{Ecuación 2.1}$$

Si dicha superficie Σ_0 es cerrada entonces, haciendo uso del teorema de divergencia de Gauss se tiene que el flujo convectivo es

$$\int_{\Sigma_0} \phi (\vec{v} \cdot \vec{n}) d\sigma = \int_{V_0} \nabla \cdot (\phi \vec{v}) dV \quad \text{Ecuación 2.2}$$

Si suponemos dicha superficie móvil y que todos sus puntos se mueven con velocidad $\vec{v}(\vec{x}, t)$, la velocidad se reemplazará por la velocidad relativa a la superficie $\vec{v} - \vec{v}_c$ y por tanto la Ecuación 2.1 quedará como

$$\int_{\Sigma_0} \phi [(\vec{v} - \vec{v}_c) \cdot \vec{n}] d\sigma \quad \text{Ecuación 2.3}$$

Observándose que no existiría flujo convectivo en el caso en el cual el fluido se mueve a la misma velocidad que la superficie móvil.

2.1.1. Teorema de Transporte de Reynolds

Al evaluar la derivada temporal de una cierta cantidad de magnitud fluida extensiva contenida en un volumen fluido $V_f(t)$ limitado por la superficie $\Sigma_f(t)$ tanto en el dominio de integración como en el tiempo ha de tenerse en cuenta la regla de Leibniz, obteniéndose

$$\frac{d}{dt} \left[\int_{V_f(t)} \phi(\vec{x}, t) dV \right] = \int_{V_f(t)} \frac{\partial \phi(\vec{x}, t)}{\partial t} dV + \int_{\Sigma_f(t)} \phi(\vec{x}, t) [\vec{v}(\vec{x}, t) \cdot \vec{n}] d\sigma \quad \text{Ecuación 2.4}$$

La resolución de los problemas fluidos consisten pues, en un análisis de una región de interés, $V_c(t)$, la que denominaremos como volumen de control, que por lo

general, estará delimitado por una superficie de control, $\Sigma_c(t)$, cuyos puntos se mueven con una velocidad $v_c(x,t)$.

Para dicho análisis se considerará que en el instante considerado, $V_c(t)$ y $\Sigma_c(t)$ coincidirán con $V_f(t)$ y $\Sigma_f(t)$, obteniendo de la ecuación 2.4. la expresión del Teorema de Transporte de Reynolds

$$\frac{d}{dt} \left[\int_{V_f(t)} \phi(\vec{x}, t) dV \right] = \int_{V_f(t)} \frac{\partial \phi(\vec{x}, t)}{\partial t} dV + \int_{\Sigma_f(t)} \phi(\vec{x}, t) [(\vec{v} - \vec{v}_c) \cdot \vec{n}] d\sigma \quad \text{Ecuación 2.5}$$

que relaciona la variación temporal de una magnitud fluida, en un volumen fluido, con la suma de la variación temporal de dicha magnitud fluida en el volumen de control y el flujo convectivo a través de la superficie que limita dicho volumen de control.

2.1.2. Ecuación de continuidad

Ésta se deriva de la ecuación de conservación de la masa, y por tanto expresa la no variación de un volumen fluido con el tiempo. Si a dicha ecuación se le aplica el Teorema de Transporte de Reynolds (Ecuación 2.5), se puede expresar dicha ecuación, para un volumen de control, de la siguiente forma:

$$\frac{d}{dt} \left[\int_{V_f(t)} \rho dV \right] = \frac{d}{dt} \left[\int_{V_c(t)} \rho dV \right] + \int_{\Sigma_c(t)} \rho [(\vec{v} - \vec{v}_c) \cdot \vec{n}] d\sigma = 0 \quad \text{Ecuación 2.6}$$

Para líquidos perfectos (ρ =constantes), como el líquido objeto de éste estudio, la Ecuación 2.6 se simplifica, quedando de la siguiente forma:

$$\frac{d}{dt} \left[\int_{V_f(t)} dV \right] = \frac{d}{dt} \left[\int_{V_c(t)} dV \right] + \int_{\Sigma_c(t)} [(\vec{v} - \vec{v}_c) \cdot \vec{n}] d\sigma = 0 \quad \text{Ecuación 2.7}$$

Indicando que el volumen fluido permanece constante durante todo el movimiento.

Si se aplica el Teorema de Divergencia de Gauss a la Ecuación 2.7 puede transformarse la integral de superficie en integral de volumen, de manera que se podría obtener un sistema tridimensional infinitesimal, el cual nos será útil a la hora de aplicar el Método de Volúmenes Finitos.

$$\nabla \cdot \vec{v} = 0$$

Ecuación 2.8

2.1.3. Ecuación de cantidad de movimiento

En este caso, a dicha ecuación se llega al reescribir la segunda Ley de Newton

$$\frac{d}{dt} \left[\int_{V_f(t)} \rho \vec{v} dV \right] = - \int_{\Sigma_f(t)} p \vec{n} d\sigma + \int_{\Sigma_f(t)} \vec{\tau} \cdot \vec{n} d\sigma + \int_{V_f(t)} \rho \vec{f}_m dV$$

Ecuación 2.9

Aplicando el Teorema de Transporte de Reynolds se puede reescribir la Ecuación 2.9, obteniendo

$$\begin{aligned} \frac{d}{dt} \left[\int_{V_c(t)} \rho \vec{v} dV \right] + \int_{\Sigma_c(t)} \rho \vec{v} [(\vec{v} - \vec{v}_c) \cdot \vec{n}] d\sigma \\ = - \int_{\Sigma_c(t)} p \vec{n} d\sigma + \int_{\Sigma_c(t)} \vec{\tau} \cdot \vec{n} d\sigma + \int_{V_c(t)} \rho \vec{f}_m dV \end{aligned}$$

Ecuación 2.10

En la Ecuación 2.10 se expresa el incremento de la cantidad de movimiento del volumen de control como la suma del flujo convectivo de cantidad de movimiento que entra a éste por sus paredes, la resultante de las fuerzas de presión, la resultante de las fuerzas viscosas y la resultante de las fuerzas másicas, en forma vectorial.

Del mismo modo que en el caso de la ecuación de continuidad, aplicando el Teorema de Divergencia de Gauss podemos transformar la expresión a su forma diferencial, obteniendo

$$\frac{\partial}{\partial t} (\rho \vec{v}) + \nabla \cdot (\rho \vec{v} \vec{v}) = -\nabla p + \nabla \cdot \vec{\tau}' + \rho \vec{f}_m$$

Ecuación 2.11

Si ahora consideramos despreciables las fuerzas másicas que actúan sobre nuestro volumen de control y se consideran los términos viscosos en la forma $\nabla \cdot \bar{\tau}' = \mu \nabla^2(\bar{v})$, se tiene la siguiente expresión

$$\frac{\partial}{\partial t}(\rho \bar{v}) + \nabla \cdot (\rho \bar{v} \bar{v}) = -\nabla p + \mu \nabla^2(\bar{v}) \quad \text{Ecuación 2.12}$$

2.2. Introducción a la Mecánica de Fluidos Computacional (CFD).

La Dinámica de Fluidos Computacional (CFD, 'Computational Fluid Dynamics') es el área de la mecánica de fluidos que trata sobre la simulación numérica de flujos fluidos, principalmente. El origen de ésta disciplina tiene lugar en la combinación del cálculo numérico y la mecánica de fluidos.

Como se ha comentado en este capítulo, las ecuaciones que rigen el comportamiento de los fluidos son las ecuaciones de Navier-Stokes y pueden ser resueltas por medio de diferentes métodos numéricos, que discretizan éstas y permiten implementarlas en algoritmos que proporcionen soluciones aproximadas a los problemas fluidos.

Los métodos de discretización más conocidos son el Método de Diferencias Finitas, el Método de Elementos Finitos y el Método de Volúmenes Finitos, siendo éste último el que está implementado en el software de cálculo utilizado para éste trabajo. Dichos métodos intercambian el dominio continuo por uno discreto, también conocido como malla computacional, que es un conjunto de volúmenes de control utilizado para representar el dominio original y que permite resolver las ecuaciones en cada uno de dichos volúmenes.

Para garantizar que estas mallas sean capaces de identificar correctamente la física del modelo, la cantidad de volúmenes de control utilizados en la misma ha de ser estudiado previamente a la simulación, en un estudio de convergencia de malla.

Una vez generada la malla se pasa a la configuración del modelo, donde se han de determinar las condiciones de contorno del problema y el 'solver' a utilizar, así como los esquemas de discretización de las ecuaciones del problema. Con estos datos se comienza un proceso iterativo hasta que la solución converge, y ésta está disponible para su post-procesado.

Finalmente, en la etapa de post-procesado es cuando se evalúan los resultados por medio de herramientas como vectores, líneas de corriente, contornos, planos de corte, etc.

Es por esto que el CFD está siendo ampliamente utilizado en diferentes ramas de la industria, desde la fase conceptual de un proyecto hasta la etapa de producción. Algunos ejemplos de aplicación son:

- Analizar la aerodinámica de vehículos.
- Simular hidrodinámica y hemodinámica.
- Desarrollar sistemas de generación de energía.
- Realizar estudios de caracterización de polución ambiental, dispersión de gases, etc.

2.3. Método de las diferencias finitas.

Como se ha comentado, la finalidad del CFD es remplazar problemas de dominio continuo por problemas de dominio discreto usando mallas computacionales que nos permitan tener un mayor control sobre los parámetros que influyen en dicho problema.

El primer paso para obtener una simulación numérica será discretizar la geometría del dominio del problema, generando una malla computacional. En diferencias finitas (FD), por lo general se utilizan mallas estructuradas donde cada nodo de la misma se considera el origen de un sistema de coordenadas local, con ejes que coindicen con las direcciones de los bordes de la malla.

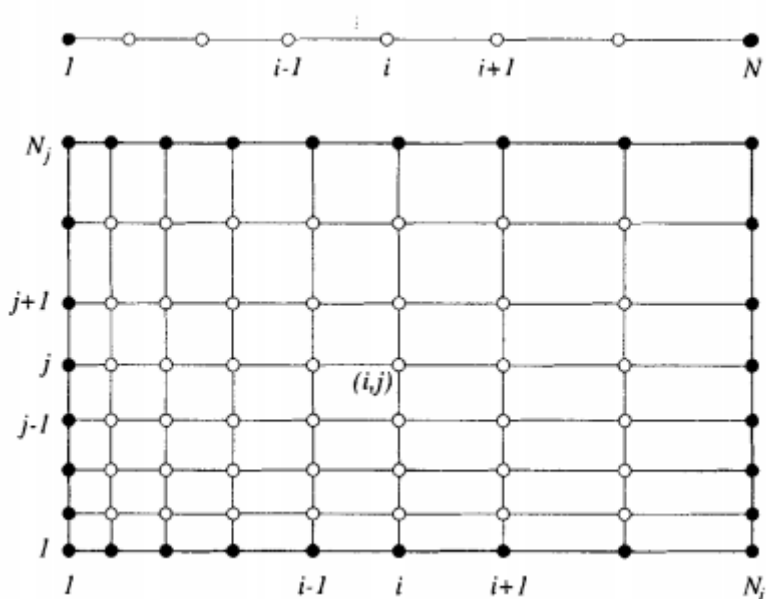


Figura 2.1. Ejemplo de mallas computacionales 1D y 2D [2]

Como se puede observar en la figura, cada nodo queda identificado por unos subíndices ((i,j) en 2D o (i,j,k) en 3D) y los nodos vecinos presentarán dichos subíndices con un incremento unidad en éstos.

El método de las diferencias finitas tiene como fin obtener una aproximación de la derivada de una función escalar, por ejemplo, $\phi(x)$ evaluada en un punto x_0 , haciendo uso del desarrollo en series de Taylor.

$$\phi(x_0 + \Delta x) = \phi(x_0) + \Delta x \frac{\partial \phi}{\partial x} \Big|_{x_0} + \frac{\Delta x^2}{2} \frac{\partial^2 \phi}{\partial x^2} \Big|_{x_0} + \dots \quad \text{Ecuación 2.13}$$

La primera derivada de $\phi(x)$ podrá aproximarse por:

$$\frac{\partial \phi}{\partial x} = \lim_{\Delta x \rightarrow 0} \frac{\phi(x_0 + \Delta x) - \phi(x_0)}{\Delta x} + O(\Delta x) \quad \text{Ecuación 2.14}$$

El primer término será el término utilizado para discretizar o correspondiente al término de primer orden del desarrollo en series de Taylor y el segundo será el correspondiente a los términos de orden superior.

En la figura se puede ver una interpretación geométrica de la aproximación anterior. Puede observarse como la línea de puntos representa la derivada entre dos puntos x_i y x_{i+1} , siendo ésta la derivada hacia adelante, la línea discontinua entre dos puntos x_{i-1} y x_i , o derivada hacia atrás, y la línea continua sería la derivada entre dos puntos x_{i-1} y x_{i+1} , o derivada centrada.

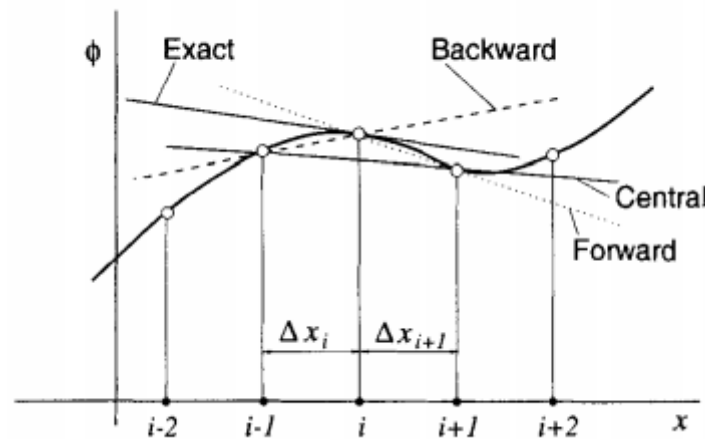


Figura 2.2. Representación geométrica de la derivada [2]

Parece lógico pensar que algunas aproximaciones son mejores que otras ya que las pendientes de las diferentes derivadas son diferentes, siendo la derivada centrada la que proporciona la pendiente más próxima a la exacta. Obviamente, si la distancia entre los puntos en los que se evalúa la derivada es cada vez menor, la calidad de la aproximación aumentará, como ocurre en las mallas refinadas al introducir nodos adicionales.

Al desconocer el valor de los términos de orden superior de las expresiones de la derivada, éstas no presentan utilidad como se han mostrado. Sin embargo, como ocurre en la mayoría de las situaciones, las distancias entre los nodos son pequeñas de modo que los términos de orden superior toman valores muy pequeños en comparación con los términos de primer orden y por tanto, podremos considerarlos despreciables y de esta manera las expresiones para cada una de las derivadas anteriores será:

- Derivada centrada:

$$\left(\frac{\partial \phi}{\partial x}\right)_i \cong \frac{\phi_{i+1} - \phi_{i-1}}{x_{i+1} - x_{i-1}} \quad \text{Ecuación 2.15}$$

- Derivada hacia adelante:

$$\left(\frac{\partial \phi}{\partial x}\right)_i \cong \frac{\phi_{i+1} - \phi_i}{x_{i+1} - x_i} \quad \text{Ecuación 2.16}$$

- Derivada hacia atrás:

$$\left(\frac{\partial \phi}{\partial x}\right)_i \cong \frac{\phi_i - \phi_{i-1}}{x_i - x_{i-1}} \quad \text{Ecuación 2.17}$$

Los términos que se han despreciado conforman el error de truncamiento y miden la precisión de la aproximación de la derivada. Este error será la suma de los términos de orden superior multiplicados por la distancia entre los nodos en los que se evalúa la derivada, Δx .

$$\epsilon_t = (\Delta x)^m \alpha_{m+1} + (\Delta x)^{m+1} \alpha_{m+2} + \dots + (\Delta x)^n \alpha_{n+1} \quad \text{Ecuación 2.18}$$

Puede observarse que, si Δx se reduce, el error de truncamiento será menor y por tanto, las derivadas anteriores convergerán con un error de $(\Delta x)^m$, siendo m el exponente del primer término del error de truncamiento.

Del mismo modo, podemos obtener aproximaciones para derivadas de orden superior:

$$\begin{aligned} \left(\frac{\partial^2 \phi}{\partial x^2}\right)_i &\cong \frac{\left(\frac{\partial \phi}{\partial x}\right)_{i+1} - \left(\frac{\partial \phi}{\partial x}\right)_i}{(x_{i+1} - x_i)} \\ &\cong \frac{\phi_{i+1}(x_i - x_{i-1}) - \phi_i(x_{i+1} - x_{i-1}) + \phi_{i-1}(x_i - x_{i-1})}{(x_{i+1} - x_i)^2(x_i - x_{i-1})} \end{aligned} \quad \text{Ecuación 2.19}$$

Para el caso de mallas con una separación constante entre nodos, la expresión anterior se simplifica

$$\left(\frac{\partial^2 \phi}{\partial x^2}\right)_i \cong \frac{\phi_{i+1} - 2\phi_i + \phi_{i-1}}{\Delta x^2} \quad \text{Ecuación 2.20}$$

Finalmente, si se desea discretizar una ecuación diferencial en dos variables o mixta bastará con realizar el mismo proceso en la dirección en la que se desee obtener la aproximación de la derivada:

$$\left(\frac{\partial^2 \phi}{\partial x \partial y}\right) \cong \frac{\partial}{\partial y} \left(\frac{\phi_{i+1,j} - \phi_{i,j}}{\Delta x} \right) \cong \frac{\phi_{i+1,j+1} - \phi_{i+1,j} - \phi_{i,j} + \phi_{i,j-1}}{\Delta x \Delta y} \quad \text{Ecuación 2.21}$$

Una vez discretizadas las ecuaciones se obtiene un sistema de ecuaciones algebraico, ya que la ecuación se particularizará para cada uno de los nodos de la malla. Para la función de la ecuación (anterior) de los ejemplos anteriores, el sistema obtenido tendrá la forma:

$$K_L \phi_L + \sum K_N \phi_N = Q_L \quad \text{Ecuación 2.22}$$

Donde el índice L denota la posición del nodo en el que se está evaluando la derivada y N la de los nodos vecinos que aparecen en la aproximación de la derivada para cada L. Por otro lado, el coeficiente K dependerá de la geometría del dominio, de las propiedades del fluido y de la ecuación que gobierne el problema, mientras que Q será el término independiente, en un principio conocido.

$$\begin{bmatrix} K_{1,1} & K_{1,2} & 0 \\ K_{2,1} & K_{2,2} & K_{2,3} \\ 0 & K_{3,2} & K_{3,3} \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix} = \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \end{bmatrix}$$

Figura 2.3. Estructura de una matriz para una malla computacional de 3 puntos.

2.4. Método de Volúmenes Finitos.

2.4.1. Ecuación de transporte escalar

A parte de los términos convectivos, pueden existir términos fuentes o sumideros locales de Φ , pudiendo ser fuentes superficiales o volumétricas, de manera que las ecuaciones de conservación quedarían como:

$$\frac{d}{dt} \left[\int_{V_c(t)} \phi(\bar{x}, t) dV \right] + \int_{\Sigma_c(t)} \phi(\bar{x}, t) (\bar{v} - \bar{v}_c) \cdot \bar{n} d\sigma = \int q_v dV - \int q_s \cdot \bar{n} d\sigma \quad \text{Ecuación 2.23}$$

Que en forma diferencial se expresa como:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi \bar{v}) = q_v - \nabla \cdot \bar{q}_s \quad \text{Ecuación 2.24}$$

Puede suponerse que las fuerzas superficiales son de carácter difusivo y por tanto, la ecuación anterior puede reescribirse de la siguiente forma:

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\phi \bar{v}) - \nabla \cdot (k \nabla \phi) = q_v \quad \text{Ecuación 2.25}$$

Los términos fuente son los más problemáticos.

2.4.2. Celdilla computacional y nomenclatura.

A diferencia del método de las diferencias finitas, donde las celdillas computacionales eran planas, bien en 1 o 2 dimensiones, en el método de los volúmenes finitos las celdillas computacionales son tridimensionales.

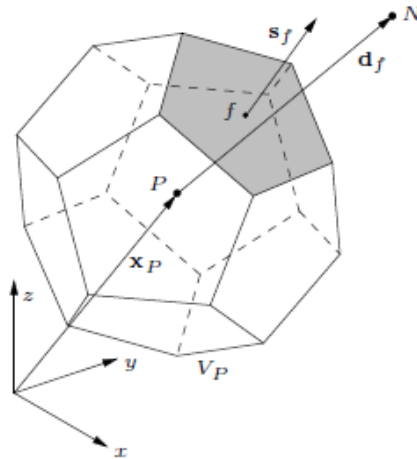


Figura 2.4. Representación de una celda computacional [3]

Puede observarse ver que ésta celda computacional es convexa y su contorno está formado por un conjunto de polígonos, también convexos.

El centroide de la misma se denota con $\vec{x}_p$ y el volumen de la misma con V_p y por definición se tiene que:

$$\int_{V_p} (\vec{x} - \vec{x}_p) dV = 0 \quad \text{Ecuación 2.26}$$

Del mismo modo, se denotará con el subíndice N a las celdillas vecinas o colindantes. De éste modo, el vector de posición que une a los puntos P y N, y que está asociado a la cara f será:

$$\vec{d}_f = \overline{PN} \quad \text{Ecuación 2.27}$$

Y debe cumplirse que, en el centro de cada cara f, la regla del centroide:

$$\int_{S_f} (\vec{x} - \vec{x}_p) dS = 0 \quad \text{Ecuación 2.28}$$

El vector de superficie $\vec{s}_f$ es normal a la cara y su magnitud es igual al área de la cara. Puesto que las caras no tienen que definir un plano, el vector $\vec{s}_f$ se calcula como una integral de superficie.

$$\vec{s}_f = \int_{S_f} \vec{n} dS \quad \text{Ecuación 2.29}$$

Las caras pueden ser: interiores si separan dos celdas o de contorno si son adyacentes a una celda y la normal de la misma apunta al exterior del dominio computacional

En la práctica, las operaciones de cálculo en las caras y volúmenes se realizan descomponiendo éstas en subregiones, de manera que el método puede admitir mallas polihédricas.

$$\int_V \frac{\partial \phi}{\partial t} dV + \oint_S \phi (\vec{n} \cdot \vec{v}) dS - \oint_S k (\vec{n} \cdot \nabla \phi) dS = \int_V q_v dV \quad \text{Ecuación 2.30}$$

La ecuación X pasa a ser la ecuación Y y ahora S es una superficie cerrada ya que se han transformado las integrales de volumen en integrales de superficie por el Teorema de Gauss.

2.4.3. Discretización espacial y temporal.

Se toma una variación espacial de Φ de tipo lineal que proporciona un esquema de discretización de segundo orden.

$$\phi(\vec{x}) = \phi_P + (\vec{x} - \vec{x}_P) \cdot (\nabla \phi)_P \quad \text{Ecuación 2.31}$$

Expresión que se aplica de manera individual a cada una de las celdillas que forman el dominio computacional, siendo $\phi_P \equiv \phi(\vec{x}_P)$.

Del mismo modo, también se tomará una variación lineal para el tiempo, proporcionando ésta un esquema de discretización de segundo orden

$$\phi(t + \Delta t) = \phi^t + \Delta t \left(\frac{\partial \phi}{\partial t} \right)^t \quad \text{Ecuación 2.32}$$

donde $\phi^t \equiv \phi(t)$.

2.4.4. Evaluación de las integrales de volumen:

De acuerdo a lo comentado en las secciones anteriores, si se toma término a término la ecuación de transporte escalar se tiene.

$$\begin{aligned} \int_V \phi \, dV &= \int_V [\phi_P + (\vec{x} - \vec{x}_P) \cdot (\nabla \phi)_P] \, dV = \phi_P \int_V dV + (\nabla \phi)_P \cdot \int_V (\vec{x} - \vec{x}_P) \, dV \\ &= \phi_P V_P \end{aligned} \quad \text{Ecuación 2.33}$$

Para discretizar las integrales de superficie se descompondrá la integral de superficie como suma de las integrales sobre las caras de la celdilla, evaluándose todas de la misma forma.

$$\oint_S \vec{n} \phi \, dS = \sum_f \int_{S_f} \vec{n} \phi_f \, dS_f = \sum_f \int_{S_f} \vec{n} [\phi_f + (\vec{x} - \vec{x}_P) \cdot (\nabla \phi)_f] \, dS_f = \sum_f \vec{s}_f \phi_f \quad \text{Ecuación 2.34}$$

Así se observan las implicaciones de asumir una variación lineal para espacio y tiempo y de haber seleccionado P como centroide de la celda, así como que f sea el centroide de la cara.

Del mismo modo que en el método de las diferencias finitas, tomaremos la división del espacio como celdas y la división del tiempo como pasos de tiempo y asumiendo una descripción discreta de una variable de campo continua. Como se ha comentado, tanto la variación espacial como la temporal son de segundo orden, y se generan expresiones para evaluar las integrales tanto de superficie como de volumen [ecuaciones anteriores.]

Las herramientas que se utilizarán para ensamblar los operadores de las diferenciales que aparecen en la ecuación de transporte escalar son:

- Variación temporal.

- Operador gradiente.
- Operador convectivo.
- Operador difusivo.
- Términos fuentes y sumideros.

2.4.4.1. Derivada primera en tiempo.

La derivada en tiempo modela el cambio de Φ con el tiempo. En la ecuación de transporte escalar solo se tiene una integral de volumen en la que aparece dicha derivada.

Como se ha comentado, la división de tiempo se dividirá en pasos de tiempo, que se denotarán con Δt , de manera que $t_{new} = t_{old} + \Delta t$, y por tanto, al definir el valor de Φ en el nuevo instante de tiempo tendremos

$$\phi^n = \phi(t = t_{new}) \quad \text{Ecuación 2.35}$$

$$\phi^o = \phi(t = t_{old}) \quad \text{Ecuación 2.36}$$

De manera que, si aproximamos la derivada primera en primer o segundo orden se tiene:

$$\frac{\partial \phi}{\partial t} = \frac{\phi^n - \phi^o}{\Delta t}, \quad (\text{Primer orden}) \quad \text{Ecuación 2.37}$$

$$\frac{\partial \phi}{\partial t} = \frac{\frac{3}{2}\phi^n - 2\phi^o + \frac{1}{2}\phi^{oo}}{\Delta t}, \quad (\text{Segundo orden}) \quad \text{Ecuación 2.38}$$

Por tanto, integral de volumen queda:

$$\int_V \frac{\partial \phi}{\partial t} dV = \frac{\phi^n - \phi^o}{\Delta t} V_P \quad \text{Ecuación 2.39}$$

$$\int_V \frac{\partial \phi}{\partial t} dV = \frac{\frac{3}{2}\phi^n - 2\phi^o + \frac{1}{2}\phi^{oo}}{\Delta t} V_P \quad \text{Ecuación 2.40}$$

2.4.4.2. Discretización del operador gradiente.

La integral del operador gradiente en un volumen se evalúa con el Teorema de Gauss.

$$\int_{V_P} \nabla \phi \, dV = \oint_{\partial V_P} d\vec{S} \phi \quad \text{Ecuación 2.41}$$

Que al descomponerse en la suma de ϕ_f en todas las caras:

$$\oint_S \vec{n} \phi \, dS = \sum_f \vec{s}_f \phi_f \quad \text{Ecuación 2.42}$$

Donde $f_x = \overline{fN}/\overline{PN}$ y

$$\phi_f = f_x \phi_P + (1 - f_x) \phi_N \quad \text{Ecuación 2.43}$$

2.4.4.3. Discretización del operador convectivo.

El término convectivo modela el transporte advectivo ya que la velocidad no es nula, y para evaluar dicha integral debe recurrirse de nuevo al Teorema de Gauss.

$$\int_V \nabla \cdot (\phi \vec{v}) \, dV = \oint_S \phi (\vec{n} \cdot \vec{v}) \, dS \quad \text{Ecuación 2.44}$$

La integral de superficie se vuelve a evaluar como la suma en las caras

$$\oint_S \phi (\vec{n} \cdot \vec{v}) \, dS = \sum_f \phi_f (\vec{s}_f \cdot \vec{v}) = \sum_f \phi_f F \quad \text{Ecuación 2.45}$$

donde $F = \vec{s}_f \cdot \vec{v}_f$, es el flujo a través de la cara y mide la cantidad de la variable Φ transportada por el fluido al cruzar la cara. Para evaluar ϕ_f se interpolará en las caras a partir de los valores de ϕ_P y ϕ_N con la ecuación Ecuación 2.43.

2.4.4.4. Discretización del operador difusivo.

Este término informa del transporte originado por las variaciones espaciales de ϕ , y del mismo modo que se ha realizado con el resto de operadores, al éste ser una integral de superficie se expresará como suma en las caras.

$$\oint_S k(\vec{n} \cdot \nabla \phi) dS = \sum_f \int_{S_f} k(\vec{n} \cdot \nabla \phi) dS = \sum_f k_f \vec{s}_f \cdot (\nabla \phi)_f \quad \text{Ecuación 2.46}$$

Donde k_f se evalúa mediante diferencias centradas y la evaluación de la componente de gradiente normal a la cara queda como

$$\vec{s}_f \cdot (\nabla \phi)_f = |\vec{s}_f| \frac{\phi_N - \phi_P}{|\vec{d}_f|} \quad \text{Ecuación 2.47}$$

La ecuación anterior es válida si $\vec{s}_f$ y $\vec{d}_f$ son paralelos, en caso contrario, ésta será la contribución en la dirección normal a la cara.

Finalmente, si es un caso no ortogonal, debe utilizarse un término de corrección, de manera que la contribución de la cara f a la matriz ensamblada es

- Valor diagonal: $a_P = -k_f \frac{|\vec{s}_f|}{|\vec{d}_f|}$
- Valor fuera de la diagonal: $a_N = k_f \frac{|\vec{s}_f|}{|\vec{d}_f|}$

2.4.4.5. Discretización de los términos fuente y sumidero.

Los términos fuente y sumidero tienen carácter local y en general son función del espacio y del tiempo, de si mismo, o de otras variables, por lo que pueden llegar a ser términos muy complejos. Éste término modela la interacción entre diferentes elementos del sistema.

Normalmente, el término q_v se hace lineal respecto a ϕ para garantizar la estabilidad y la convergencia del método. De éste modo se tiene:

$$q_v(\phi) = q_u + q_d\phi \quad \text{Ecuación 2.48}$$

Donde $q_d = \frac{\partial q_v(\phi)}{\partial \phi} < 0$ para sumideros.

Estos términos no dependen de nodos vecinos y en la matriz se crea una entrada diagonal para los términos q_d y una contribución explícita al término fuente, q_u .

2.5. Simulaciones FSI.

El objeto de estudio de éste trabajo la interacción entre el flujo de líquido cefalorraquídeo y el movimiento de la pared de la dura madre, es decir, una simulación con interacción fluido-estructura (FSI, 'Fluid Structure Interaction'). Este tipo de problemas son muy frecuentes en ingeniería ya que se encuentran presentes desde en problemas aeroespaciales hasta problemas de biomecánica ya que permite conseguir mejores diseños.

En éste tipo de problemas, aparte de las ecuaciones que gobiernan el movimiento fluido se han de resolver las ecuaciones que modelan el comportamiento estructural, es decir, son problemas donde tenemos dos dominios computacionales diferentes, el sólido que se tratará con el Método de los Elementos Finitos, y el fluido que se tratará con el Método de Volúmenes Finitos, comentado anteriormente. Por tanto, al tener dos problemas diferentes se hará necesario realizar un acoplamiento entre las soluciones de cada uno de los casos. En función de cómo de independientes sean un campo de otro, la solución debe ser acoplada de un modo u otro, como puede verse en la **Figura 2.5**.

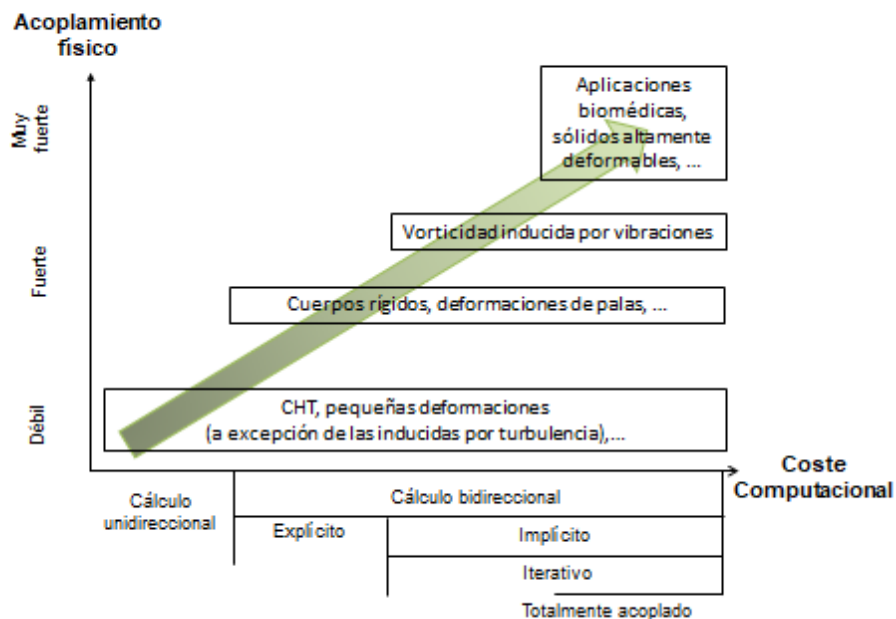


Figura 2.5. Esquema de los modelos FSI

Las diferencias entre los métodos unidireccional y bidireccional son que en el primer caso, la interacción entre el sólido y el fluido a través de la superficie mutua es tratada por separado, obteniendo una solución menos estable y precisa, usada en los casos en los que la dependencia entre un campo y otro es débil, a diferencia del método bidireccional que trata ambos mediante un modelo matemático y esquema de discretización unificado, lo cual proporciona mayor estabilidad y precisión pero es muy costoso computacionalmente.

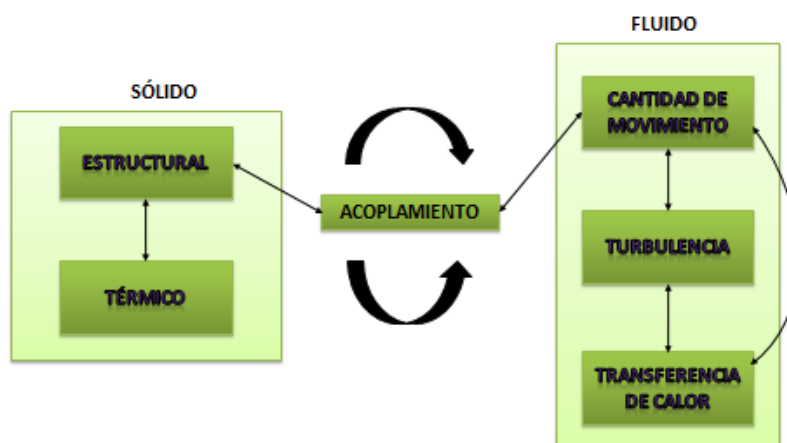


Figura 2.6. Esquema del acoplamiento entre campos sólidos y fluidos en modelos FSI.

Teniendo en cuenta lo que muestra la Figura 2.6, para el objeto de estudio de este trabajo, para unas condiciones de contorno para el CSF se obtendría un campo de presiones en las paredes de la duramadre, lo cual serviría para calcular las deformaciones de la misma, que a su vez, afectaría al flujo de CSF en el canal espinal.

En este trabajo, se presentan como alternativa al FSI los métodos de malla dinámica debido al alto coste computacional que requieren estos métodos, de manera que en lugar de calcular las deformaciones de la duramadre mediante FSI, impondremos estas conociendo cuál es el flujo volumétrico que debe atravesar cada sección de la médula espinal en cada instante de tiempo, por tanto solo se deberá resolver el problema fluido. Para ello, como ha sido comentado en la introducción, se utilizarán datos de estudios previos con el fin de obtener una solución lo más precisa y realista posible.

2.6. Métodos de malla dinámica.

En ANSYS Fluent, están disponibles tres métodos de movimiento de malla diferentes para actualizar el volumen de las mallas en las zonas deformables sujetas al movimiento de los contornos:

- Método Smoothing.

- Método dinámico Layering.
- Método Remeshing.

2.6.1. Métodos Smoothing.

2.6.1.1. Método Spring/Laplace.

En éste método los bordes entre nodos de la malla se consideran enlazados de la misma forma que en una red de muelles ideales. La posición inicial o espaciado de los bordes antes de cualquier movimiento constituye el equilibrio de la malla, de manera que cualquier desplazamiento de la malla dará lugar a una fuerza proporcional a dicho desplazamiento a lo largo de los muelles conectados a dicho nodo, como se puede ver en la Figura 2.7. Puede usarse la Ley de Hook para expresar la fuerza que aparece en cada nodo,

$$\vec{F}_i = \sum_j^{n_i} k_{ij}(\Delta\vec{x}_j - \Delta\vec{x}_i) \quad \text{Ecuación 2.49}$$

siendo $\Delta\vec{x}_i$ y $\Delta\vec{x}_j$ los desplazamientos del nodo i y el nodo vecino j, n es el número de nodos vecinos conectados al nodo i, y k_{ij} es la constante elástica entre los nodos i y j, la cual viene definida por

$$k_{ij} = \frac{j}{\sqrt{|\vec{x}_i - \vec{x}_j|}} \quad \text{Ecuación 2.50}$$

En el equilibrio la fuerza neta debida a los muelles conectados a cada nodo debe ser nula, lo cual nos lleva a una ecuación iterativa

$$\Delta\vec{x}_i^{m+1} = \frac{\sum_j^n k_{ij}\Delta\vec{x}_j^m}{\sum_j^n k_{ij}} \quad \text{Ecuación 2.51}$$

Una vez conocidos los desplazamientos en el contorno (tras haber sido actualizadas las posiciones de los nodos de dicho contorno), la ecuación es resuelta por el método de Jacobi en los nodos interiores de la malla

$$\vec{x}_i^{n+1} = \vec{x}_j^n + \Delta \vec{x}_i^{m,converged} \quad \text{Ecuación 2.52}$$

Donde n y $n+1$ denotan las posiciones en el instante actual y el siguiente, respectivamente.

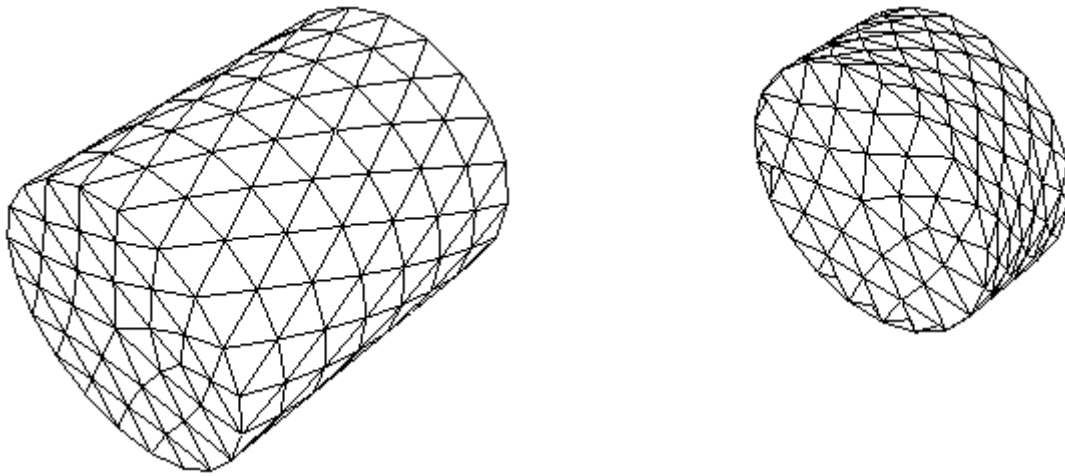


Figura 2.7 Ejemplo de malla deformada con el método 'Smoothing-Spring' [4]

El método Spring puede utilizarse para actualizar cualquier celdilla o cara de la cara a la que se le impone el movimiento.

En el caso de utilizar mallas tetrahédricas se recomienda que el movimiento de los contornos sea unidireccional para cada nodo, siendo dicha dirección recomendada la normal.

2.6.1.2. Método Smoothing basado en la difusión.

En éste método, el movimiento sigue la ecuación de difusión

$$\nabla \cdot (\gamma \vec{\nabla} u) = 0 \quad \text{Ecuación 2.53}$$

En los contornos deformables, las condiciones de contorno deben ser tales que el movimiento sea tangente al contorno.

El coeficiente de difusión γ controlará como el movimiento del contorno afecta al interior de la malla. Si dicho coeficiente es constante el movimiento se difunde de manera uniforme a través de la malla, mientras que un coeficiente no constante hará que los nodos en las zonas de alta difusión tiendan a moverse juntos. Dicho coeficiente, en ANSYS FLUENT se implementa mediante la expresión

$$\gamma = \frac{1}{d^\alpha} \quad \text{Ecuación 2.54}$$

donde d es la distancia normalizada del contorno o distancia normalizada de pared y $\alpha \geq 0$ es un parámetro de entrada.

La ecuación de difusión se discretiza en ANSYS FLUENT mediante el método de volúmenes finitos, obteniendo una matriz resuelta de manera iterativa utilizando el solver 'Algebraic Multigrid' (AMG). La velocidad del desplazamiento será interpolada en los nodos y la posición de los mismos se actualizará de acuerdo a

$$\vec{x}_{new} = \vec{x}_{old} + \vec{u} \Delta t \quad \text{Ecuación 2.55}$$

Este método es una alternativa al método Spring y está disponible para todos los tipos de mallas y elementos.

El método de difusión es más costoso computacionalmente que el método Spring pero proporciona mejores resultados, y del mismo modo que el método Spring, trata mejor movimientos de traslación que de rotación.

2.6.1.3. Método Laplacian.

Este es el método más utilizado en mallas dinámicas en las que se aplican los métodos Smoothing. En este caso la posición de cada vértice de la malla será ajustada por el centro geométrico de los vértices vecinos.

Es un método menos costoso pero no garantiza que se mantenga la calidad de la malla durante el movimiento, para evitar dichos problemas ANSYS FLUENT solo realiza dicho ajuste en la posición de los vértices si se mejora la calidad de la malla.

El cálculo de las posiciones de los nodos se puede calcular mediante

$$\bar{\vec{x}}_i^m = \frac{\sum_j^{n_i} \vec{x}_j^m}{n_i} \quad \text{Ecuación 2.56}$$

donde $\bar{\vec{x}}_i^m$ es la posición media del nodo i en la iteración m, $\vec{x}_j^m$ es la posición del nodo j vecino en la iteración m, n_i es el número de nodos vecinos del nodo i. La nueva posición del nodo $\vec{x}_j^{m+1}$ puede calcularse como

$$\vec{x}_j^{m+1} = \vec{x}_i^m(1 - \beta) + \bar{\vec{x}}_j^m\beta \quad \text{Ecuación 2.57}$$

donde β es el factor de relajación de los nodos del contorno.

2.6.2. Método Layering.

En mallas hexahédricas, puede utilizarse dicho método para destruir o crear nuevas capas de celdillas adyacentes al contorno móvil en base a la altura de la capa adyacente a la móvil. Éste método, en ANSYS FLUENT, permite especificar la altura de la capa para cada contorno móvil.

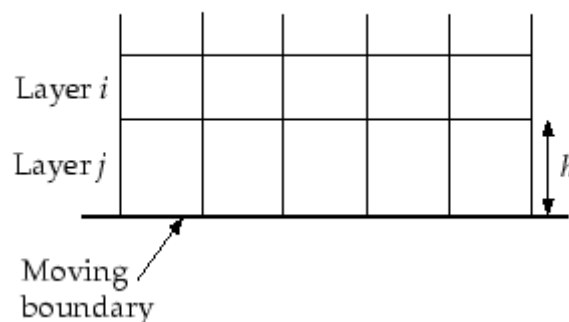


Figura 2.8. Representación de las capas cercanas a un contorno móvil en una malla dinámica. [4]

Si las celdillas de la capa j se expande, la altura hasta la que dichas celdillas pueden crecer será

$$h_{min} > (1 + \alpha_s) h_{ideal} \quad \text{Ecuación 2.58}$$

donde h_{min} es la altura mínima de las celdillas de la capa j , h_{ideal} es la altura ideal de la celdilla, y α_s es el factor de división de capas (layer Split factor). En ANSYS FLUENT es posible definir un valor constante de h_{ideal} o una función que defina la variación de éste.

Si por el contrario las celdillas de la capa j se contraen, la altura mínima hasta la que pueden comprimirse dichas celdillas será

$$h_{min} < \alpha_c h_{ideal} \quad \text{Ecuación 2.59}$$

donde α_c es el factor de colapso de la capa. Cuando se alcanza dicha condición, las capa j se une a la capa i .

2.6.3. Método Remeshing.

Como se ha mencionado anteriormente, en mallas triangulares o tetrahédricas se suele utilizar el método Spring. Cuando los contornos son muy grandes comparados con el tamaño de la celdilla, la calidad de la malla puede verse deteriorada con dicho método, lo que implica una solución poco precisa.

Éste método es más apropiado para realizar movimientos grandes. El funcionamiento es similar al de los dos métodos anteriores con la diferencia de que en el caso de que la malla sufriese grandes distorsiones, las zonas distorsionadas se ‘remallan’ para corregir las mismas.

2.6.4. Selección del método de malla dinámica.

En base a lo comentado para cada uno de los métodos anteriores y las características de nuestro problema, se ha considerado que el mejor método será el

método ‘Smoothing- Spring’ ya que no es un método que requiera una gran potencia de cálculo y no distorsionará en gran medida los elementos, obteniendo resultados relativamente precisos.

2.6.4. Macros para UDFs en simulaciones con métodos de malla dinámica.

ANSYS FLUENT dispone de 6 macros diferentes para definir las UDFs que controlan el comportamiento de las mallas dinámicas. En la siguiente tabla se muestran dichas macros y las funciones de cada una de ellas.

MACRO	FUNCIÓN
DEFINE_CG_MOTION	Movimiento del centro de gravedad
DEFINE_DYNAMIC_ZONE_PROPERTY	Variación de la altura de la capa de celdillas
DEFINE_GRID_MOTION	Movimiento de la malla
DEFINE_GEOM	Deformación de la geometría
DEFINE_SDOF_PROPERTIES	Propiedades para movimientos con 6 grados de libertad
DEFINE_CONTACT	Detección de contacto

Tabla 2.1. Resumen de las funciones de las diferentes macros DEFINE para mallas dinámicas.

Se tratarán las macros **DEFINE_CG_MOTION** y **DEFINE_GRID_MOTION** para determinar cuál de las dos se adapta mejor a las condiciones de nuestro problema.

2.2. DEFINE_CG_MOTION.

Con la macro **DEFINE_CG_MOTION** puede especificarse el movimiento de una determinada zona, proporcionando la velocidad lineal o angular de dicha zona para cada instante de tiempo. Dicha velocidad es utilizada para actualizar la posición de los nodos de la zona para cada instante de tiempo. Dicho movimiento se realizará de manera que todo el contorno se mueve como un sólido rígido, generándose saltos escalones entre las zonas desplazadas y las no desplazadas.

En caso de no conocer la velocidad del desplazamiento de los nodos, si no el valor de dicho desplazamiento se utilizará la macro **DEFINE_GRID_MOTION**.

3.2. DEFINE_GRID_MOTION.

Con esta macro se actualizarán las posiciones de los nodos del contorno dinámico aplicando la ecuación de movimiento del sólido. Esto implica que no existe un movimiento relativo entre los nodos de la zona dinámica, pero sin embargo, puede controlarse el movimiento, de manera independiente, de cada uno de los nodos del contorno en movimiento, por tanto, a diferencia de con la macro `DEFINE_CG_MOTION`, con `DEFINE_GRID_MOTION` se puede obtener un movimiento continuo sobre una superficie en la que cada uno de los nodos tiene una amplitud de desplazamiento diferente.

Es por esto por lo que se ha decidido a utilizar `DEFINE_GRID_MOTION` para generar las UDFs de las simulaciones de interés ya que podremos conseguir un movimiento más realista de las paredes de la médula y que, a diferencia de lo que sucedería con `DEFINE_CG_MOTION`, no deben definirse interfaces entre las diferentes zonas de movimiento.

En estudios previos se han utilizado mallas donde se realizaba una segmentación a lo largo de la médula y en cada una de las secciones transversales, de manera que para cada uno de esos segmentos se define una función diferente, así como la interfaz entre los diferentes segmentos, como se puede observar en la Figura 2.9. En este trabajo se generará un movimiento continuo en función de un caudal de CSF experimental obtenido de dichos estudios, como se comentará en las siguientes secciones, de manera que no se deberán segmentar las mallas y por tanto, no deberán generarse dichas interfaces, lo cual simplificará el trabajo.

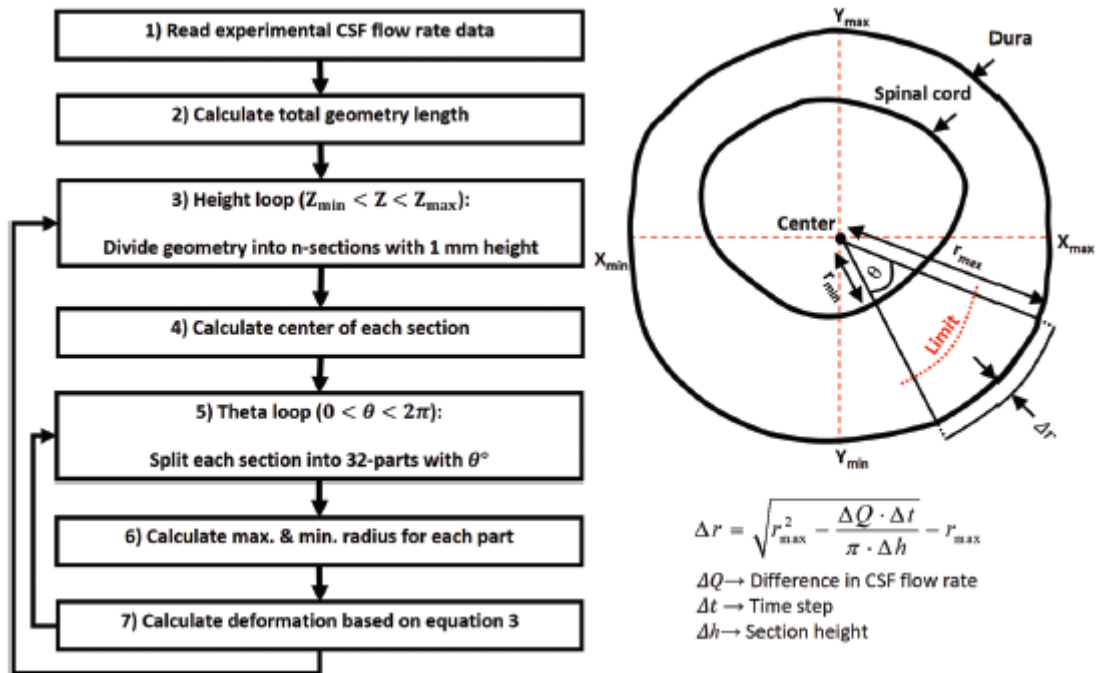


Figura 2.9 Metodología utilizada en [5] para reproducir un caudal real en una geometría compleja mediante simulaciones con malla móvil.

3. DEFINICIÓN Y OBTENCION DE LOS DOMINIOS COMPUTACIONALES.

Como se ha comentado a lo largo de la introducción, el objetivo final del trabajo es el de realizar diferentes simulaciones en una malla con geometría realista. Antes de realizar dichas simulaciones se realizarán unas simulaciones previas ésta con el fin de obtener una serie de resultados relacionados con dicha simulación, por tanto, debemos generar una geometría simplificada con dimensiones aproximadamente iguales a la de la geometría realista con el fin de que los resultados sean lo más próximos posible.

3.1 Geometría Simplificada.

La geometría simplificada será una geometría compuesta por dos cilindros de igual longitud y cuyos valores de diámetro serán los diámetros equivalentes a los de la duramadre y piamadre en el foramen magnum, con el fin de que dicha geometría sea lo más próxima a la realidad posible. Los cilindros están dispuestos de manera que, en función del caso, presentarán o no cierta excentricidad uno con respecto a otro.

Para generar la geometría, las dimensiones de la misma deben ser tales que cumplan la condición:

$$L \gg R_{dura} \gg e$$

Ecuación 3.1

La geometría simplificada está compuesta por dos cilindros, concéntricos o con una cierta excentricidad, e , en función del caso a tratar, con una longitud igual a la longitud del modelo de la médula. El valor de los diámetros será tal que nos proporcione el valor de perímetro igual a los perímetros de la dura y pia madre, respectivamente.

Para generar estas geometrías se ha utilizado el bloque 'Design Modeler' de ANSYS.

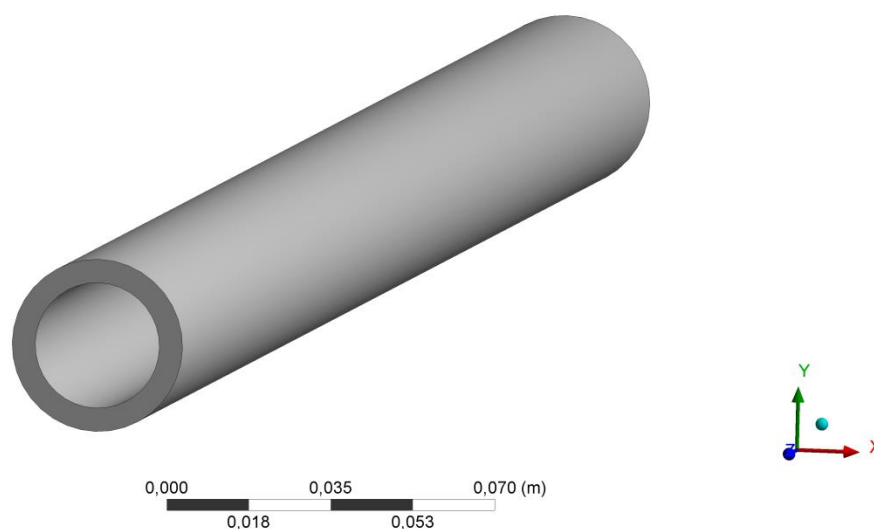


Figura 3.1. Geometría simplificada en un caso centrado.

Con el objetivo de obtener unos resultados lo más realistas posibles, las dimensiones de la geometría realista se han impuesto en base a los datos medidos de una geometría real [apartado geometría real]. En la Figura 3.2 se muestran las dimensiones de la geometría.

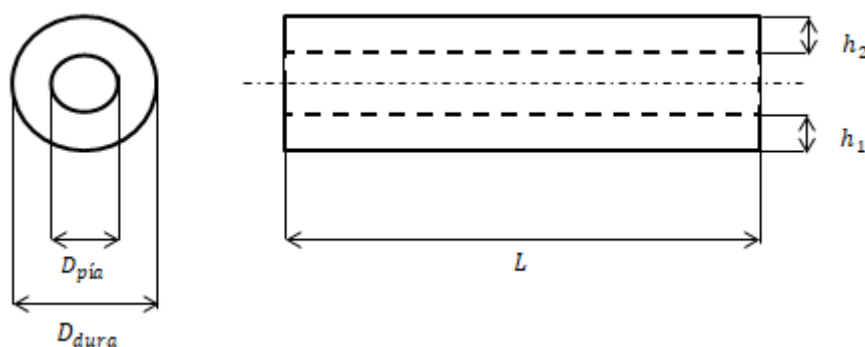


Figura 3.2. Dimensiones de la geometría simplificada.

En base a las dimensiones de la Figura 3.2 se define el parámetro β a partir de la Ecuación 3.2 y que nos dará el valor de la excentricidad entre ambos cilindros, e.

$$\beta = \frac{h_2 - h_1}{h_2 + h_1}$$

Ecuación 3.2

En la Tabla 3.1 se muestran los valores de las dimensiones para los diferentes casos a tratar con dicha geometría.

		Dimensiones (m)				
		D_{pia}	D_{dura}	L	h_1	h_2
Casos	$\beta = 0$	0,01344	0,01844	0,604	0,005	0,005
	$\beta = 0,25$	0,01344	0,01844	0,604	0,00375	0,00125
	$\beta = 0,5$	0,01344	0,01844	0,604	0,0025	0,0075
	$\beta = 0,75$	0,01344	0,01844	0,604	0,00125	0,00875

Tabla 3.1. Valores de cada una de las dimensiones de la geometría simplificada para los diferentes casos .

Se puede ver como la única dimensión que cambia de un caso a otro es la excentricidad que presenta la pía madre respecto a la dura. En la siguiente figura se muestra la diferencia entre cada una de las geometrías.

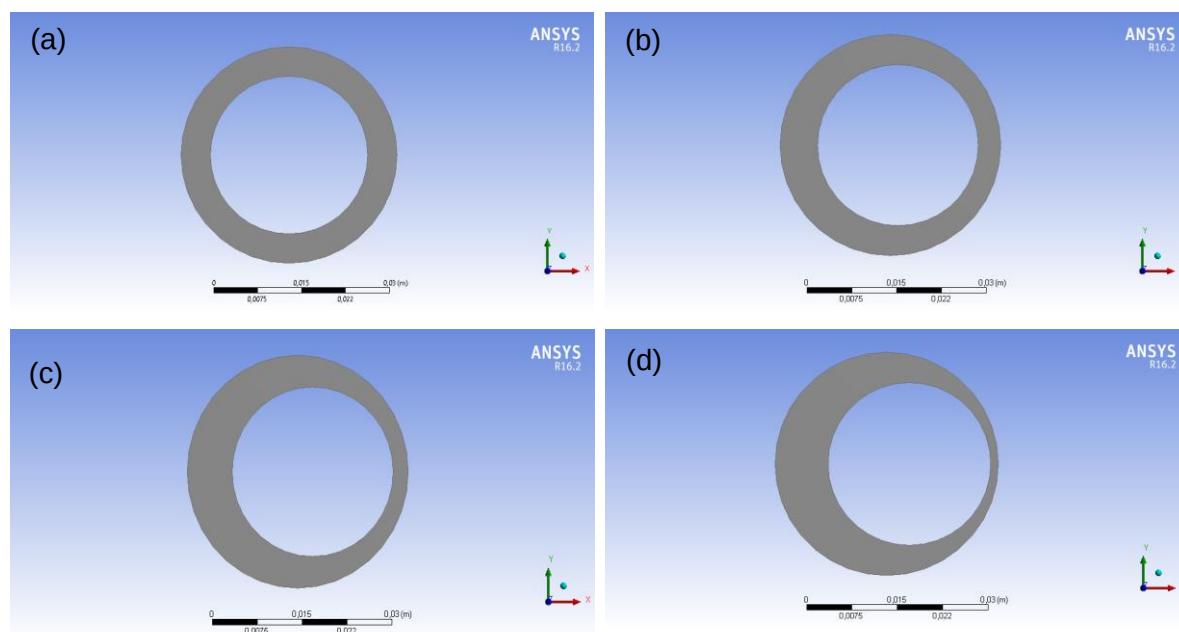


Figura 3.3. Capturas de la entrada de la malla en: a) caso $\beta=0$, b) caso $\beta=0.25$, c) caso $\beta=0.5$, d) caso $\beta=0.75$

Una vez generadas las geometrías para cada uno de los diferentes casos de β se importarán éstas en ANSYS Meshing y se realizará el mallado de cada una de ellas con el mismo número de elementos, obteniéndose las mallas que se pueden apreciar en la Figura 3.4

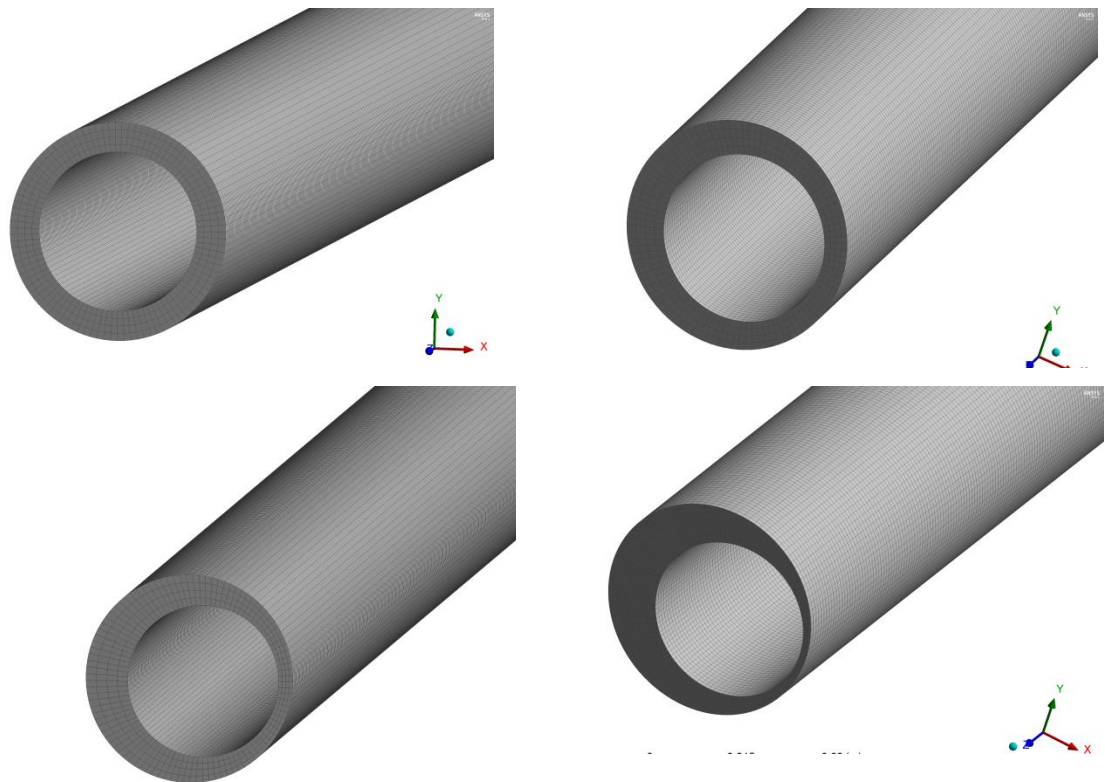


Figura 3.4. Representación de las mallas obtenidas para los diferentes valores de β .

Además se puede comprobar como la calidad de las mallas (Figura 3.5) es lo suficientemente buena como para obtener unos resultados precisos, atendiendo a las limitaciones de cálculo existentes.

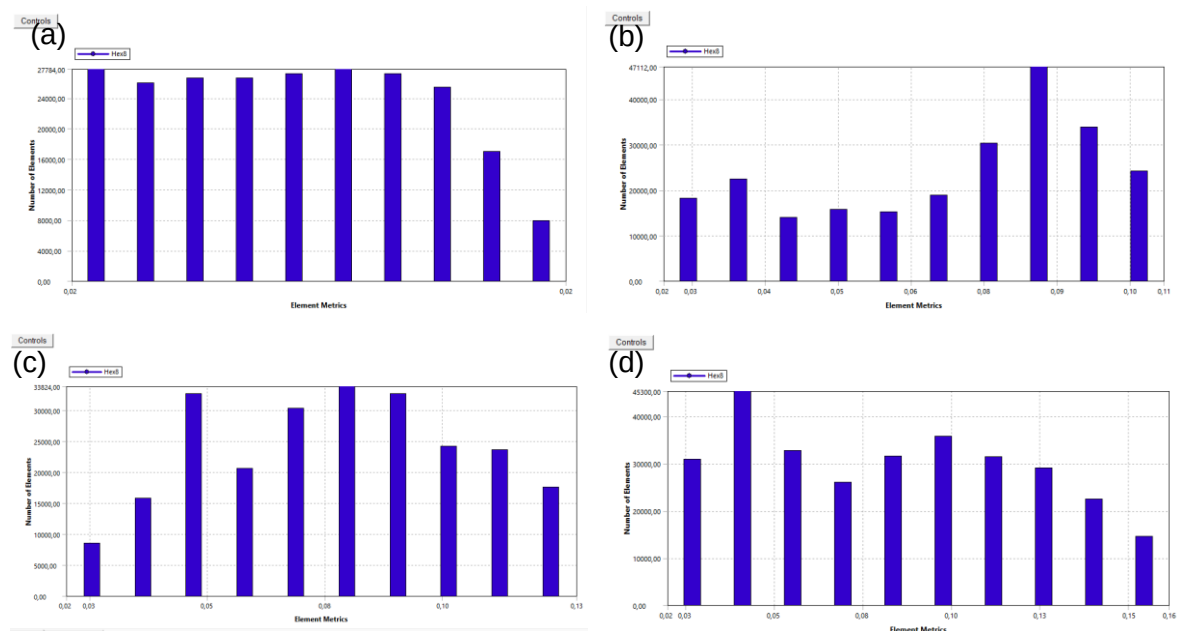


Figura 3.5. Histograma del parámetro de oblicuidad de las mallas para los casos: a) $\beta = 0$, b) $\beta = 0.25$, c) $\beta = 0.5$, d) $\beta = 0.75$.

De acuerdo a las estadísticas mostradas en la Figura 3.5 podemos decir que todas las mallas presentan una calidad excelente atendiendo al criterio de 'skewness' u oblicuidad ya que presentan un valor menor a 0,25 en todas sus celdillas, teniendo valores de la Tabla 3.2

CASO	OBLICUIDAD MEDIA
$\beta = 0$	0.020915
$\beta = 0.25$	0.069234
$\beta = 0.5$	0.077372
$\beta = 0.75$	0.083379

Tabla 3.2. Valor de la oblicuidad media para las diferentes mallas simplificadas.

3.2. Geometría real.

Ésta geometría ha sido facilitada por la Universidad de Jaén para crear el dominio computacional utilizado en los casos realistas, en los que se utilizará un caudal realista, como podrá verse en los próximos apartados. Dicha geometría se ha construido partiendo de una nube de puntos que a su vez se obtiene por medio de técnicas MRI a un sujeto humano.

La longitud de ésta dicha geometría es aproximadamente la misma que la utilizada para los casos simplificados ya que, como se ha comentado anteriormente, se busca obtener los resultados más realistas posibles. Del mismo modo, la forma de la geometría es diferente a la que se tendría si el ensayo MRI se hubiese realizado a un individuo diferente, por lo que al observar los resultados de la simulación con las condiciones de contorno que se generarán en los apartados siguientes [6], los contornos, velocidades, presiones, etc, variarán a las que se tendrían en un sujeto diferente, pero no deben variar los fenómenos que ocurren en ésta.

Como se ha comentado, la Universidad de Jaén ha facilitado nubes de puntos para generar las superficies de la duramadre, piamadre y las raíces nerviosas, siendo obviadas éstas últimas ya que su consideración dificultaría tanto el mallado como la simulación del problema.

Para poder generar una malla computacional con dicha geometría debe tratarse previamente la misma ya que presenta zonas que dificultan el mallado, haciéndose necesario utilizar un programa CAD que permita realizar un suavizado de superficies, utilizándose, en este caso SolidWorks.

SolidWorks tiene un complemento Premium, Scanto3D, que nos permite, a partir de una nube de puntos modificar dicha superficie o modificar una geometría en formato .stl o .obj, generando una superficie cerrada con la que poder generar un sólido 3D que pueda ser editado.

Dicho sólido 3D se generará al convertir la nube de puntos en una malla triangular (Figura 3.6), para lo que puede ser necesario eliminar ciertos puntos de la geometría que puedan producir errores al tratar la misma. Una vez generada la malla triangular es procesada en Scanto3D, que crea un contorno a partir de los nodos de dicha malla. Scanto3D ofrece la posibilidad generar superficies con diferentes sensibilidades, en función del número de sub superficies que contenga la superficie principal.

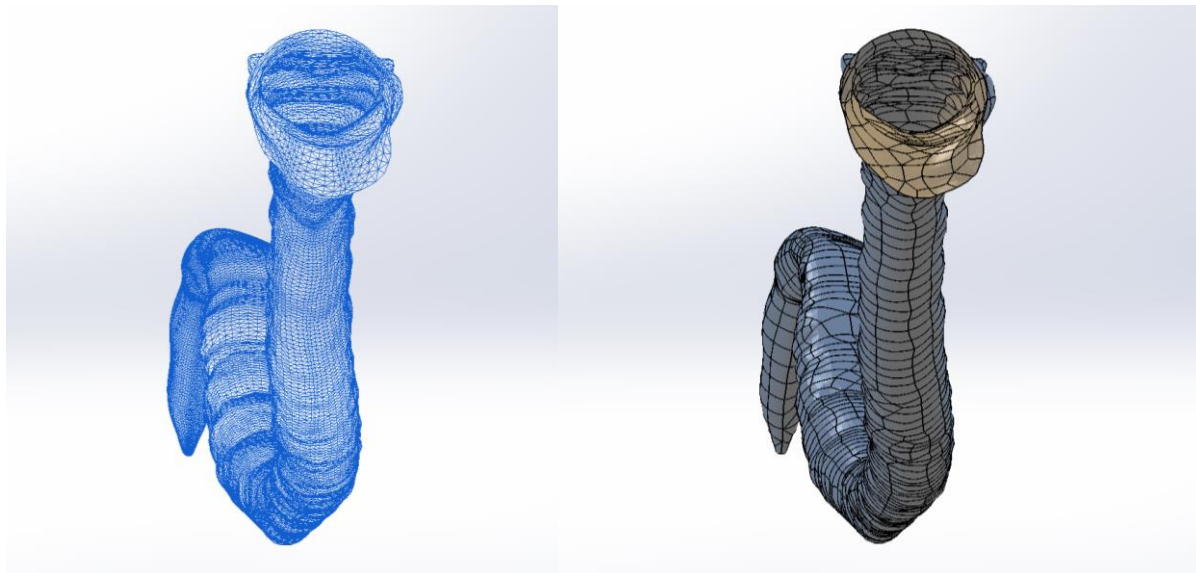


Figura 3.6. Malla triangular y superficie dividida en sub-superficies.

Tras haber indicado el nivel de sensibilidad se notifican aquellas sub-superficies que presentan errores y deberá ser reparadas de manera manual, obteniendo finalmente la superficie suavizada.

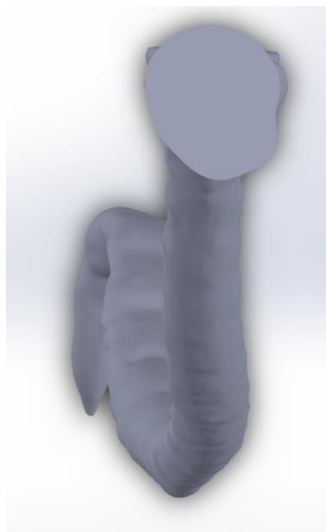


Figura 3.7. Geometría de la médula espinal suavizada con SolidWorks.

Una vez obtenida dicha geometría se comienza con el proceso de mallado, importando dicho sólido en Designer Modeler para definir los contornos de la malla y posteriormente, importar dicho sólido a ANSYS MESHING donde se indicarán los parámetros de la misma, obteniendo la malla que se muestra en la Figura 3.8.

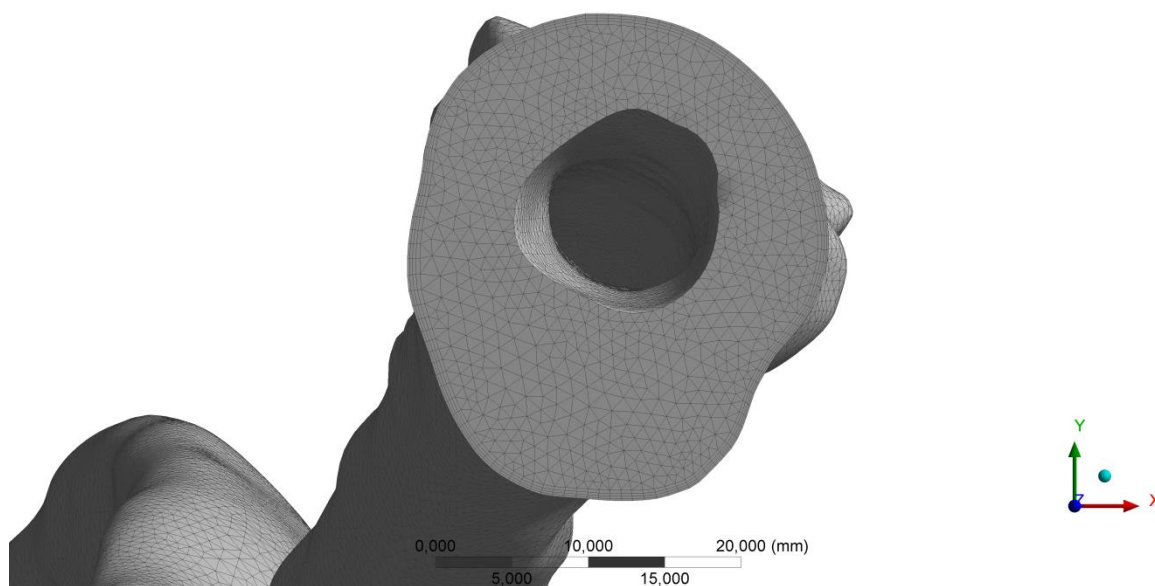


Figura 3.8. Malla computacional de la geometría realista.

4. OBTENCIÓN DE LAS UDF PARA MOVIMIENTO DE LA PARED.

4.2. Introducción a las UDFs.

Las UDF ('User Defined Functions') son funciones programables que pueden ser cargadas en ANSYS FLUENT con la finalidad de mejorar las características estándar del código de éste, de manera que agregan funcionalidades a ANSYS FLUENT como definir condiciones, propiedades, o términos fuente, así como especificar parámetros de algún modelo entre otras.

El lenguaje de las UDF está basado en lenguaje de programación C y puede escribirse en cualquier editor siempre y cuando el archivo sea guardado con extensión .c.

Para definir las UDF deben usarse los macros DEFINE, los cuales pueden consultarse en las guías de usuario de ANSYS FLUENT, [4]. Además todas las UDF deben tener un archivo udf.h que permita las definiciones de la macro seleccionada.

Las UDF pueden ser de dos tipos:

- **Interpretadas:** son las UDF que pueden ser interpretadas directamente desde la interfaz y pueden ser transferidas. Dichas funciones tienen una utilidad limitada ya que no permiten acceder a los datos del solver. Por otro lado, aunque son rápidamente asimiladas por el programa, son más lentas a la hora de realizar el cálculo.
- **Compiladas:** son las UDF que se están construidas del mismo modo que el ejecutable de ANSYS FLUENT, mediante un archivo compilado mediante el comando 'Build' dentro de la interfaz. Una vez construido dicho archivo, la UDF compilada puede ser cargada y usada. Son más rápidas que las UDF interpretadas a la hora de realizar cálculos pero no son transferibles.

4.2. Obtención de la UDF con caudal real.

Como se ha comentado anteriormente, se configurarán simulaciones con dos UDF diferentes, una con un caudal real, obtenido a partir de los datos obtenidos en estudios realizados previamente [5], y otra a partir de un caudal senoidal perfecto. En ambos casos la metodología a seguir es la misma, en primer lugar se expondrá la obtención del caudal de la misma, así como el amortiguamiento y el desfase que se introducirá en la misma, así como una representación del mapa de caudales a lo largo de la médula con el tiempo y finalmente, la obtención del desplazamiento para dicho caudal, así como la representación del mismo.

Para deducir la función del desplazamiento de las paredes de la médula recurrimos a la expresión de la variación del caudal a lo largo de la médula

$$\Delta Q(t, z) = -(Q(t, z = z_i) - Q(t, z = z_{i+1})) \quad \text{Ecuación 4.1}$$

Donde z_i indica la posición en la dirección axial de una sección cualquiera de la médula y z_{i+1} la posición en la dirección axial de la sección inferior a ésta. Si relacionamos la variación del caudal a lo largo de la médula con la variación del volumen del canal espinal se tiene que:

$$\Delta V = \Delta Q \cdot \Delta t \quad \text{Ecuación 4.2}$$

Si además se conoce que las paredes de la piamadre permanecen rígidas y no ven modificada su posición, la variación del volumen del canal espinal será proporcional al desplazamiento de las paredes de la duramadre.

$$\Delta V = \Delta h \, l(t, z) \, \Delta \bar{r} \quad \text{Ecuación 4.3}$$

De manera que, teniendo en cuenta todo esto, podemos expresar la variación del radio de la duramadre en función del caudal que atraviesa cada sección para cada instante de tiempo con Ecuación 4.4.

$$\Delta \bar{r} = \frac{\Delta Q}{\Delta h} \frac{\Delta t}{l(t, z)} \quad \text{Ecuación 4.4}$$

Que expresada en forma diferencial queda de la siguiente forma:

$$dr = - \frac{\partial Q(z, t)}{\partial z} \frac{dt}{l(t, z)} \quad \text{Ecuación 4.5}$$

Al haber tomado el origen de coordenadas en la parte superior de la médula, esto es el foramen magnum, si analizamos el flujo a lo largo de la médula la parcial tomará un valor negativo, por lo que se le ha dado a ésta un valor negativo, por otro lado, se considerará constante en el tiempo el perímetro de cada una de las secciones ya que el desplazamiento que sufren las paredes es muy pequeño, de manera que la Ecuación 4.5 queda como en la Ecuación 4.6, lo que simplificará el problema.

$$dr = - \frac{\partial Q(z, t)}{\partial z} \frac{dt}{2\pi R} \quad \text{Ecuación 4.6}$$

Dicha función será implementada con diferentes caudales en las UDFs de las diferentes simulaciones a realizar.

4.1.1. Obtención de la señal de caudal.

Como puede observarse en la Ecuación 4.6, para obtener el valor del desplazamiento para una determinada posición e instante de tiempo, debe conocerse el caudal en ese instante.

De datos experimentales es conocido que a medida que el flujo avanza por la médula espinal sufre una disminución en la amplitud, así como un desfase con respecto a la señal que se tiene en el foramen magnum, como puede verse en la Figura 4.3. En dicha figura se muestra el Stroke Volume, esto es el área encerrada

bajo la curva del primer semiperiodo de la señal del caudal, y del ángulo de desfase de la señal del caudal para un sujeto humano [6].

Del mismo modo, para obtener una señal de caudal realista se han tomado los datos utilizados en otro estudio [5], en el cual se tomaba como sujeto un macaco de cola larga, cuya anatomía, salvando las diferencias, es relativamente similar a la humana. Los valores del caudal de dicho estudio se extrajeron mediante técnicas MRI (Figura 4.1), y para éste trabajo se han extraído mediante la herramienta 'WebPlotDigitalizaer' [7]

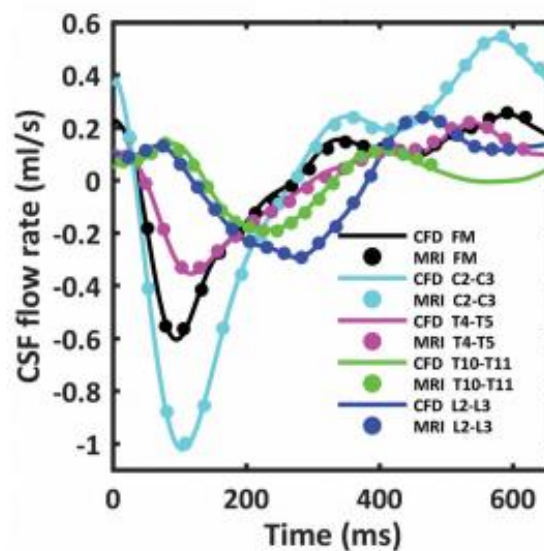


Figura 4.1 Señales de flujo másico a lo largo del tiempo para diferentes alturas de la médula [5]

Una vez se habían obtenido los valores de caudal para cada instante de tiempo se interpoló dicha curva en Matlab mediante un ajuste en series de Fourier de grado 8, obteniendo la curva mostrada en la Figura 4.2.

$$Q'(z, t) = [a_0 + a_1 \sin(wt) + \dots + a_n \sin(n \cdot wt) + b_1 \cos(wt) + \dots + b_n \cos(n \cdot wt)] \quad \text{Ecuación 4.7}$$

Término	Coeficiente a	Coeficiente b	W
0	-1,274	-	9,642
1	-3,874	-25,8	
2	11,78	-13,03	
3	7,396	-3,978	
4	5,99	2,393	
5	1,795	3,11	
6	0,3514	2,454	
7	-1,665	1,072	
8	-0,3288	0,6413	

Tabla 4.1. Valor de los coeficientes de Fourier para la Ecuación 4.7

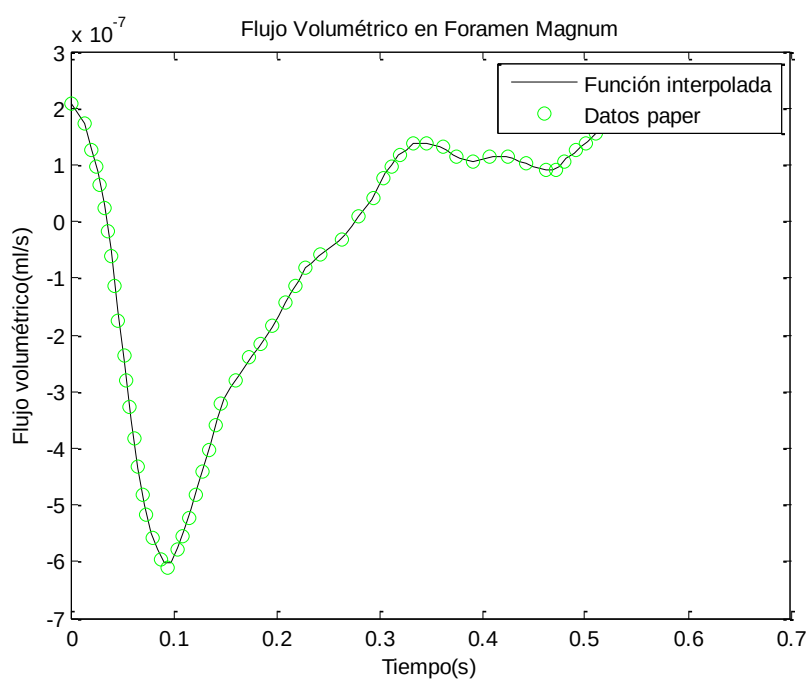


Figura 4.2. Curva de caudal en el foramen magnum interpolada a partir de los datos de [5].

4.1.1. Amortiguamiento y desfase de la señal.

Como se ha comentado, es conocido que la señal de caudal a lo largo de la coordenada longitudinal se amortigua y desfase, por lo que se ha tratado de reproducir dicho fenómeno añadiendo a nuestra señal de caudal un coeficiente de amortiguamiento que varía con la posición longitudinal, como puede verse en la Ecuación 4.8.

$$Q(z, t) = Ca(z) \cdot Q'(z, t)$$

Ecuación 4.8

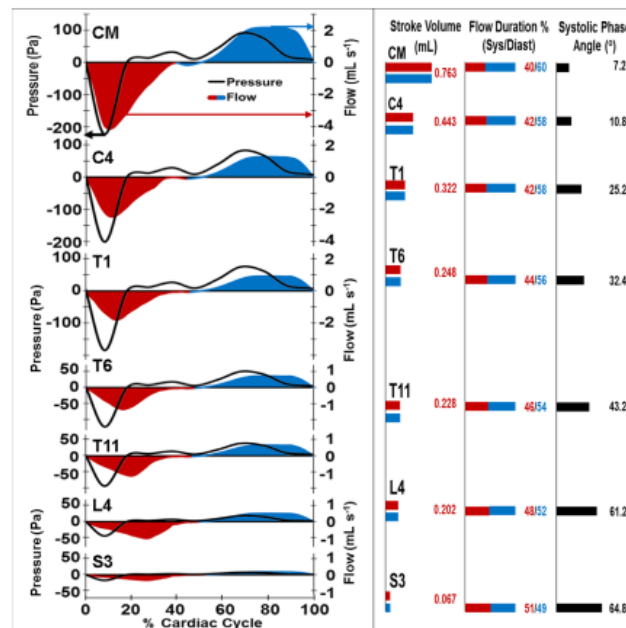


Figura 4.3 Datos de Stroke Volume y Desfase de la señal a diferentes alturas [6]

De acuerdo a dichos valores, para cada una de las alturas a las que se encuentran dichos valores para el sujeto del estudio realizado en [6], se han obtenido mediante polinomios las funciones que interpolan dichos puntos, de manera que se han obtenido tanto una función para el amortiguamiento de la señal con la posición, $Ca(z)$, como una función que proporciona un desfase a la señal en función de la posición, $\phi(z)$, como se muestra en la Ecuación 4.9 y Ecuación 4.10.

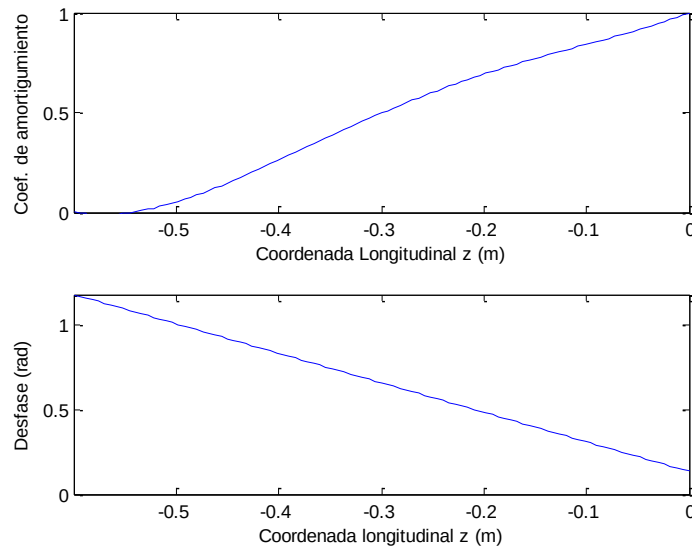


Figura 4.4 Funciones interpoladas para el coeficiente de amortiguamiento y desfase de la señal de caudal.

$$Ca(z) = 26,26 \cdot z^4 + 25,31 \cdot z^3 + 6,274 \cdot z^2 + 1,989 \cdot z + 1 \quad \text{Ecuación 4.9}$$

$$\phi(z) = -1,7473 \cdot z + 0,1353 \quad \text{Ecuación 4.10}$$

Teniendo en cuenta estas dos últimas consideraciones, la ecuación del caudal presentará la forma mostrada en

$$Q(z, t) = Ca(z) \cdot [a_0 + a_1 \sin(wt - \phi(z)) + \dots + a_n \sin(n \cdot wt - \phi(z)) + b_1 \cos(wt - \phi(z)) + \dots + b_n \cos(n \cdot wt - \phi(z))] \quad \text{Ecuación 4.11}$$

Si se representa la Ecuación 4.11 para un ciclo cardiaco completo a lo largo de la médula se obtiene el mapa de flujo volumétrico de la Figura 4.5.

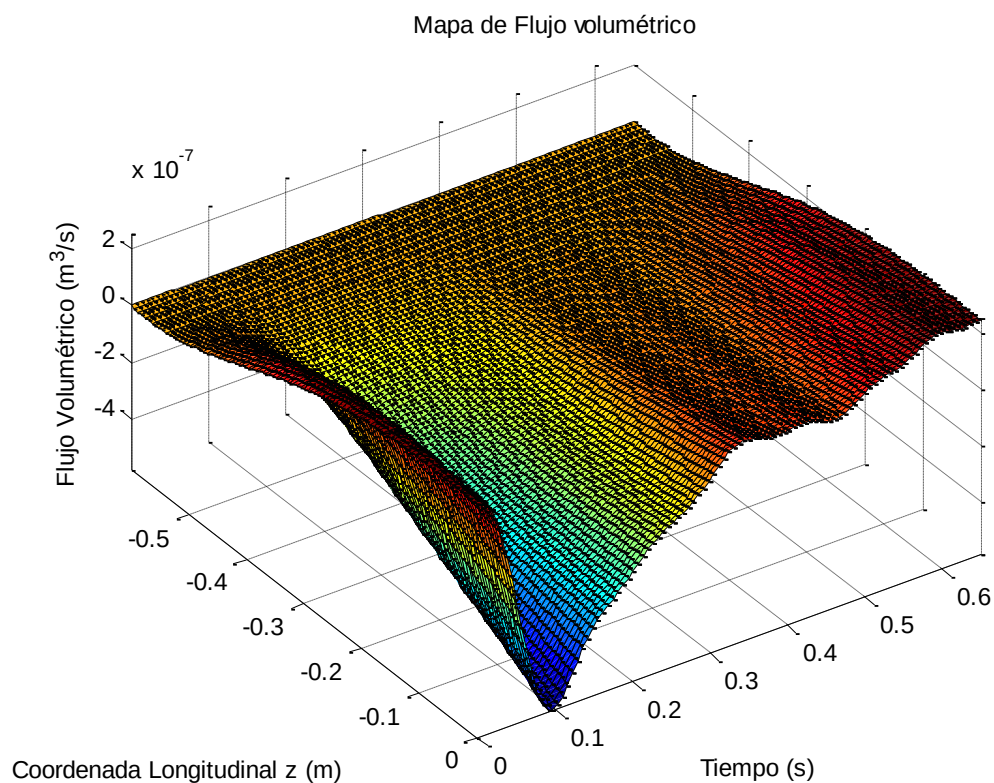


Figura 4.5 Mapa de flujo volumétrico para la señal de caudal real.

Una vez obtenida la función del caudal, se calcularán los desplazamientos para cada instante de tiempo y posición a partir de la Ecuación 4.6, obteniendo el siguiente mapa de desplazamientos.

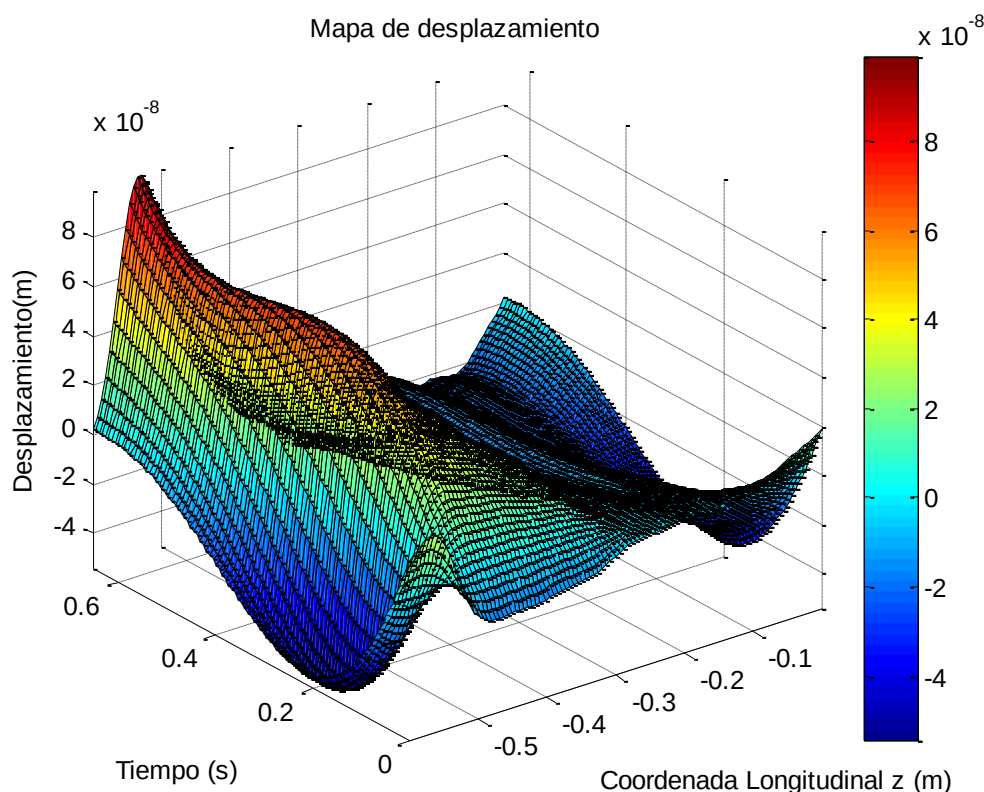


Figura 4.6. Mapa de desplazamientos obtenidos de un caudal real.

En la Figura 4.6 puede apreciarse que del mismo modo que sucede con el caudal, a medida que nos alejamos del foramen magnum, la señal de desplazamiento se atenúa, llegando a sus valores mínimos en la parte inferior del canal espinal, con la diferencia de que los desplazamientos no llegan a tener un valor nulo constante en dicha posición ya que aparecen los términos relacionados con la derivada del caudal del desplazamiento.

En la Figura 4.7 se han representado las señales de desplazamiento a diferentes alturas, pudiendo apreciarse como disminuye el valor máximo a medida que varía la posición, así como un desfase entre éstas.

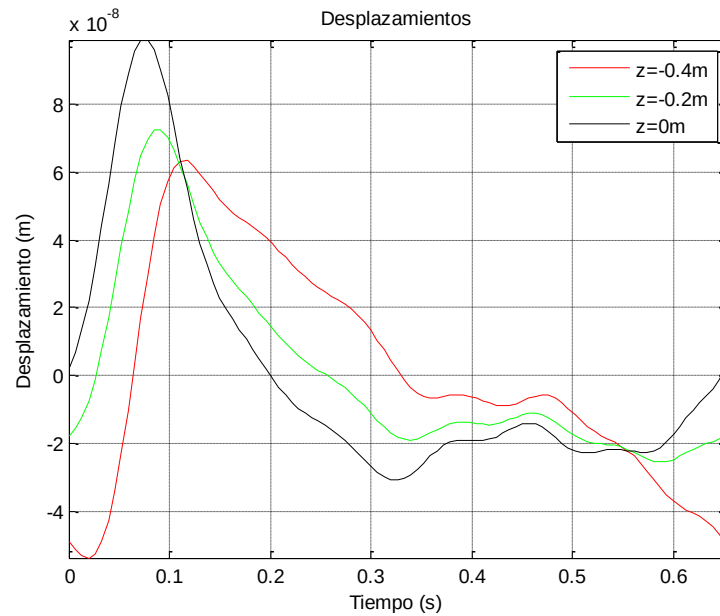


Figura 4.7 Señal de desplazamiento para un caudal real a diferentes alturas.

4.2. Obención de la UDF para los casos canónicos.

Con el objetivo de simplificar el problema, con el fin de poder apreciar de una manera más sencilla los fenómenos del estado transitorio se propone un caudal senoidal de amplitud igual a la de la señal del caudal real del apartado anterior, de manera que se repetirá el proceso de obtención de la señal de caudal, con la diferencia de que la señal de caudal para éstos casos será más simple ya que al ser una onda senoidal perfecta no deberá aproximarse por Fourier.

La ecuación del caudal presenta la forma:

$$Q_{sin} = A \cdot \sin\left(\frac{2\pi}{T} t\right) \quad \text{Ecuación 4.12}$$

El valor de la amplitud de dicha onda será de 0,6 ml/s y el periodo de 0,6517 s, tal y como se ha utilizado para la señal real.

Al igual que con la onda real, con el fin de obtener una solución lo más próxima a la real se introducirán el coeficiente de amortiguamiento () y desfase de la señal (), de manera que la Ecuación 4.12 quedará como:

$$Q_{sin} = 0.6 \cdot 10^{-6} \cdot Ca(z) \cdot \sin\left(\frac{2\pi}{T}t - \phi(z)\right)$$

Ecuación 4.13

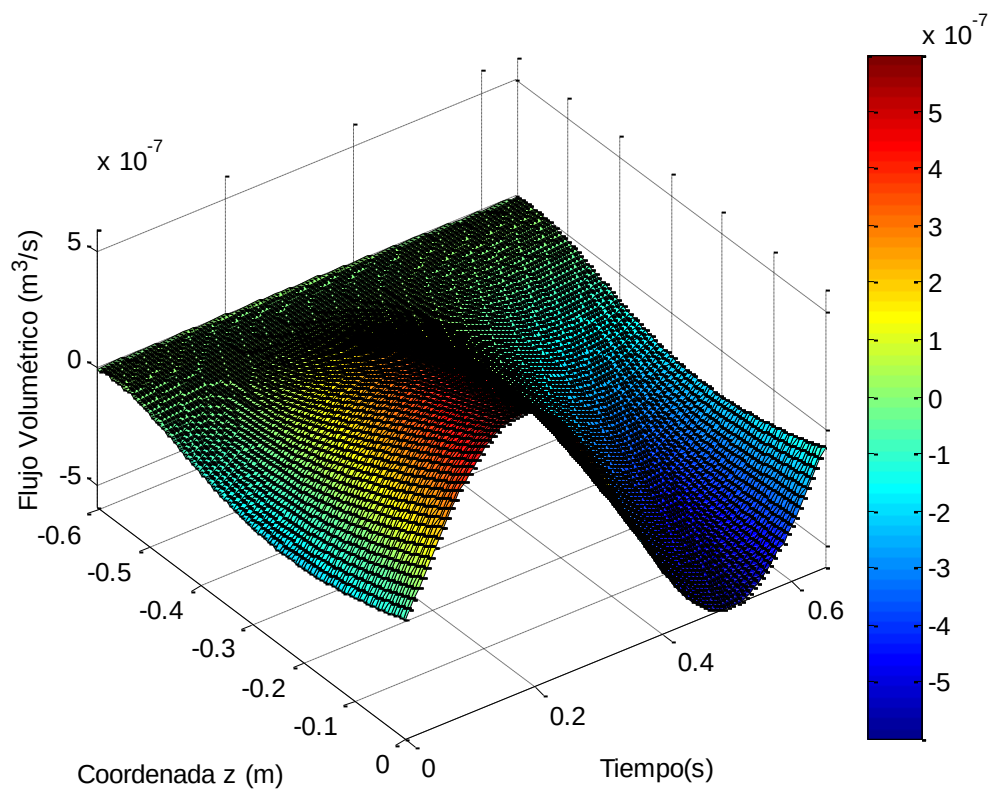


Figura 4.8. Mapa de Flujo volumétrico.

Partiendo de dicho caudal se calculará el campo de desplazamientos en para cada altura e instante de tiempo, obteniendo el siguiente mapa de desplazamientos.

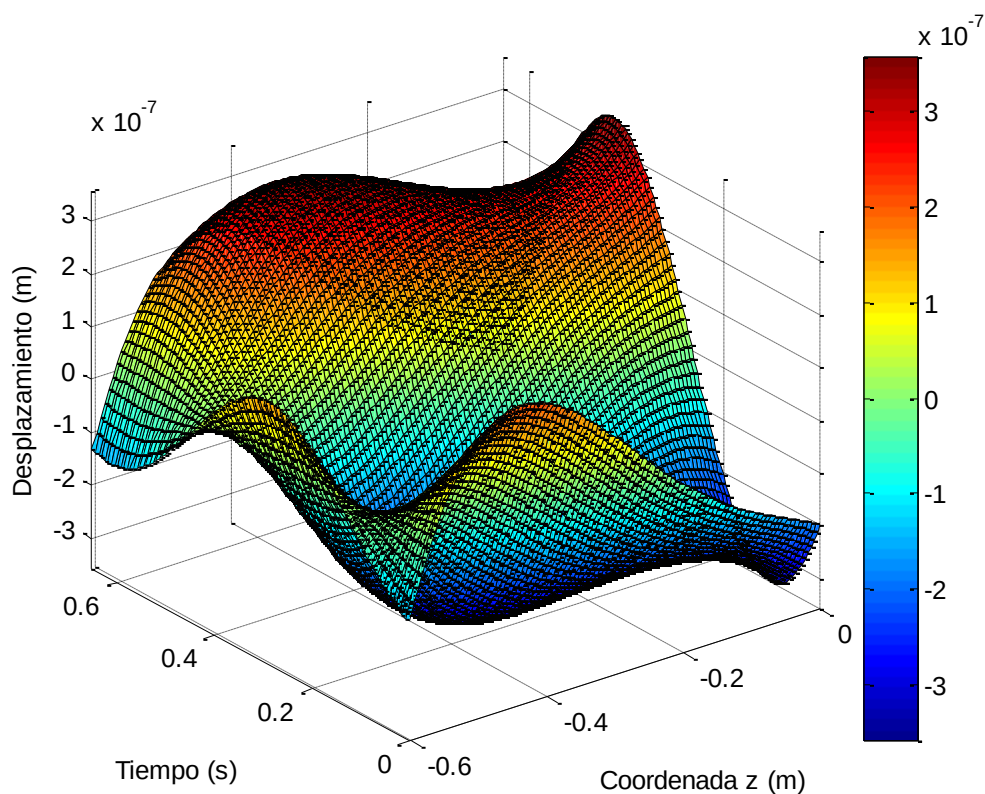


Figura 4.9 Mapa de desplazamientos obtenidos de un caudal senoidal.

Puede observarse que, al igual que ocurría en el caso del caudal real, el desplazamiento se amortigua a medida que avanza en la dirección longitudinal pero no llega a hacerse nula, si no que toma valores muy pequeños debido a los términos de la derivada del coeficiente de amortiguamiento.

5. CONFIGURACIÓN DE LOS CASOS.

Una vez se han definido tanto las 'UDF' como los dominios computacionales para las diferentes simulaciones se configurarán las diferentes simulaciones, como hemos dicho, en los casos canónicos las funciones utilizadas son idénticas, cambiando tan solo la geometría del caso. Por tanto, a continuación se detallará como se han configurado las diferentes simulaciones del caso canónico.

En primer lugar se configurarán las simulaciones como simulaciones transitorias 3D y con 'solvers' basados en la presión, con formulación en velocidad absoluta.

Como las velocidades que alcanza el flujo son tan pequeñas, el número de Reynolds lo será también, por tanto suponemos que el flujo por la médula espinal será laminar, sin transferencias de calor, ni especies, por lo que no será necesario introducir ningún modelo adicional.

El siguiente paso será definir el fluido utilizado durante la simulación, que en este caso será el CSF cuyas propiedades hidrodinámicas se considerarán equivalentes a las del agua a la temperatura corporal, esto es una densidad de $\rho = 993,3 \text{ kg/m}^3$ y una viscosidad dinámica de $\mu = 0,6913 \text{ mPa} \cdot \text{s}$, de acuerdo a las condiciones detalladas en [5].

El siguiente paso será configurar las condiciones de contorno. Todos los contornos se configuran con condición de pared sin deslizamiento a excepción del foramen magnum, el cual se configura con una condición de presión en la entrada con un valor de 0 Pa.

Se habilitará el método de malla dinámica, de acuerdo a lo comentado en el capítulo anterior, se seleccionará el método 'Smoothing' con un 'Spring Factor' bajo (en este caso 0.05), de manera que la deformación de las paredes afecte en la menor medida posible a las celdillas más lejanas al contorno. Una vez habilitados los métodos dinámicos se deberá definir el movimiento de las zonas deseadas, así como configurar los contornos contiguos al contorno con movimiento. De ésta forma, en función de la geometría utilizada se configurará como:

• **Geometría simplificada:** el movimiento se ha se le concede a la dura madre, dándole movimiento mediante las UDF detalladas anteriormente, que han debido ser compiladas correctamente, y los contornos con los que colinda dicho contorno móvil, que son la entrada de la malla o foramen magnum y la base de la malla o bottom. Para ambos contornos ha de indicarse el plano en el que están contenidos, siendo estos planos perpendiculares a la dirección del flujo, z, y un punto contenido en dichos contornos, siendo éstos los puntos (0,0,0) m y (0,0,-0.604) m, respectivamente.

• **Geometría real:** en este caso, al igual que en el anterior el movimiento se le dará a la dura madre, con la diferencia de que en esta geometría tan solo se definirá la posición del foramen magnum ya que no ahora la base será la cola de caballo, la cual también tiene movimiento. La posición del foramen magnum para esta geometría es idéntica a la anterior.

Se ha utilizado un solver SIMPLE para el acoplamiento de presión y velocidad, y para la discretización espacial se han utilizado los una aproximación por minimos cuadrados basado en las celdillas para el término gradiente, de segundo orden para la presión y Upwind de segundo orden para el momento. Finalmente, para la discretización espacial se ha utilizado una discretización implícita de segundo orden.

Para comprobar que el cálculo está realizándose de manera correcta se han introducido monitores de flujo volumétrico con el tiempo, de manera que puede comprobarse que la señal de flujo volumétrico a lo largo del canal espinal es correcta.

Se ha comentado que uno de los objetivos fundamentales del trabajo es el de verificar la existencia de una velocidad estacionaria encargada de la recirculación del CSF, por lo que se simularán 40 ciclos para asegurarnos que el problema llega a ese estado estacionario, de los cuales guardaremos los datos de cada paso de tiempo pero solo se almacenará el último paso de tiempo, con el objetivo de ahorrar memoria. Dichos 40 ciclos no será necesario simularlos con gran resolución por lo que se dará un paso de tiempo tal que cada ciclo cardiaco tenga 25 pasos de tiempo. Una vez alcanzados los 40 ciclos se volverá a simular con una mayor precisión en términos de paso de tiempo, reduciendo éste y obteniendo 101 puntos

por cada ciclo cardíaco, los cuales serán guardados para su post-procesado. A parte de modificar el paso de tiempo en la interfaz de ANSYS FLUENT debe cambiarse la UDF a utilizar ya que, como se puede apreciar en la Ecuación 4.6, el valor del desplazamiento está directamente relacionado con el valor del paso de tiempo.

6. ANÁLISIS DE LOS RESULTADOS.

6.1. Casos con geometría simplificada.

Como sabemos, en el canal espinal la posición del centroide de la piamadre no es constante en función de la coordenada axial y por tanto, en diferentes secciones, siendo el flujo diferente en éstas. Por tanto, al realizar las simulaciones con la geometría simplificada desplazando el cilindro interior (piamadre) con respecto al exterior (duramadre) podremos estudiar el comportamiento que tiene el flujo en esos casos para finalmente, extrapolarlo a un caso de una médula real.

Por tanto, con la geometría simplificada obtenida en el apartado 3.1 se han realizado simulaciones para diferentes valores de β (Figura 3.3) con los caudales obtenidos en los apartados 4.2 y 4.2.

6.1.2. Resultados casos canónicos.

El objetivo de analizar los resultados de dichas simulaciones, entre otros, es el de verificar que existe una velocidad estacionaria, calculada como el promedio temporal de la velocidad longitudinal, aproximadamente 2 órdenes de magnitud menor que la velocidad oscilatoria. Como se comenta en la introducción, la dinámica del CSF ha tomado una gran importancia en tratamientos del CNS mediante inyección intratecal ya que es esta velocidad estacionaria la responsable de dicho movimiento recirculatorio de CSF.

Del mismo modo, se validará que las funciones implementadas en las UDFs reproducen de manera el flujo volumétrico deseado en el canal espinal, para lo que se tomarán los datos de simulación a diferentes alturas y se compararán con los valores teóricos, tanto a lo que amplitud de la señal se refiere como al desfase del mismo, así como estudiar el desfase que se produce en la dirección perpendicular al flujo de CSF.

El resto de variables, tanto en este como en el caso de la geometría realista será tratado en [referencia trabajo raul]

6.1.1 Caso canónico 1, $\beta=0$

Tanto en este caso como en el demás, las alturas a las cuales se tomarán los datos serán en la entrada de la malla (foramen magnum), y a unas distancias de 200 y 400 mm desde la misma.

En primer lugar vamos a analizar la velocidad axial, v_z , en un ciclo cardiaco completo en un punto medio entre los diámetros ambos cilindros, ya que es en dicha zona donde se alcanzan las velocidades máximas.

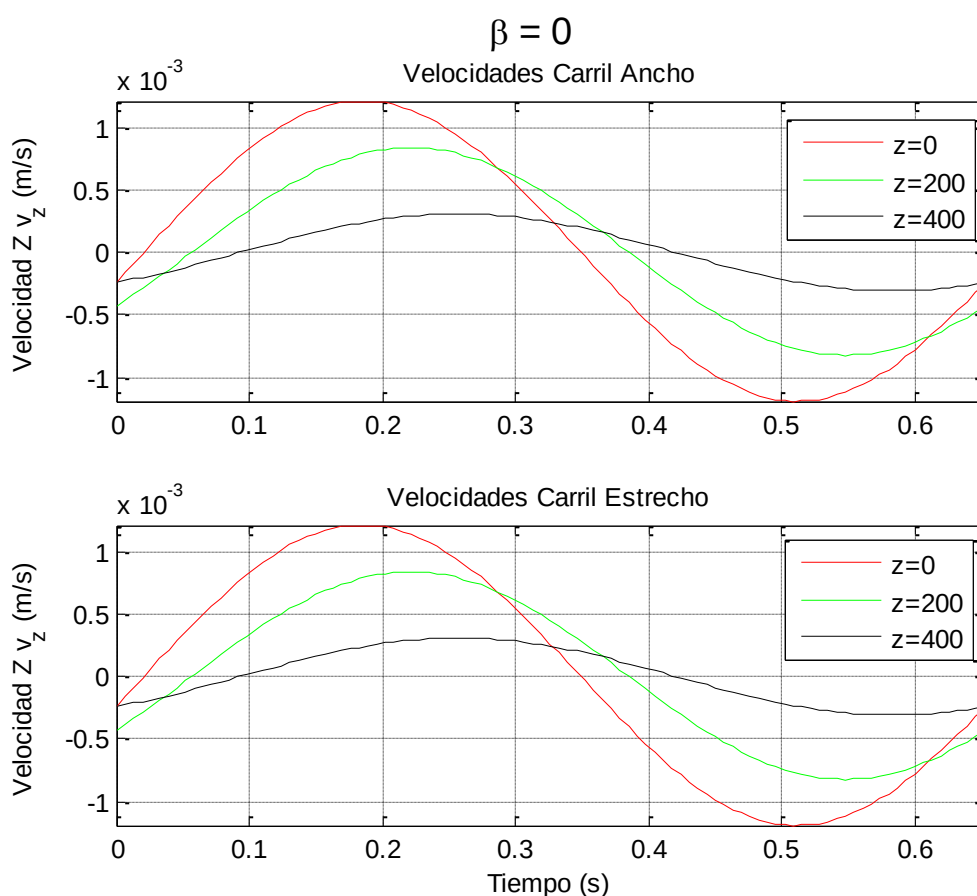


Figura 6.1. Velocidades longitudinales en el punto medio de cada canal para $\beta = 0$

En este caso, al no existir una excentricidad entre ambos cilindros, el flujo que circula por ambos carriles es exactamente igual y por tanto, las velocidades son exactamente iguales ya que el desplazamiento de las paredes del cilindro exterior es igual para todos los nodos de cada sección en un determinado instante de tiempo.

También puede apreciarse, al igual que en la Figura 6.2 que se produce un desfase entre las velocidades a medida que nos alejamos de la entrada de la malla.

En la Figura 6.2 se comparan los flujos volumétricos obtenidos de las simulaciones con los caudales utilizados para implementar el código de las UDFs, obteniendo unas diferencias prácticamente inexistentes.

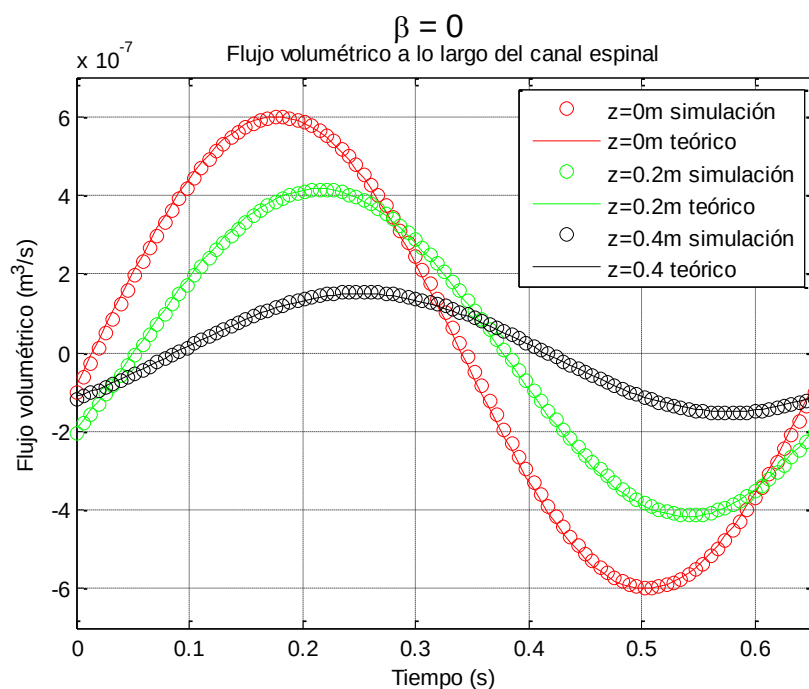


Figura 6.2 Comparación entre los caudales obtenidos mediante simulación y los teóricos para $\beta = 0$

Por otro lado, como ha sido comentado anteriormente, se mostrarán las velocidades estacionarias en diferentes puntos en una misma sección en cada una de las alturas mencionadas anteriormente, obteniendo los perfiles de Womersley de la Figura 6.5, para cada uno de los carriles, que en este caso son prácticamente iguales, por lo que solo se mostrará uno.

Puede observarse en la Figura 6.5 como la velocidad estacionaria es fundamentalmente de sentido positivo, pero a medida que se desciende por el canal espinal se va reduciendo hasta hacerse completamente negativo.

Así mismo, se representarán de manera conjunta, en la Figura 6.4 para una serie de puntos tanto la velocidad oscilatoria como la velocidad estacionaria, pudiendo comprobar que la velocidad estacionaria es aproximadamente 2 órdenes

de magnitud menor. Las velocidades han sido tomadas de 6 puntos equidistantes en uno de los dos canales en la entrada de la malla, como se muestra en la Figura 6.3.

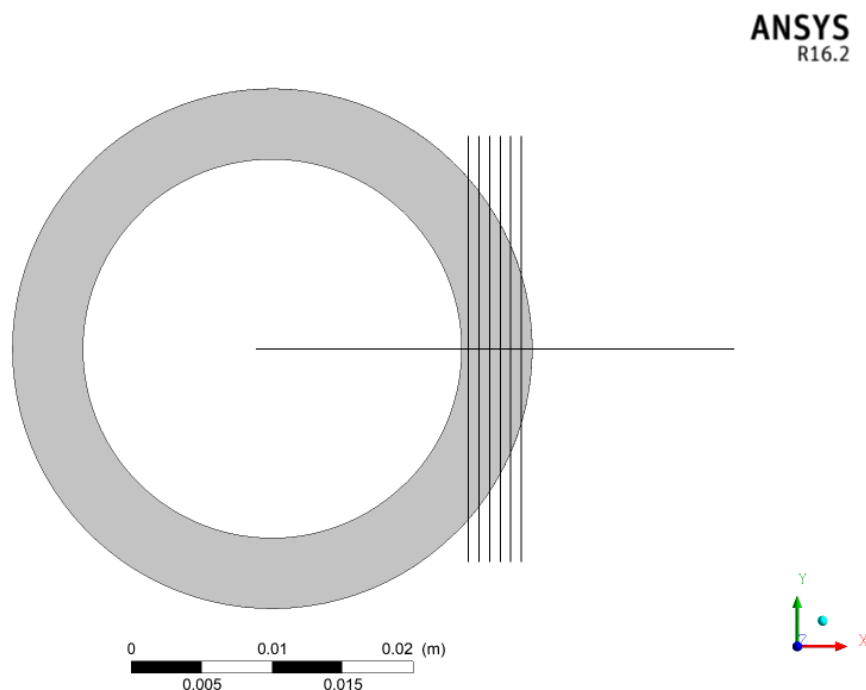


Figura 6.3. Posición de los puntos tomados para representar la velocidad.

Las coordenadas de dichos puntos de izquierda a derecha son:

Punto	X (m)	Y (m)	Z (m)
1	0.01394	0	0
2	0.01469		
3	0.01544		
4	0.01619		
5	0.01694		
6	0.01769		

Tabla 6.1 Coordenadas de los puntos de la Figura 6.3

En la Figura 6.4 puede observarse como la velocidad presenta un cierto desfase, no solo en la dirección axial sino que también está desfasada, en menor medida, en la dirección radial, de manera que durante un breve periodo de tiempo para el mismo canal de una sección se pueden observar velocidades tanto de subida como de bajada, teniendo flujo en ambas direcciones.

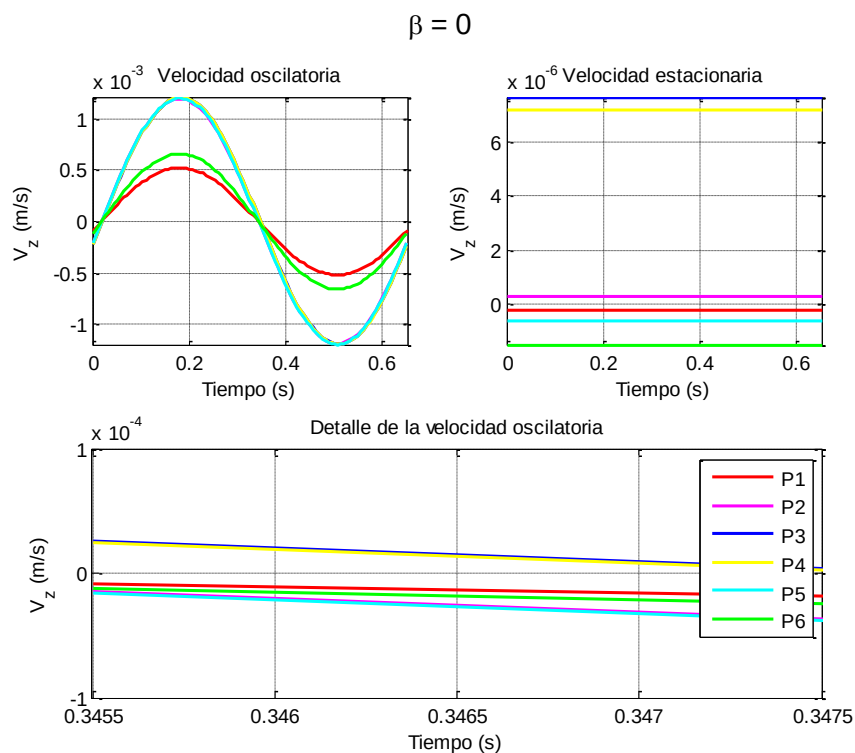


Figura 6.4 Comparativa entre las velocidades oscilatoria y estacionaria en diferentes puntos para $\beta = 0$.

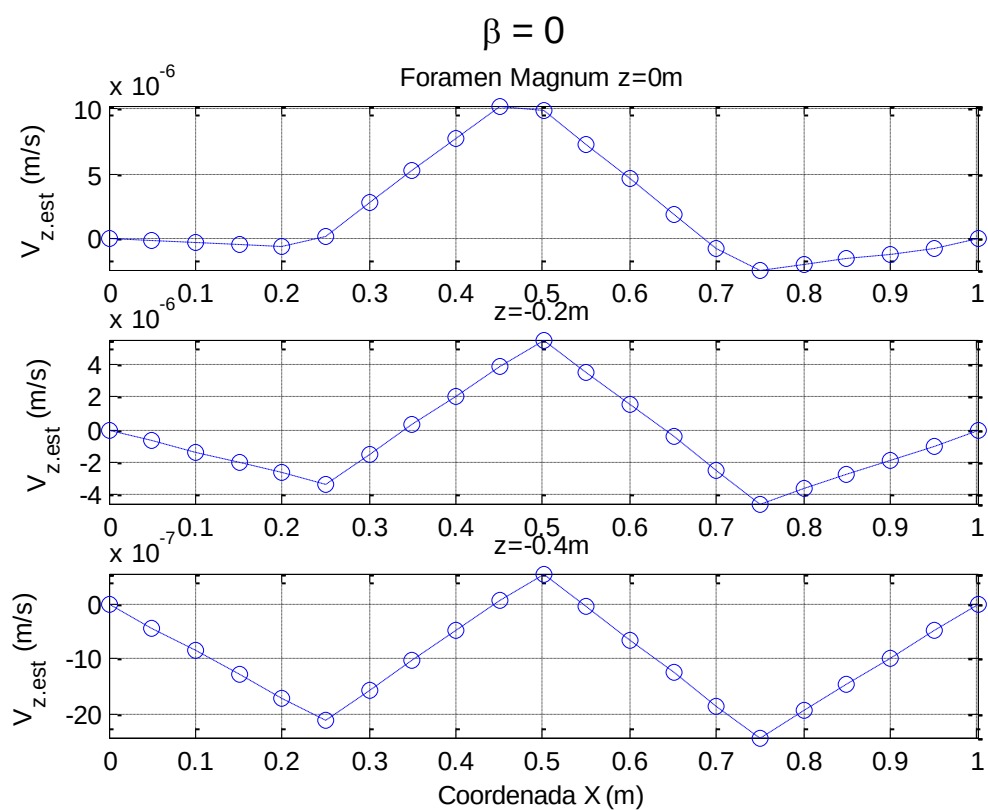


Figura 6.5. Perfil de velocidad estacionaria para $\beta = 0$.

6.1.2 Caso canónico 2, $\beta=0.25$.

Del mismo modo que para el caso anterior, se representarán las velocidades en el punto medio entre los diámetros de ambos cilindros para cada uno de los dos canales, que esta vez tienen diferente tamaño ya que en este caso sí que existe una cierta excentricidad.

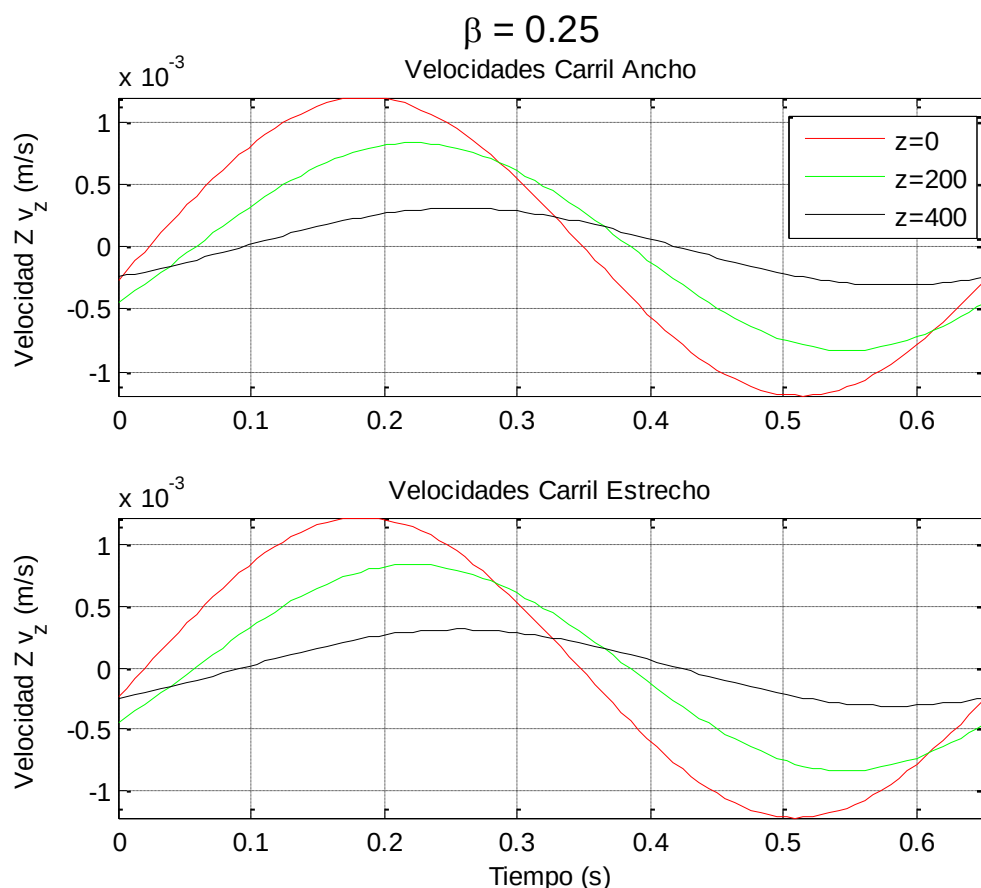


Figura 6.6 Velocidades longitudinales en el punto medio de cada canal para $\beta = 0.25$

En la Figura 6.6 observarse como las velocidades en los puntos del canal ancho y estrecho son prácticamente las mismas, existiendo una diferencia de aproximadamente 2% entre los valores pico en ambos puntos para una misma altura e instante de tiempo.

De nuevo se verificará que los caudales obtenidos en la simulación para diferentes alturas a lo largo del tiempo se ajustan a los valores utilizados en las funciones para implementar las UDFs.

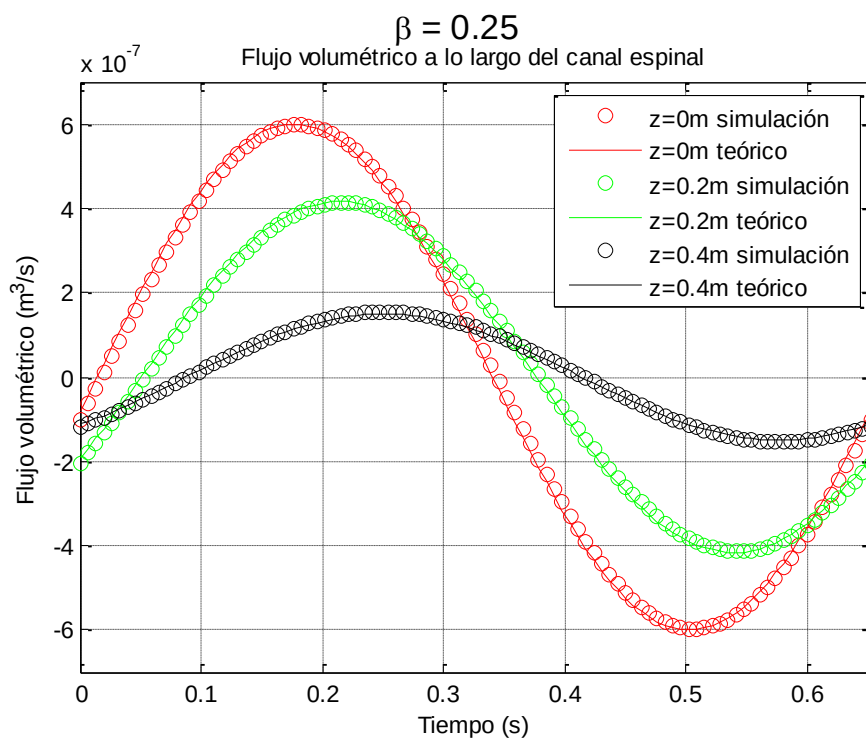


Figura 6.7 Comparación entre los caudales obtenidos mediante simulación y los teóricos para $\beta = 0.25$

En este caso se representarán los perfiles de Womersly para los dos canales ya que en este caso no son iguales. Del mismo modo, se representará tanto la velocidad oscilatoria como la velocidad estacionaria en los puntos que se muestran en la Figura 6.8, con el objetivo de ver el sentido de las velocidades estacionarias en dichos puntos, así como comprobar cuál es su orden de magnitud con respecto a las velocidades oscilatorias.

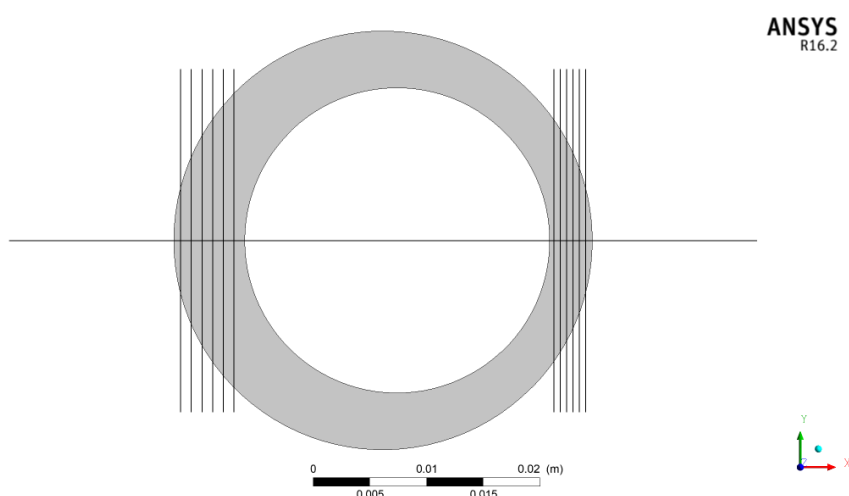


Figura 6.8. Posición de los puntos tomados para representar la velocidad.

Las coordenadas de los puntos de la Figura 6.8 se muestran en la siguiente tabla:

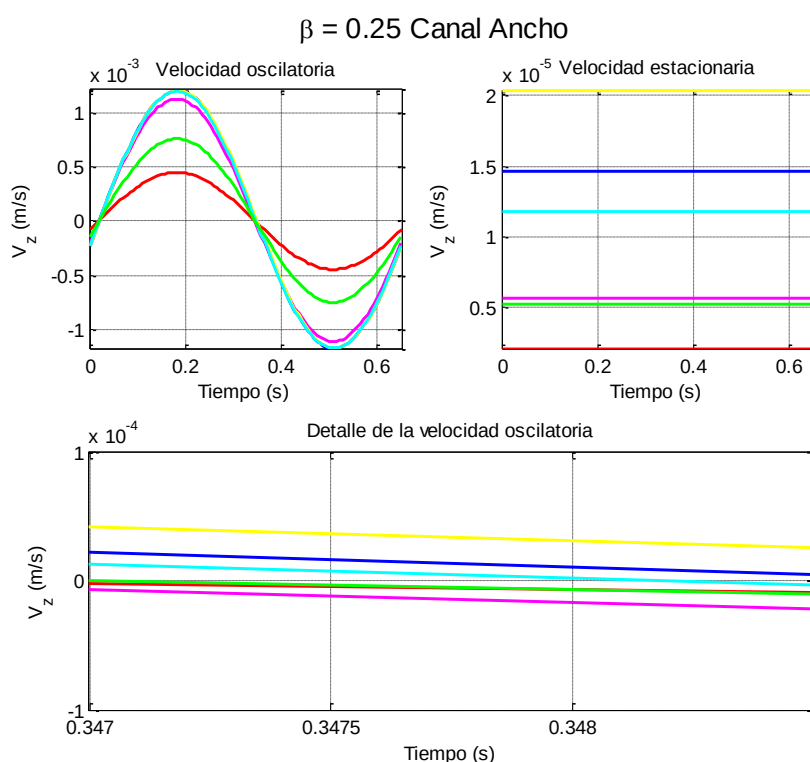
Punto	X (m)	Y(m)	Z(m)
1	-0.017815	0	0
2	-0.0168775		
3	-0.01594		
4	-0.0150025		
5	-0.014065		
6	-0.0131275		
7	0.015065		
8	0.0156275		
9	0.01619		
10	0.0167525		
11	0.017315		
12	0.0178775		

Tabla 6.2. Coordenadas de los puntos de la Figura 6.8

Como puede observarse en la Figura 6.9, las velocidades oscilatorias son de 2 órdenes de magnitud mayores que las velocidades estacionarias, como se sabe de datos experimentales. Así mismo, en la Figura 6.11 puede verse como en el canal ancho las velocidades estacionarias tienen valor positivo en la entrada de la malla y, a medida que se avanza a lo largo de la coordenada longitudinal comienzan a

aparecer velocidades de valor negativo en las zonas cercanas a las paredes, esto es, zonas con recirculación donde la velocidad toma el sentido opuesto, mientras que en el canal estrecho la velocidad estacionaria se mantiene negativa a lo largo del canal espinal.

Si se comparan los valores máximos de las velocidades estacionarias del canal ancho (Figura 6.11) y del canal estrecho (Figura 6.12) se puede apreciar como las del canal ancho son, en valor absoluto, ligeramente mayores a las del canal estrecho. Del mismo modo, si se comparan los valores de velocidad máxima en el canal ancho obtenidos en éste caso con los obtenidos en el caso anterior vemos que éstos han aumentado, para las diferentes alturas.



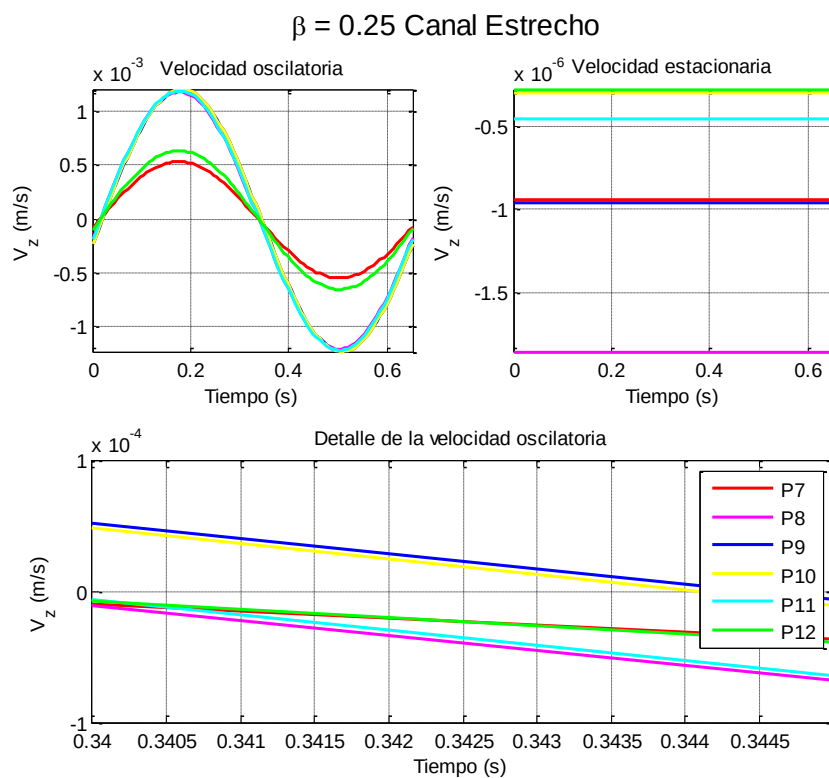


Figura 6.10. Comparativa entre las velocidades oscilatoria y estacionaria en diferentes puntos del canal estrecho para $\beta = 0.25$

Tanto en la Figura 6.9 como en la Figura 6.10 aparecen desfases entre las velocidades en dirección radial, aparte del existente en la coordenada longitudinal.

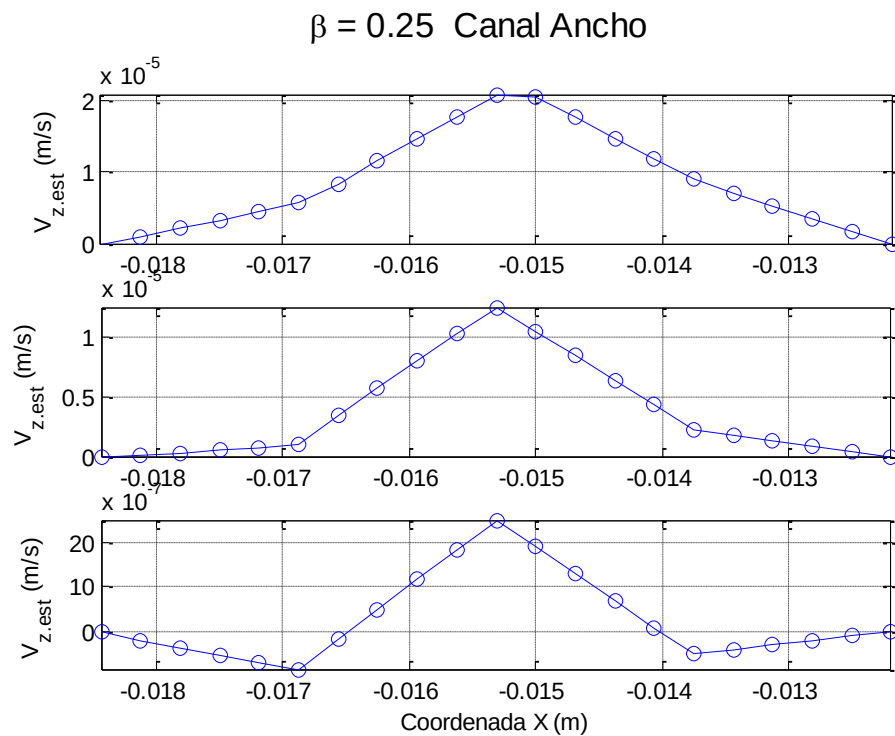


Figura 6.11. Perfil de velocidades estacionarias para el canal ancho en el caso $\beta = 0.25$

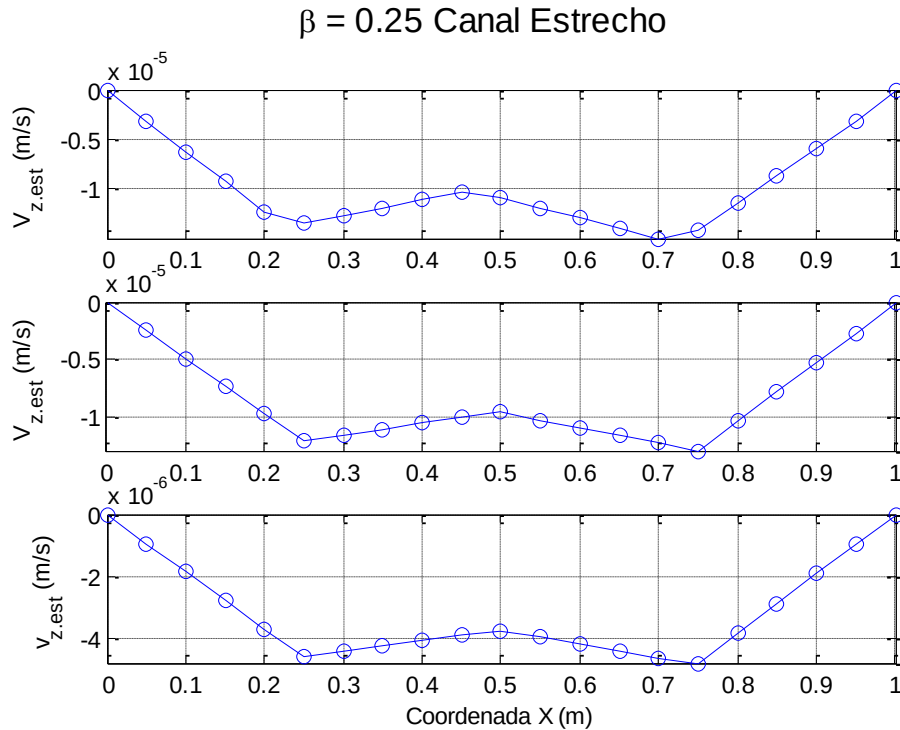


Figura 6.12. Perfil de velocidades estacionarias para el canal estrecho en el caso $\beta = 0.25$

6.1.3. Caso canónico 3, $\beta=0,5$.

De nuevo, para el caso de $\beta = 0.5$ se representarán las velocidades en el punto medio entre ambos cilindros en secciones a las mismas alturas que en casos anteriores, observándose en la Figura 6.13, que al igual que antes las velocidades presentan un desfase prácticamente nulo. Puede apreciarse también como la amplitud de la velocidad en ambas secciones sigue siendo prácticamente la misma en ambos canales en una misma sección.

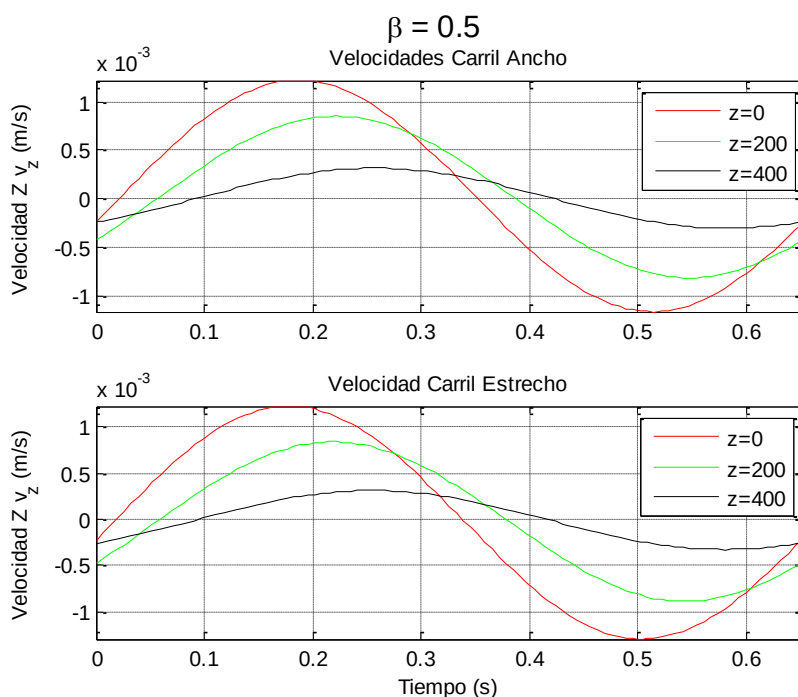


Figura 6.13 Velocidades longitudinales en el punto medio de cada canal para $\beta = 0.5$

Una vez más, en la Figura 6.14 se comparan los caudales obtenidos de la simulación con los caudales que han sido impuestos, volviendo a comprobar que los resultados de la simulación son correctos ya que se reproduce el caudal con una diferencia prácticamente nula.

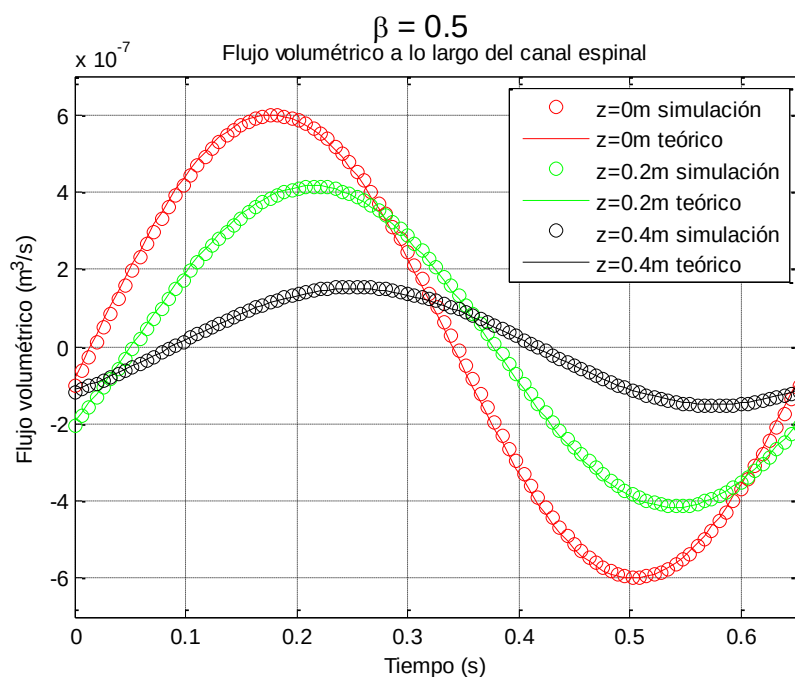


Figura 6.14 Comparación entre los caudales obtenidos mediante simulación y los teóricos para $\beta = 0.5$

Al igual que en los casos anteriores, se han tomado seis puntos equidistantes (figura abajo) en cada uno de los canales con la finalidad de comparar las velocidades estacionarias y oscilatorias en los mismos

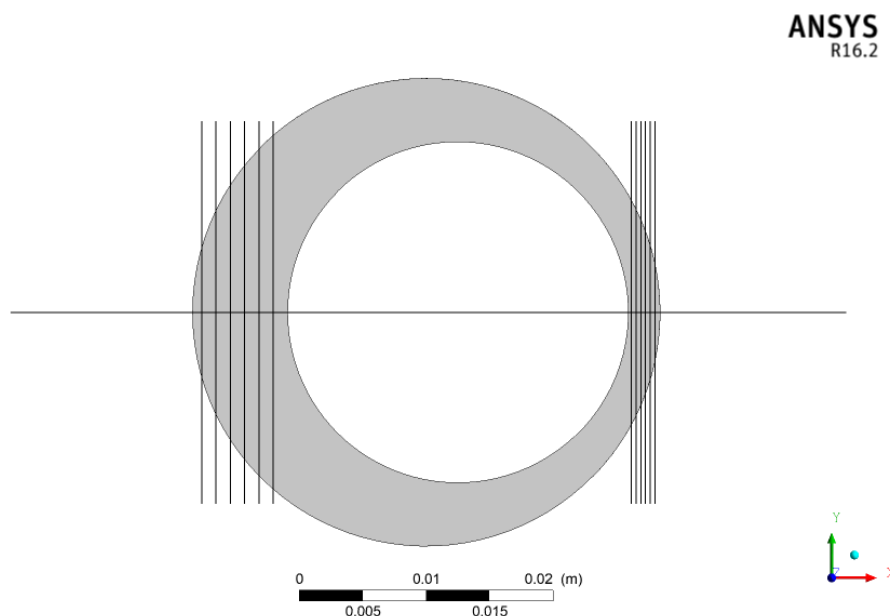


Figura 6.15. Posición de los puntos tomados para representar las velocidades estacionarias.

Punto	X (m)	Y(m)	Z(m)
1	-0.017690	0	0
2	-0.016565		
3	-0.015440		
4	-0.0143150		
5	-0.013190		
6	-0.0120650		
7	0.016190		
8	0.016565		
9	0.016940		
10	0.0173150		
11	0.017690		
12	0.0180650		

Tabla 6.3. Coordenadas de los puntos de la Figura 6.15

En la Figura 6.16 puede verse como las velocidades estacionarias en el canal ancho siguen siendo dos órdenes de magnitud inferiores a las velocidades estacionarias. Además si comparamos dichos valores con los del caso anterior, vemos que la velocidad estacionaria en el canal ancho se va reduciendo ligeramente a medida que se aumenta la excentricidad.

Por otro lado, si se observa la Figura 6.17 las velocidades estacionarias, al igual que en el canal ancho, siguen siendo dos órdenes de magnitud menores que las oscilatorias. Además puede verse como en este caso las velocidades estacionarias en el canal estrecho han aumentado conforme aumenta la excentricidad, al contrario que ha sucedido en el canal ancho.

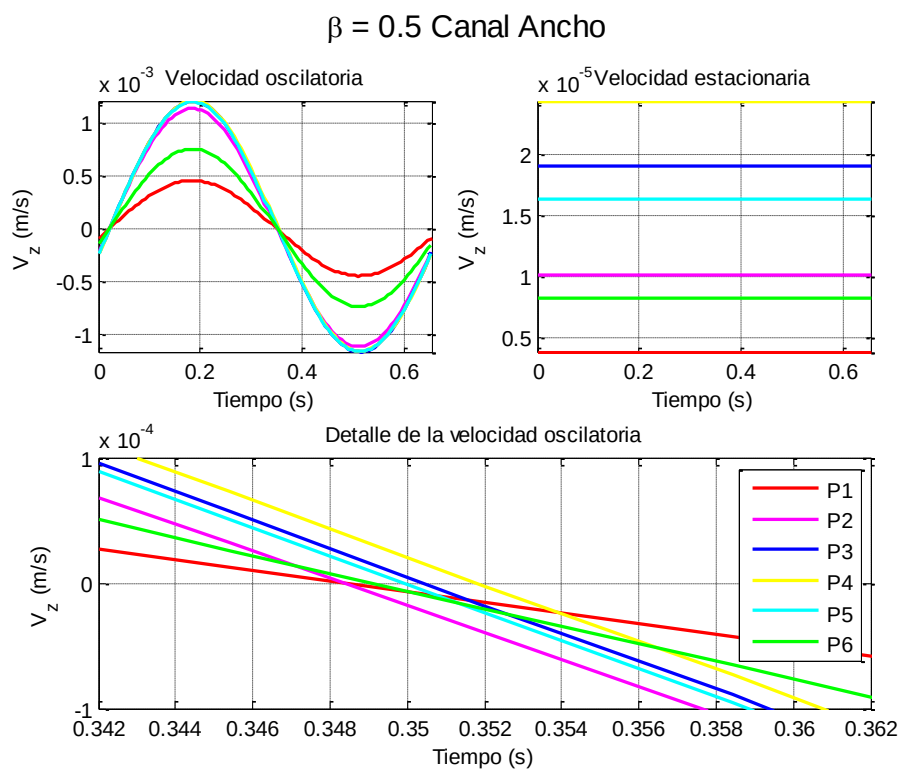


Figura 6.16. Comparativa entre las velocidades oscilatoria y estacionaria en diferentes puntos del canal ancho para $\beta = 0.5$

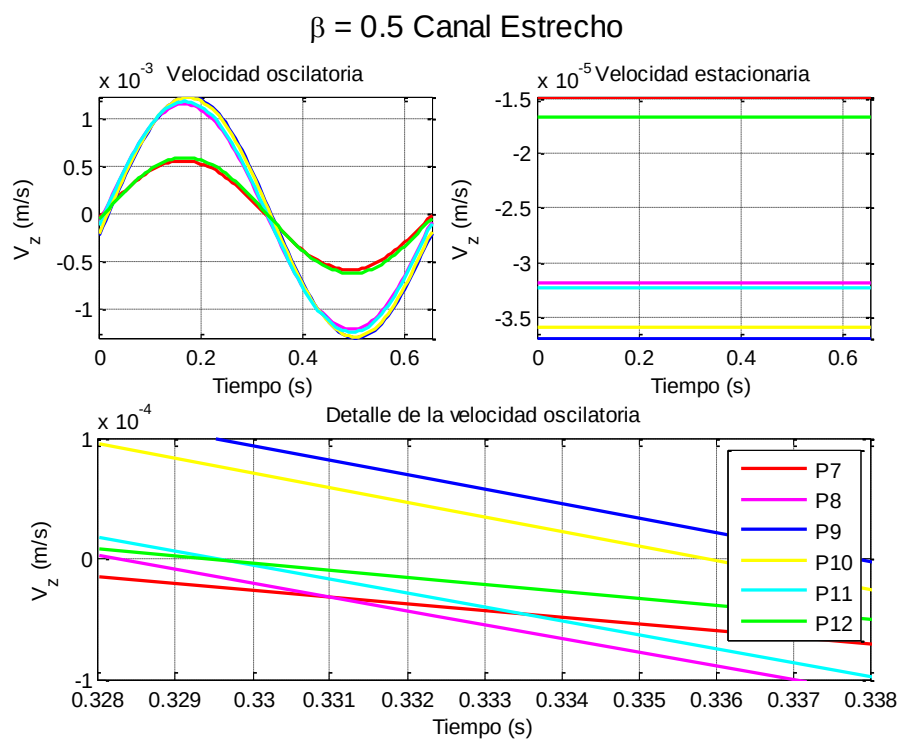


Figura 6.17. Comparativa entre las velocidades oscilatoria y estacionaria en diferentes puntos del canal estrecho para $\beta = 0.5$

Finalmente, se han representado los perfiles de velocidad estacionaria en el canal ancho (Figura 6.18) y en el canal estrecho (Figura 6.19), observándose que, al igual que antes, el canal estrecho presenta velocidades de bajada, siendo los valores pico de éstas mayores que en los casos anteriores, tal y como se ha podido comprobar ya en la Figura 6.17, mientras que el canal ancho presenta velocidades de subida, cuyos valores máximos también son mayores que en el caso anterior.

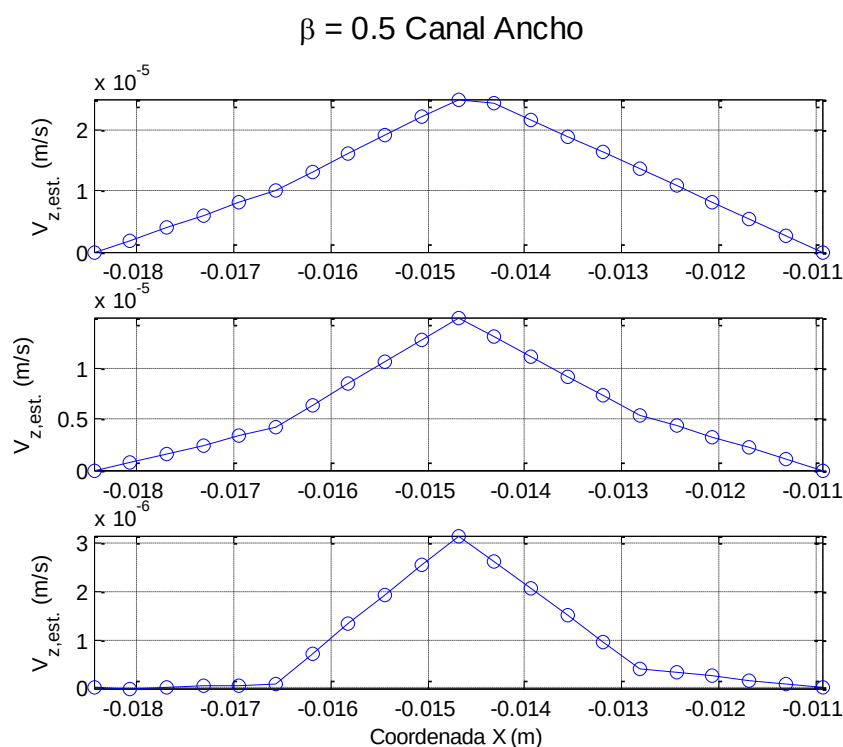


Figura 6.18. Perfiles de velocidad estacionaria en el canal ancho para $\beta = 0.5$

$\beta = 0.5$ Canal Estrecho

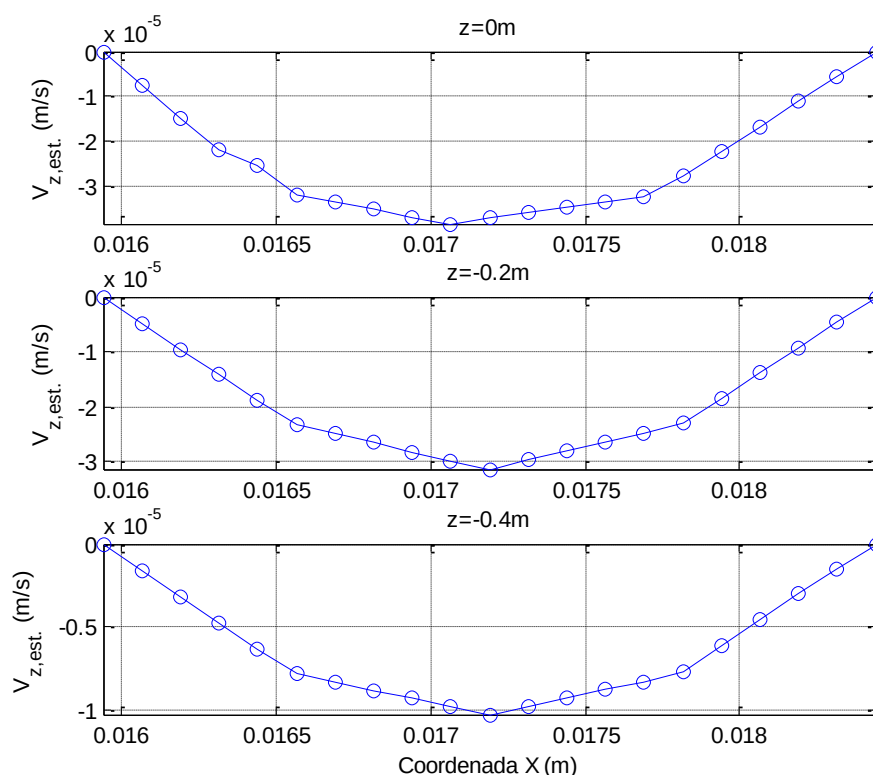


Figura 6.19. Perfiles de velocidad estacionaria en el canal estrecho para $\beta = 0.5$

6.1.4. Caso canónico 4, $\beta=0,75$.

Al igual que para los casos canónicos anteriores se ha representado el caudal obtenido de las simulaciones a diferentes alturas y se ha comparado con el caudal que se ha impuesto en la función, observándose en la **Figura 6.20 Comparación entre los caudales obtenidos mediante simulación y los teóricos para $\beta = 0.75$** que una vez más los valores coinciden con un error prácticamente nulo. Del mismo modo se han vuelto a comparar las velocidades axiales en los puntos medios entre los diámetros de los cilindros en la zona de mayor anchura y mayor estrechamiento (Figura 6.21), observándose que ahora las velocidades entre los dos puntos situados a la misma altura se encuentran más desfasadas que en el caso anterior, esto se debe a que, a diferencia de los casos anteriores, para el caso de valores de β elevados, la velocidad en el canal estrecho no se produce en el punto de mayor

estrechamiento si no que ésta se produce en las proximidades de dicho estrechamiento (Figura 6.27)

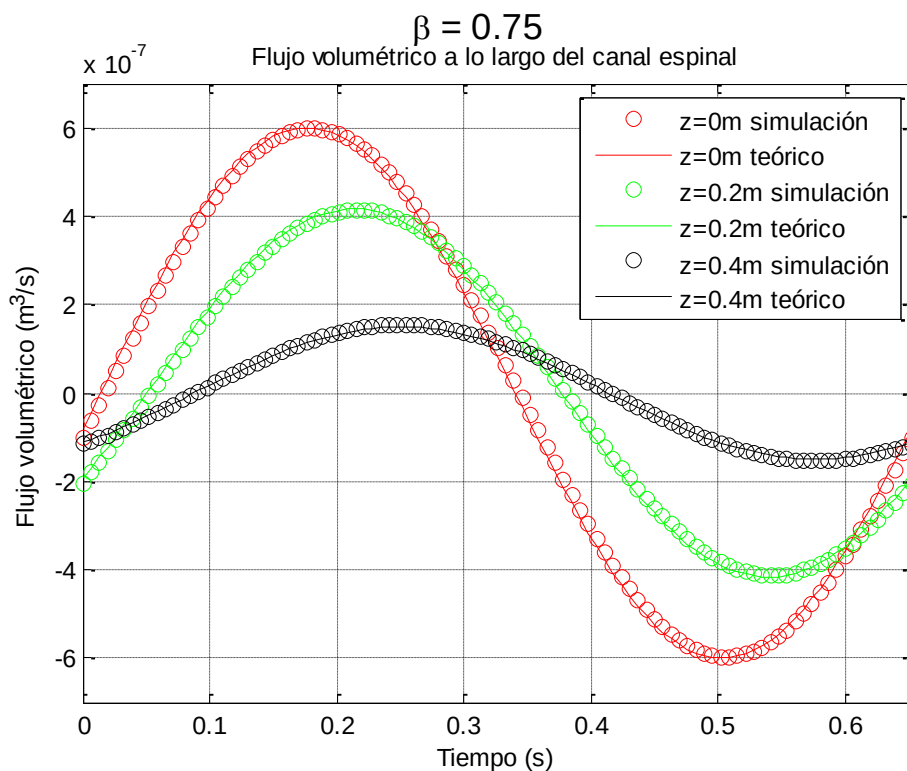


Figura 6.20 Comparación entre los caudales obtenidos mediante simulación y los teóricos para $\beta = 0.75$

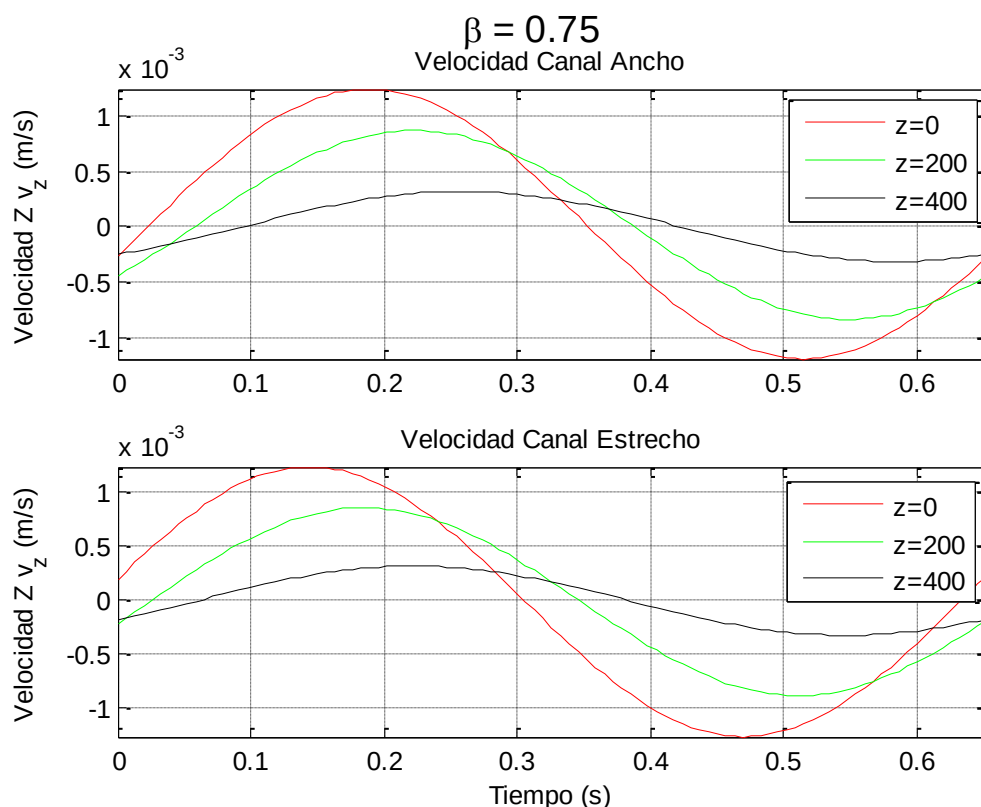


Figura 6.21. Velocidades longitudinales en el punto medio de cada canal para $\beta = 0.75$

En cuanto a los fenómenos del régimen estacionario, para comparar una vez más los valores de la velocidad estacionaria con los de la velocidad oscilatoria se han tomado de nuevo seis puntos equidistantes en cada uno de los canales en una misma sección, los cuales pueden verse en la Figura 6.22

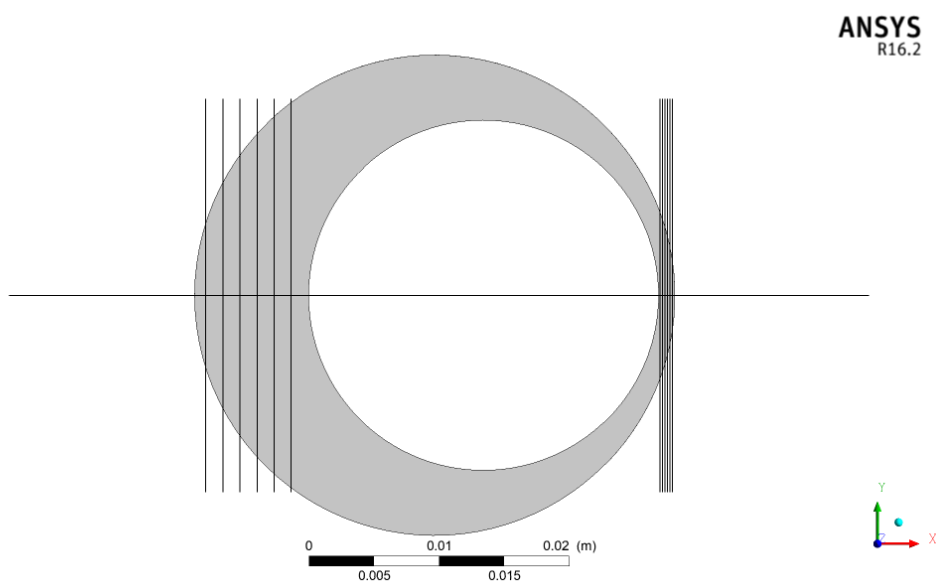


Figura 6.22. Puntos tomados para representar la velocidad estacionaria.

Punto	X (m)	Y(m)	Z(m)
1	-0.0175650	0	0
2	-0.0162525		
3	-0.01494		
4	-0.0136275		
5	-0.012315		
6	-0.0110025		
7	0.017315		
8	0.0175025		
9	0.01769		
10	0.0178775		
11	0.01794		
12	0.0182525		

Tabla 6.4 Coordenadas de los puntos de la Figura 6.22

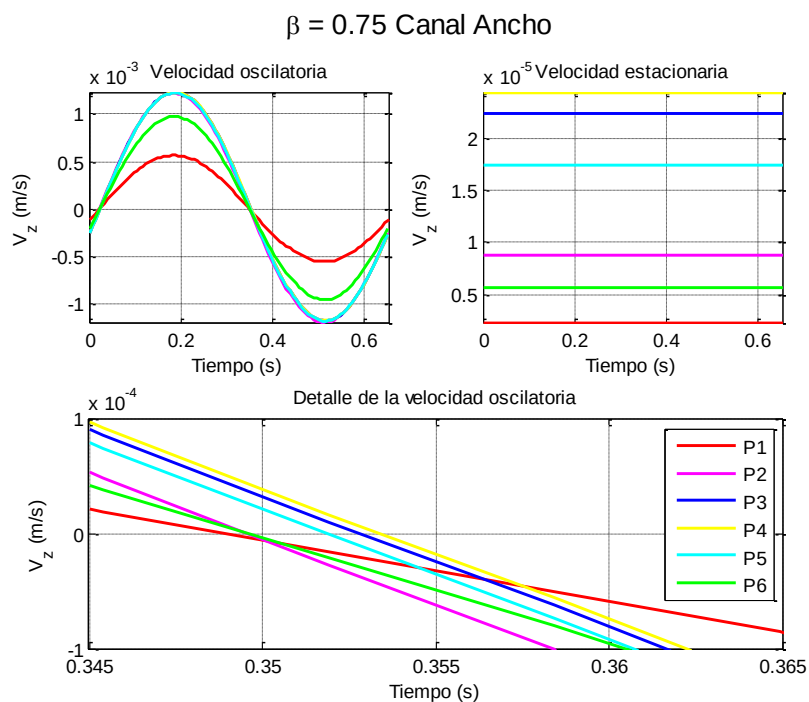


Figura 6.23. Comparativa entre las velocidades oscilatoria y estacionaria en diferentes puntos del canal ancho para $\beta = 0.75$

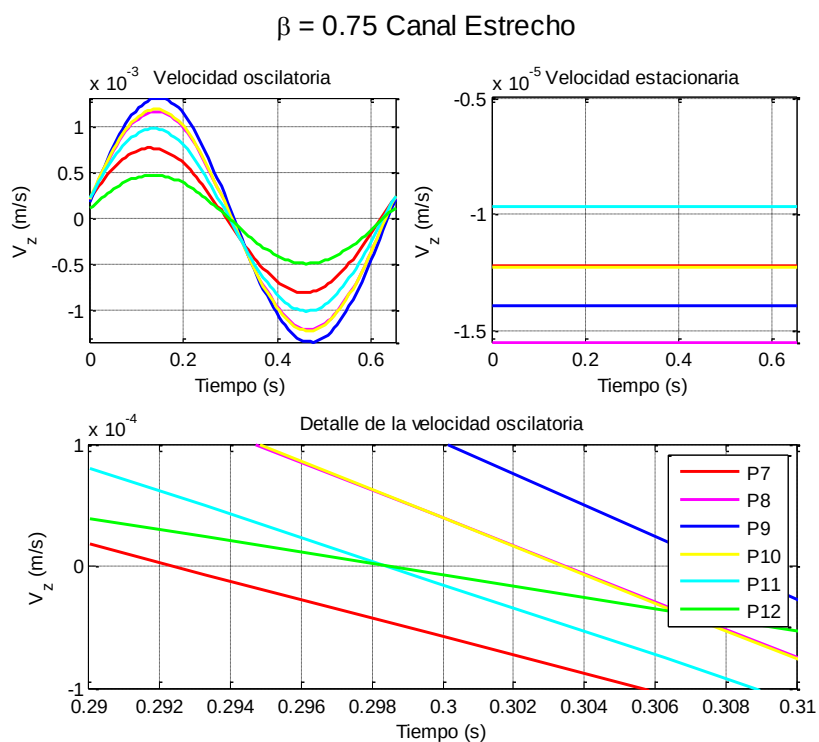


Figura 6.24 Comparativa entre las velocidades oscilatoria y estacionaria en diferentes puntos del canal estrecho para $\beta = 0.75$

Al igual que en casos anteriores, las velocidades estacionarias siguen teniendo dos órdenes de magnitud menos que las velocidades transitorias. A diferencia de los casos anteriores donde, en los puntos en los que se han monitorizado éstas velocidades en el canal estrecho, en el caso de $\beta = 0.75$ las velocidades estacionarias en dichos puntos no son máximas y por tanto, el perfil de velocidades estacionarias no tendrá valores máximos superiores a los de casos anteriores en éstos puntos, ya que como se ha mencionado, las velocidades máximas en éste caso se producen en las proximidades del estrechamiento máximo y no en éste (Figura 6.27).

Por otro lado, en la Figura 6.25 podemos ver que el perfil de velocidades estacionarias en el canal ancho sí que presenta unos valores ligeramente superiores a los que se podían apreciar para el caso anterior.

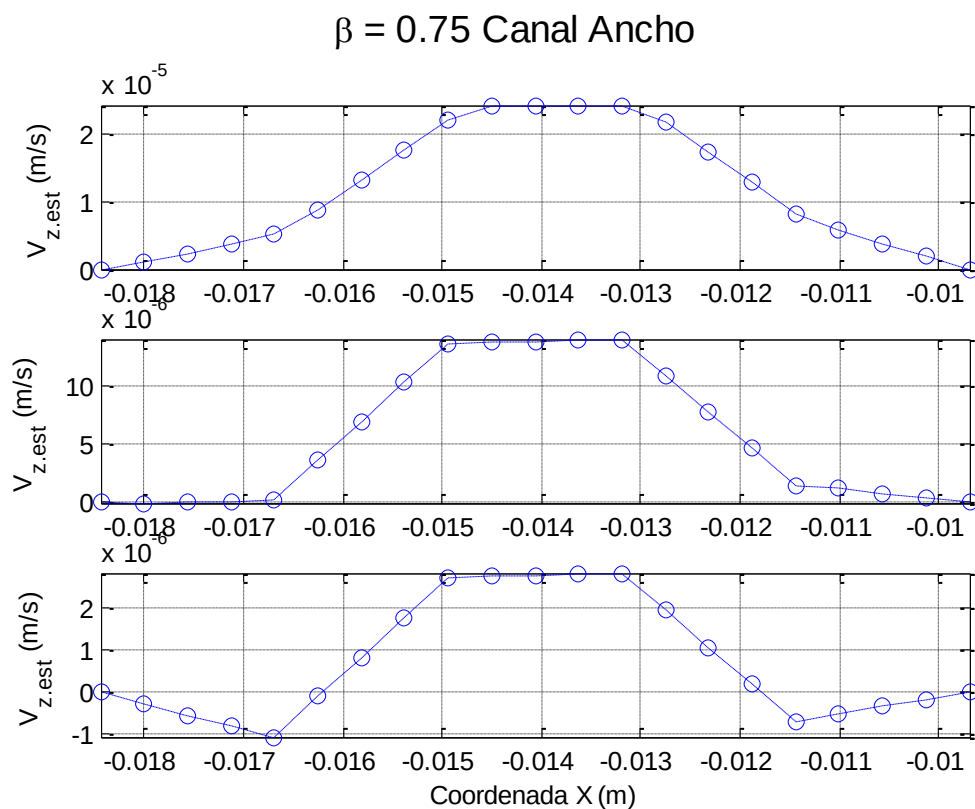


Figura 6.25. Perfiles de velocidad estacionaria en el canal ancho para el caso $\beta = 0.75$

$\beta = 0.75$ Canal Estrecho

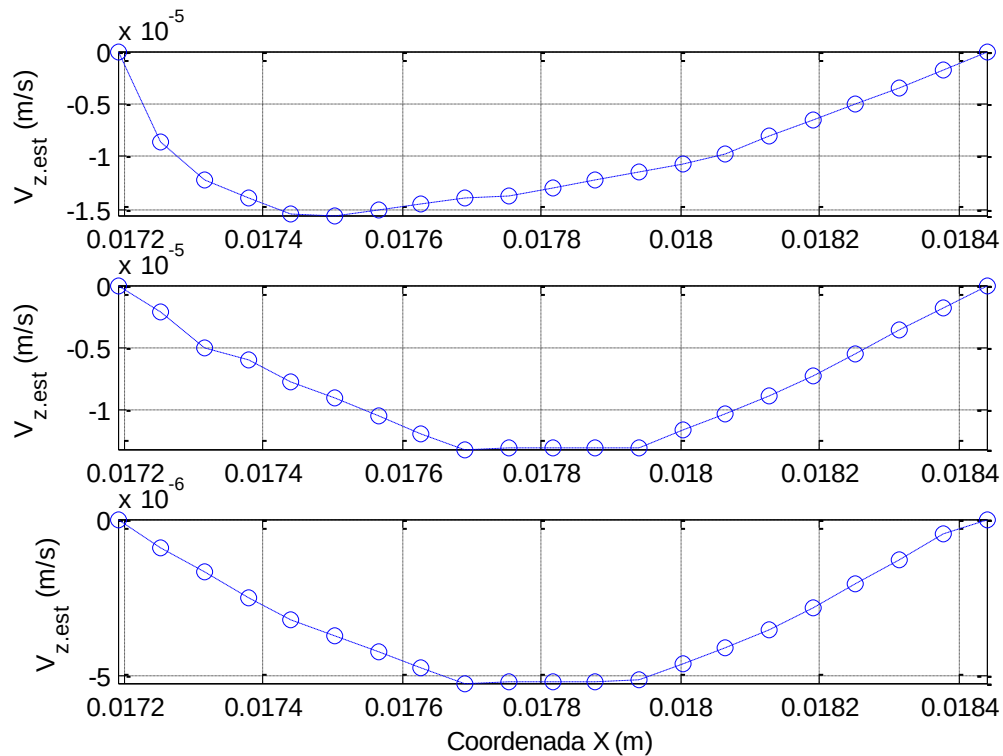


Figura 6.26 . Perfil de velocidades estacionarias en el canal estrecho para el caso de $\beta = 0.75$

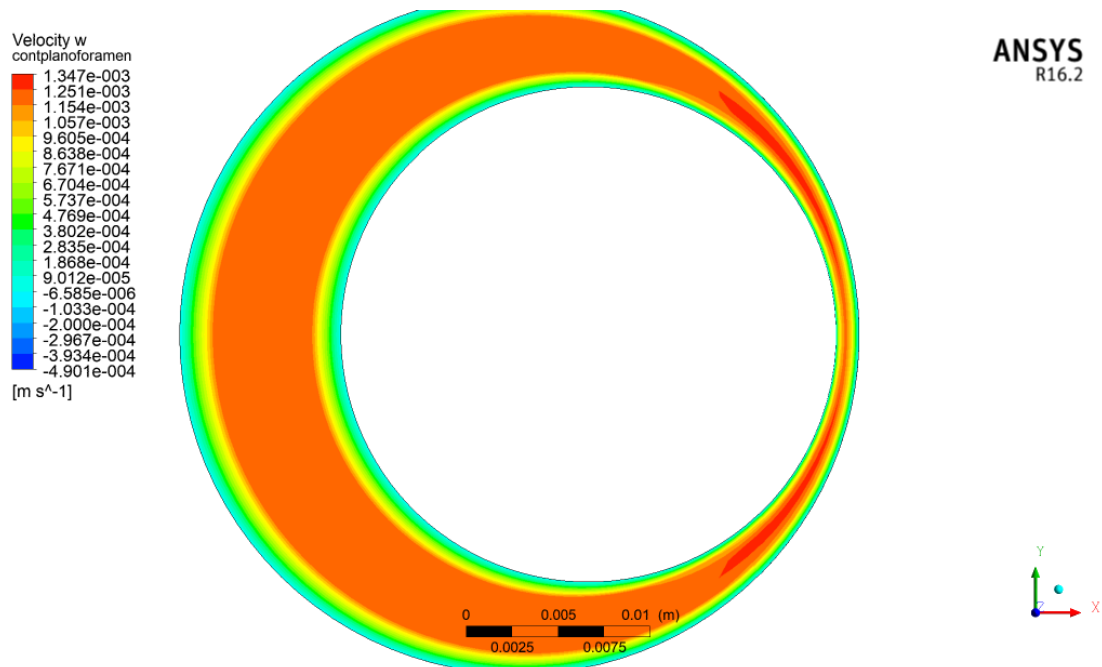


Figura 6.27. Contorno de velocidades axiales para $\beta = 0.75$

6.1.2. Resultados del caso con caudal realista.

Para finalizar las simulaciones en la geometría simplificada se ha configurado una simulación con el caudal realista obtenido en el apartado [apartado c. real] con el objetivo de comprobar si, independientemente de la forma de la onda del caudal impuesto, el modelo es capaz de reproducirlo de manera correcta.

Del mismo modo, se obtendrán datos sobre la velocidad estacionaria en diferentes alturas tanto para el canal ancho como para el estrecho con el objetivo de comprobar si tienen lugar los mismos fenómenos que en los casos en los que se utilizaban caudales senoidales.

En la Figura 6.28 puede observarse que, del mismo modo que en los casos anteriores, el caudal obtenido en las simulaciones es prácticamente el mismo que el impuesto, concluyendo que, para una misma geometría el modelo reproduce fielmente el caudal impuesto, independientemente de la forma que éste presente.

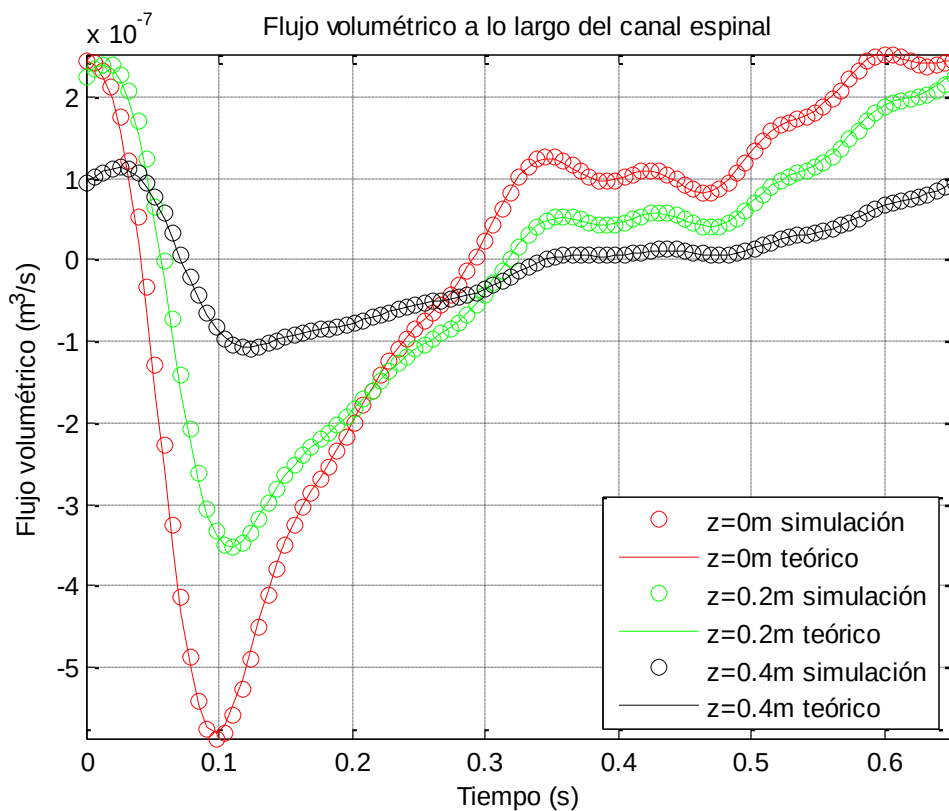


Figura 6.28. Comparación de los caudales impuestos y obtenidos de la simulación.

Por otro lado, en la Figura 6.29 puede comprobarse que, al igual que sucede con el caso de caudal senoidal con $\beta = 0.25$, apenas existe desfase entre las velocidades máximas en el canal ancho y el estrecho, y que las velocidades máximas en el canal estrecho son ligeramente superiores a las del canal ancho, ya que por continuidad si el flujo que atraviesa la sección es el mismo para cualquier geometría, en las zonas donde el estrechamiento sea mayor la velocidad debe ser mayor también para compensar dicha disminución de la sección.

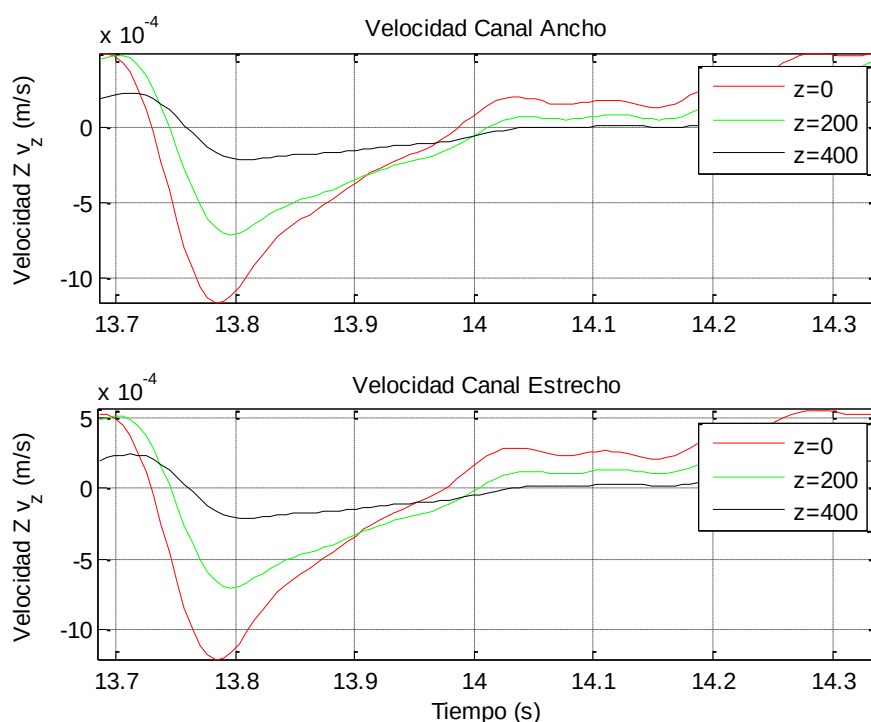


Figura 6.29. Velocidades longitudinales en el punto medio de cada canal para el caudal real.

Finalmente, se han representado las velocidades estacionarias en ambos canales a diferentes alturas.

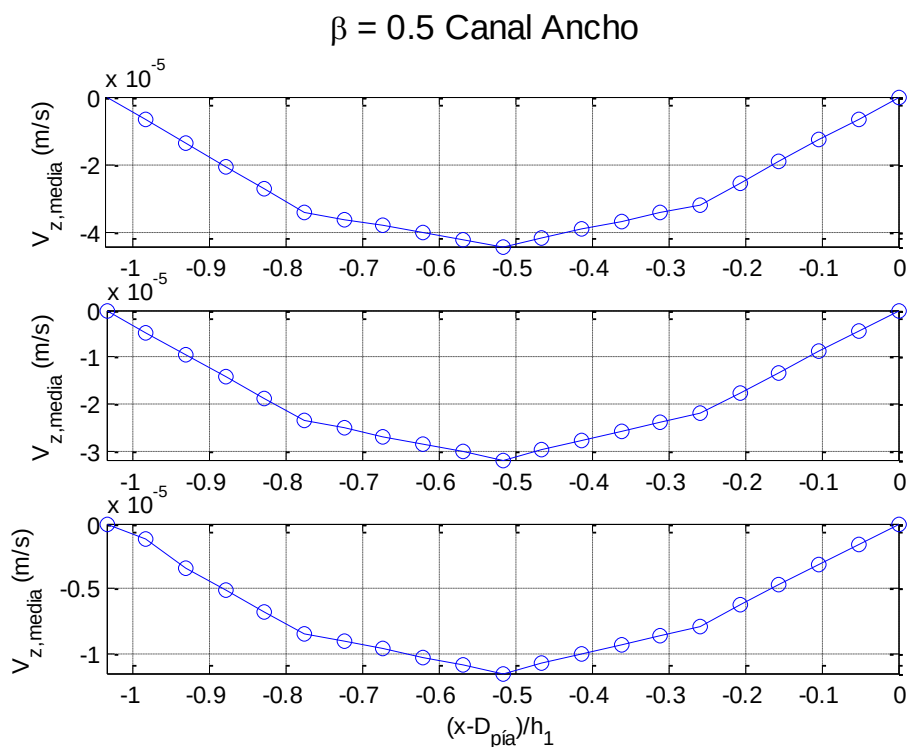


Figura 6.30. Perfil de velocidades en el canal ancho para la simulación con caudal real.

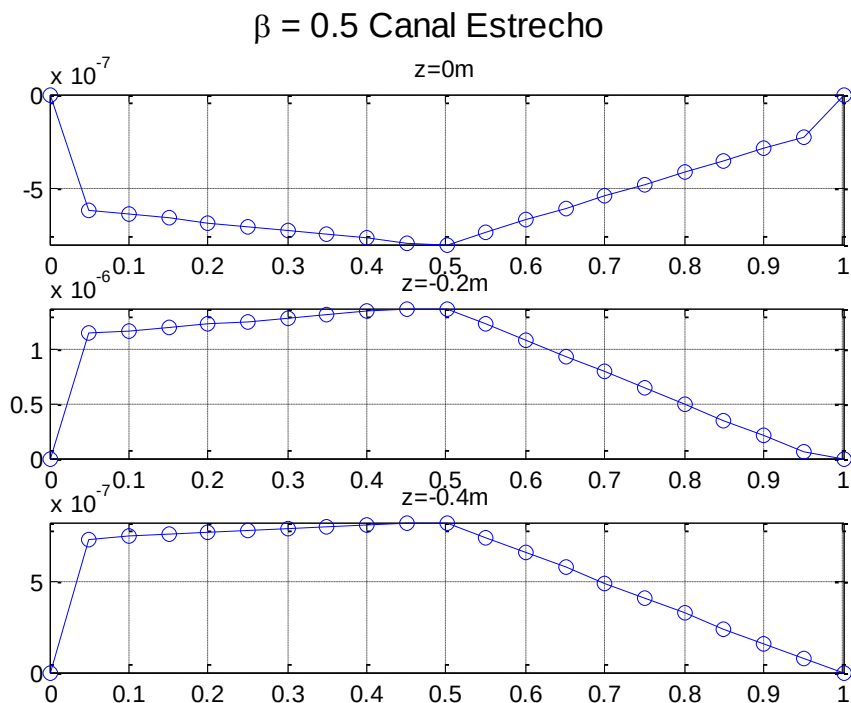


Figura 6.31. Perfil de velocidades en el canal estrecho para la simulación con caudal real.

En la Figura 6.30 puede observarse como para el caso del canal ancho la velocidad estacionaria es tan solo un orden de magnitud inferior a la oscilatoria,

siendo a lo largo de éste el perfil de velocidades negativo, y por tanto de bajada, mientras que en el caso del canal estrecho, como muestra la Figura 6.31, las velocidades estacionarias son tres órdenes de magnitud menores que las oscilatorias y aparecen un perfil de velocidades de bajada en la parte superior de la malla y de subida en secciones más bajas.

Debido a éstas diferencias puede concluirse que, en el caso del caudal real, al no tener un desplazamiento neto nulo en la malla, ya que el área positiva que encierra la curva de caudal es diferente al área negativa (por lo que si sumamos todos los desplazamientos que tienen lugar en una determinada sección para un ciclo cardíaco completo tendremos un desplazamiento resultante), no llegará a un estado estacionario y por tanto, no reproducirá dicho fenómeno correctamente.

6.2. Geometría simplificada con caudal real.

Finalmente, una vez se ha comprobado que, para una geometría simple y sea cual sea la excentricidad de la piamadre respecto de la duramadre, el caudal que reproduce el modelo a lo largo del canal espinal es el mismo, comprobaremos que éste será también igual para geometrías complejas, como la geometría realista obtenida en el apartado 3.2.

Para ello, del mismo modo que en el resto de casos que han sido simulados, se comparará el caudal en el foramen magnum y en secciones a 200 y 400 mm del foramen magnum.

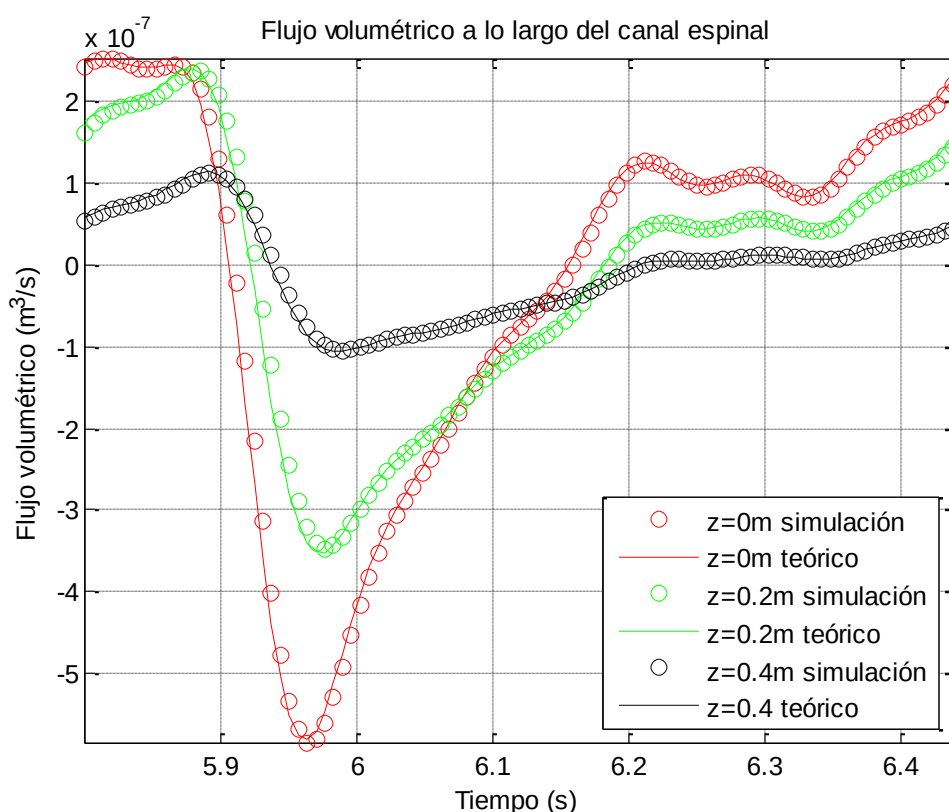


Figura 6.32. Comparación de los caudales obtenidos de simulación e impuestos en una geometría realista y caudal real.

En la Figura 6.32 puede verse como el caudal se reproduce de manera correcta, al igual que en el resto de las simulaciones por lo que se puede decir que, independientemente de la forma que presente la onda del caudal y de la complejidad de la geometría que se éste simulando, el caudal obtenido de la misma es exactamente igual al impuesto en la UDF.

Finalmente, se han representado las velocidades en diferentes puntos a diferentes alturas de la misma forma que en los apartados anteriores con el fin de observar como varían las velocidades estacionarias en estos.

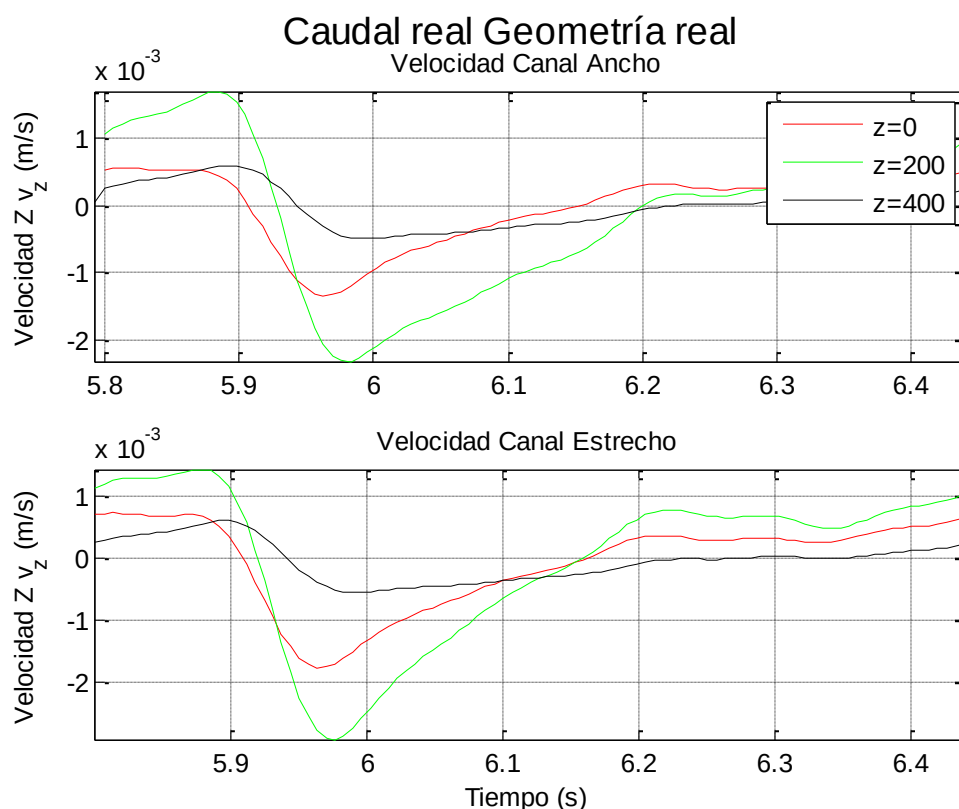


Figura 6.33. Velocidades longitudinales en el punto medio de cada canal para un caudal real y geometría realista.

De acuerdo a los resultados obtenidos en el apartado 6.1.2 no se han representado valores de velocidad estacionaria ya que vemos que, en el caso de dicho caudal, no puede alcanzarse el estado estacionario y por tanto no se reproducen de manera correcta dichos fenómenos.

7. CONCLUSIONES.

En primer lugar ha de destacarse la utilidad que presenta Matlab tanto como herramienta de post-procesado como de pre-procesado ya que ha permitido realizar los ajustes necesarios a los datos obtenidos de manera que han podido realizarse interpolaciones de funciones que han hecho posible la implementación del código de las diferentes UDFs utilizadas a lo largo del proyecto.

Por otro lado, se ha comprobado el potencial que presentan las UDF para implementar casos no típicos en ingeniería, ya que han podido imponerse condiciones de malla móvil para diferentes geometrías, tanto reales como complejas, del mismo modo que éstas permiten realizar ajustes de parámetros, cálculo de determinadas variables, etc, de manera que conociendo las diferentes macros puede configurarse una gran variedad de modelos que no son posibles de configurar si se hace sin éstas.

En cuanto a los resultados, vemos que el caudal impuesto en los diferentes casos es reproducido por el modelo de manera correcta, independientemente de la forma que presenta la onda de caudal como de la complejidad que presenta la geometría, por lo que podemos decir que el movimiento continuo de las paredes de la médula conseguido mediante las UDFs implementadas reproduce los caudales de manera más exacta que en estudios previos.

Por otro lado, teniendo en cuenta las geometrías simplificadas y la variación del parámetro β , puede observarse como, a medida que se incrementa el valor de éste se produce un desfase en las velocidades de ambos canales, como puede observarse en la Figura 7.1

Como se ha comentado a lo largo del trabajo, se han estudiado los fenómenos de la velocidad estacionaria la cual está relacionada con la recirculación del CSF y que por tanto es de gran interés para el transporte de fármacos en dicho fluido, en la geometría sencilla ya que presentaba una forma más sencilla y por tanto, en ella la sería más fácil captarlos, para diferentes valores de β , llegando a la conclusión de que, para los casos en los que se ha utilizado un caudal senoidal:

- En el canal ancho el perfil de velocidades toma valores positivos, con pequeñas zonas negativas en las proximidades de la pared, aumentando los valores máximos de los mismos a medida que aumenta el valor de β .
- En el canal negativo, por el contrario, el perfil de velocidades presenta valores negativos y por tanto el flujo irá desde la entrada hasta la base de la malla.
- Por otro lado, se ha observado que, para el modelo implementado y la geometría utilizada, para valores de β elevados, las velocidades máximas en el canal estrecho comienzan a aparecer en las proximidades de la zona de estrechamiento máximo, pero no en ella, a diferencia de los casos en los que β es menor.

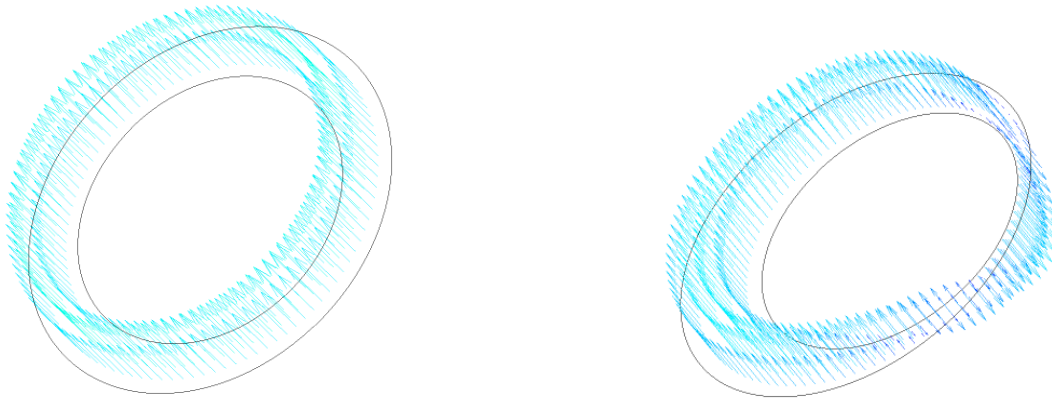


Figura 7.1. Comparación entre los perfiles de velocidad para dos casos con diferente valor de β en un mismo instante.

Por otro lado, se ha intentado reproducir lo mismo para un caudal realista en la geometría simplificada con un valor $\beta = 0.5$, llegando a la conclusión de que:

- Tanto en el canal ancho como en el estrecho los perfiles de velocidad estacionaria obtenidos presentan sentidos de bajada o negativos y presentan un orden de magnitud menos que la velocidad oscilatoria, cuando de estudios realizados previamente se conoce que ésta es dos órdenes de magnitud menor.
- El hecho de obtener un desplazamiento neto no nulo de las paredes de la malla a lo largo de un ciclo cardíaco puede provocar que dicha simulación

no alcance el estado estacionario, por lo que se hace necesario implementar una función que tenga en cuenta que dichos desplazamientos deben tener un valor neto nulo.

Cabe resaltar que sería interesante para trabajos futuros la implementación de código de UDFs que consideren variaciones del perímetro de la duramadre, con la finalidad de obtener resultados aún más precisos ya que, al producir desplazamientos de la pared de la misma, aunque muy pequeños, y no considerarlos estamos introduciendo un mínimo error. Así mismo, como se ha comentado, obtener una señal de caudal real tal que el desplazamiento neto de las paredes de la malla sea cero, permitiendo así comprobar que ocurre cuando el problema con dicho caudal alcanza el estado estacionario.

Bibliografía

- [1] [En línea]. Available:
<https://www.monografias.com/trabajos93/meninges/meninges.shtml>.
- [2] F. Peric, Computational Methods for Fluid Dynamics, 2002.
- [3] P. Bohorquez Rodriguez de Medina, «Volúmenes Finitos,» Jaén, 2010.
- [4] ANSYS INC, «ANSYS FLUENT User's Guide,» 2010.
- [5] M. K. C. G. J. N. O. G. R. S. J. R. B. A. M. Tao Xing, «Nonuniform Moving Boundary Method for Computational Dynamics Simulation of Intrathecal Cerebrospinal Flow Distribution in a Cynomolgus Monkey,» Agosto 2017.
- [6] Y. H. D. C. Z. A. A. L. Kevin M. Tangen, «CNS wide simulation of flow resistance and drug transport due to spinal microanatomy,» 15 Febrero 2015.
- [7] «WebPlotDigitizer,» [En línea]. Available: <https://automeris.io/WebPlotDigitizer/>.
- [8] R. Bautista González, Simulación del flujo oscilatorio del líquido cefalorraquídeo en el canal espinal en geometría realista, 2019.
- [9] Kuttler, Andrea, «Understanding pharmacokinetics using realistic computational models of fluid dynamics: biosimulation of drug distribution within the CSF space for intrathecal drugs »
- [10] Helgeland Anders, «Numerical simulations of the pulsating flow of cerebrospinal fluid flow in the cervical spinal canal of Chiari patient.»
- [11] Mohammed Khani, Lucas R. Sass, «A 3D subject-specific model of the spinal subarachnoid space with anatomically realistic ventral and dorsal spinal cord nerve rootles»

APÉNDICES.

APÉNDICE A. CÓDIGOS DE LAS UDFs

```
#include "udf.h"
#define timestep 0.006517
#define r 0.01844

DEFINE_GRID_MOTION(sin_wave, domain, dt, time, dtime)
{
    Thread *tf = DT_THREAD((Dynamic_Thread *)dt);
    face_t f;
    Node *v;
    real dis;
    real dgdz;
    real NV_VEC(axis);
    real A[ND_ND];
    real AMag;
    int n;

    SET_DEFORMING_THREAD_FLAG(THREAD_T0(tf));

    begin_f_loop(f, tf)
    {
        f_node_loop(f, tf, n)
        {
            v = F_NODE(f, tf, n);
            F_AREA(A, f, tf);
            AMag = NV_MAG(A);
            A[0] = A[0] / AMag;
            A[1] = A[1] / AMag;
            A[2] = A[2] / AMag;

            NV_D(axis, =, A[0], A[1], A[2]);

            dgdz = -1.03638 * pow(10, -
6) * (1 + 1.989 * NODE_Z(v) + 6.274 * pow(NODE_Z(v), 2) + 25.31 * pow(NODE_Z(v), 3) + 26.26 *
pow(NODE_Z(v), 4)) * cos(0.1378 - 9.64122 * CURRENT_TIME
-
1.7473 * NODE_Z(v)) + 6 * pow(10, -
7) * (1.989 + 12.548 * NODE_Z(v) + 75.93 * pow(NODE_Z(v), 2) + 105.04 * pow(NODE_Z(v), 3))
* sin(0.1378 - 9.64122 * CURRENT_TIME - 1.7473 * NODE_Z(v));

            if (NODE_POS_NEED_UPDATE(v))
            {
                NODE_POS_UPDATED(v);

                dis = ((dgdz) / (2 * M_PI * r) * timestep);
            }
        }
    }
}
```



```

        NV_V_VS(NODE_COORD(v),=,NODE_COORD(v),+,axis,*,dis);

    }
}
Update_Face_Metrics(f,tf);
}
end_f_loop(f,tf);
}

```

```

#include "udf.h"
#define freq 1
#define tstep 0.006517 /*Paso de tiempo de la simulación [s]*/

DEFINE_GRID_MOTION(Sistole_Diastole, domain, dt, time, dtime)
{
    Thread *tf = DT_THREAD((Dynamic_Thread *)dt);
    face_t f;
    Node *v;
    real displ;
    real NV_VEC(axis);
    real A[ND_ND]; /* face area normal vector */
    real AMag; /* face area magnitude */
    int n;
    real Ca;
    real Q;
    real phi;
    real dQ_zt; /*Derivada parcial de la señal Q(t,z) respecto a z*/

    real cgx; /*Coordenada en X del centro geométrico del
    contorno*/
    real cgy; /*Coordenada en Y del centro geométrico del
    contorno*/
    real cgz; /*Coordenada en Z del centro geométrico del
    contorno*/
    real NV_VEC(cg); /*Vector que guarda las coordenadas del centro
    geométrico*/
    real NV_VEC(cg2D); /*Vector que representa la proyección de "cg"
    en el plano XY*/
    real NV_VEC(riprim); /*Vector posición de los nodos respecto al centro
    geométrico cg*/
    real NV_VEC(riprim2D); /*Vector que representa la proyección de "riprim"
    en el plano XY*/
    real riprimag; /*Módulo de "riprim"*/
    real riprimag2D; /*Módulo de "riprim2D"*/
    real h; /*Incremento de altura de cada nodo,
    que se aproximará al circundiámetro de cada elemento triangular*/
    real perim; /*Perímetro de la duramadre en función de la
    coordenada axial sacada de los datos de resonancias*/

```

```

SET_DEFORMING_THREAD_FLAG(THREAD_T0(tf));

begin_f_loop(f, tf)

{
    f_node_loop(f,tf,n)
    {
        v=F_NODE(f,tf,n);

        cgx=0; /*Introducir función de la coordenada en X del centro
geométrico en función de Z [m]*/
        cgy=0; /*Introducir función de la coordenada en Y del centro
geométrico en función de Z [m]*/

        NV_D(cg,=,cgx,cgy,NODE_Z(v));
        NV_VV(riprim,=,NODE_COORD(v),-,cg);
        NV_D(riprim2D,=,riprim[0],riprim[1],0);
        riprimag=NV_MAG(riprim);

        F_AREA(A,f,tf); /*A is a vector, f face index, t thread
index */
        AMag=NV_MAG(A); /* Face normal vector */
        A[0]=A[0]/AMag; /* X component of vector */
        A[1]=A[1]/AMag; /* Y component of vector */
        A[2]=A[2]/AMag; /* Z component of vector */

        NV_D(axis, =, A[0], A[1], A[2]);
        perim=0.09298*sin(5.46*NODE_Z(v)+2.832) +
0.05481*sin(10.54*NODE_Z(v)+0.9754) + 0.009569*sin(33.92*NODE_Z(v)+2.319)
+ 0.01053*sin(20.21*NODE_Z(v)+0.2468) +
0.008196*sin(39.18*NODE_Z(v)+0.4447) +
0.003179*sin(61.34*NODE_Z(v)+0.5976) +
0.002383*sin(70.86*NODE_Z(v)+0.02949) +
0.002508*sin(50.93*NODE_Z(v)+0.5606);
        dQ_zt=-(1 + 1.989*NODE_Z(v) + 6.274 *pow(NODE_Z(v),2) + 25.31
*pow(NODE_Z(v),3) + 26.26 *pow(NODE_Z(v),4) )*(1.10772*pow(10,-8)
*cos(0.1378 - 77.136*CURRENT_TIME - 1.7273 *NODE_Z(v)) + 1.85167*pow(10,-
8) *cos(0.1378 - 67.494*CURRENT_TIME - 1.7273 *NODE_Z(v)) +
4.23879*pow(10,-8) *cos(0.1378 - 57.852 *CURRENT_TIME - 1.7273 *NODE_Z(v))
+ 5.3719*pow(10,-8) *cos(0.1378 - 48.21 *CURRENT_TIME - 1.7273 *NODE_Z(v))
+ 4.13343*pow(10,-8) *cos(0.1378 - 38.568 *CURRENT_TIME - 1.7273
*NODE_Z(v)) - 6.8712*pow(10,-8) *cos(0.1378 - 28.926 *CURRENT_TIME -
1.7273 *NODE_Z(v)) - 2.25067*pow(10,-7) *cos(0.1378 - 19.284 *CURRENT_TIME
- 1.7273 *NODE_Z(v)) - 4.45643*pow(10,-7) *cos(0.1378 - 9.642
*CURRENT_TIME - 1.7273 *NODE_Z(v)) - 5.67936*pow(10,-9) *sin(0.1378 -
77.136 *CURRENT_TIME - 1.7273 *NODE_Z(v)) - 2.87595*pow(10,-8) *sin(0.1378
- 67.494 *CURRENT_TIME - 1.7273 *NODE_Z(v)) + 6.06973*pow(10,-9)
*sin(0.1378 - 57.852 *CURRENT_TIME - 1.7273 *NODE_Z(v)) + 3.1005*pow(10,-
8) *sin(0.1378 - 48.21 *CURRENT_TIME - 1.7273 *NODE_Z(v)) +

```

```

1.03465*pow(10,-7) *sin(0.1378 - 38.568 *CURRENT_TIME - 1.7273 *NODE_Z(v))
+ 1.27751*pow(10,-7) *sin(0.1378 - 28.926 *CURRENT_TIME - 1.7273
*NODE_Z(v)) + 2.03476*pow(10,-7) *sin(0.1378 - 19.284 *CURRENT_TIME -
1.7273 *NODE_Z(v)) - 6.69156*pow(10,-8) *sin(0.1378 - 9.642 *CURRENT_TIME
- 1.7273 *NODE_Z(v))) - (1.989 + 12.548 *NODE_Z(v) + 75.93
*pow(NODE_Z(v),2) + 105.04 *pow(NODE_Z(v),3))*(-1.274*pow(10,-8) -
3.288*pow(10,-9) *cos(0.1378 - 77.136 *CURRENT_TIME - 1.7273 *NODE_Z(v)) -
1.665*pow(10,-8) *cos(0.1378 - 67.494 *CURRENT_TIME - 1.7273 *NODE_Z(v)) +
3.514*pow(10,-9) *cos(0.1378 - 57.852 *CURRENT_TIME - 1.7273 *NODE_Z(v)) +
1.795*pow(10,-8) *cos(0.1378 - 48.21 *CURRENT_TIME - 1.7273 *NODE_Z(v)) +
5.99*pow(10,-8) *cos(0.1378 - 38.568 *CURRENT_TIME - 1.7273 *NODE_Z(v)) +
7.396*pow(10,-8) *cos(0.1378 - 28.926 *CURRENT_TIME - 1.7273 *NODE_Z(v)) +
1.178*pow(10,-7) *cos(0.1378 - 19.284 *CURRENT_TIME - 1.7273 *NODE_Z(v)) -
3.874*pow(10,-8) *cos(0.1378 - 9.642 *CURRENT_TIME - 1.7273 *NODE_Z(v)) -
6.413*pow(10,-9) *sin(0.1378 - 77.136 *CURRENT_TIME - 1.7273 *NODE_Z(v)) -
1.072*pow(10,-8) *sin(0.1378 - 67.494 *CURRENT_TIME - 1.7273 *NODE_Z(v)) -
2.454*pow(10,-8) *sin(0.1378 - 57.852 *CURRENT_TIME - 1.7273 *NODE_Z(v)) -
3.11*pow(10,-8) *sin(0.1378 - 48.21 *CURRENT_TIME - 1.7273 *NODE_Z(v)) -
2.393*pow(10,-8) *sin(0.1378 - 38.568 *CURRENT_TIME - 1.7273 *NODE_Z(v)) +
3.978*pow(10,-8) *sin(0.1378 - 28.926 *CURRENT_TIME - 1.7273 *NODE_Z(v)) +
1.303*pow(10,-7) *sin(0.1378 - 19.284 *CURRENT_TIME - 1.7273 *NODE_Z(v)) +
2.58*pow(10,-7) *sin(0.1378 - 9.642 *CURRENT_TIME - 1.7273 *NODE_Z(v)));
/*Esta derivada tiene como señal de caudal la correspondiente
al Foramen Magnum y con una altura de médula de 604 mm aproximadamente*/

    if (NODE_POS_NEED_UPDATE(v))
    {
        NODE_POS_UPDATED(v);
        displ=((dQ_zt)/(perim))*tstep;

        NV_V_VS(NODE_COORD(v), =, NODE_COORD(v), +,
axis,*,displ);
    }
    Update_Face_Metrics(f, tf);
}
end_f_loop(f,tf);
}

```

APÉNDICE B. CÓDIGOS DE POST PROCESADO EN MATLAB

```

%%CONDICIONES CONTORNO PROBLEMA CON CAUDAL REAL

format long
clear all
close all

tend=0.6517;
zend=-0.6;

```

```
load('Q(t)- FM.mat');

t=t/1000;
puntos=100;
T=t(length(t));

tinterp=(0:(T/puntos):T);
Q_interp= interp1(t,Q,tinterp)/10^6;
figure(1)
plot(tinterp,Q_interp,'k',t,Q/10^6,'go')
xlabel('Tiempo(s)');
ylabel('Flujo volumétrico(ml/s)')
title('Flujo Volumétrico en Foramen Magnum')
legend('Función interpolada','Datos paper')
axis tight

%INTERPOLAMOS LA FUNCION ANTERIOR POR FOURIER

tstep=0.006517;
r=0.01844;

a0 = -1.274e-08;
a1 = -3.874e-08;
b1 = -2.58e-07;
a2 = 1.178e-07;
b2 = -1.303e-07;
a3 = 7.396e-08;
b3 = -3.978e-08;
a4 = 5.99e-08;
b4 = 2.393e-08;
a5 = 1.795e-08;
b5 = 3.11e-08;
a6 = 3.514e-09;
b6 = 2.454e-08;
a7 = -1.665e-08;
b7 = 1.072e-08;
a8 = -3.288e-09;
b8 = 6.413e-09;
w = 9.642;

for i=1:length(tinterp)
    Q_fourier(i)=a0 + a1*cos(tinterp(i)*w)+b1*sin(tinterp(i)*w)+...
    a2*cos(2*tinterp(i)*w)+b2*sin(2*tinterp(i)*w)+a3*cos(3*tinterp(i)*w)+...
    b3*sin(3*tinterp(i)*w)+a4*cos(4*tinterp(i)*w)+b4*sin(4*tinterp(i)*w)+...
    a5*cos(5*tinterp(i)*w)+b5*sin(5*tinterp(i)*w)+a6*cos(6*tinterp(i)*w)+...
    b6*sin(6*tinterp(i)*w)+a7*cos(7*tinterp(i)*w)+b7*sin(7*tinterp(i)*w)+...
    a8*cos(8*tinterp(i)*w)+b8*sin(8*tinterp(i)*w);
end

figure(2)
plot(tinterp,Q_fourier,'k--',tinterp,Q_interp,'go-')
```

```
%AMORTIGUAMIENTO Y DESFASE
z=linspace(-0.6,0,101);
Ca=26.26*z.^4+ 25.31*z.^3 + 6.274*z.^2 + 1.989*z+ 1.00;

figure(3)
f1=subplot(2,1,1)
plot(f1,z,Ca,'b')
xlabel('Coordenada Longitudinal z (m)');
ylabel('Coef. de amortigumiento');
axis tight

f2=subplot(2,1,2)
phi=0.1353-1.7473*z;
plot(f2,z,phi,'b')
xlabel('Coordenada longitudinal z (m)')
ylabel('Desfase (rad)')

axis tight

%MAPA DE CAUDALES

for i = 1:length(z)
    for j =1:length(tinterp)

        caudal(j,i)=Ca(i)*(a0+a1*cos(tinterp(j)*w-phi(i))+...
        b1*sin(tinterp(j)*w-phi(i))+a2*cos(2*tinterp(j)*w-phi(i))+...
        b2*sin(2*tinterp(j)*w-phi(i))+a3*cos(3*tinterp(j)*w-phi(i))+...
        b3*sin(3*tinterp(j)*w-phi(i)) + a4*cos(4*tinterp(j)*w-phi(i))+ ...
        b4*sin(4*tinterp(j)*w-phi(i)) + a5*cos(5*tinterp(j)*w-phi(i))+ ...
        b5*sin(5*tinterp(j)*w-phi(i)) + a6*cos(6*tinterp(j)*w-phi(i))+...
        b6*sin(6*tinterp(j)*w-phi(i)) + a7*cos(7*tinterp(j)*w-phi(i))+...
        b7*sin(7*tinterp(j)*w-phi(i)) +a8*cos(8*tinterp(j)*w-phi(i))+...
        b8*sin(8*tinterp(j)*w-phi(i)));

    end
end

figure(4)
surf(tinterp,z,caudal)
ylabel('Tiempo (s)');
xlabel('Coordenada Longitudinal z (m)');
zlabel('Flujo Volumétrico (m^3/s)');
title('Mapa de Flujo volumétrico')
axis tight

%MAPA DE DESPLAZAMIENTOS

for i=1:length(z)
    for j=1:length(tinterp)

        disp(i,j)=(1/(2*pi*r)*tstep)*(-(1.+ 1.989*z(i)+6.274*z(i)^2+...
        25.31*z(i)^3+26.26*z(i)^4)*(1.7473*b8*cos(0.1353-8*tinterp(j)*w-
        ...
        1.7473*z(i))+1.7473*b7*cos(0.1353-7*tinterp(j)*w-1.7473*z(i))+...
        1.7473*b6*cos(0.1353-6*tinterp(j)*w-1.7473*z(i))+...
        1.7473*b5*cos(0.1353-5*tinterp(j)*w-1.7473*z(i))+...
```

```

1.7473*b4*cos(0.1353-4*tinterp(j)*w-1.7473*z(i))+...
1.7473*b3*cos(0.1353-3*tinterp(j)*w-1.7473*z(i))+...
1.7473*b2*cos(0.1353-2*tinterp(j)*w-1.7473*z(i))+...
1.7473*b1*cos(0.1353-tinterp(j)*w-1.7473*z(i))+...
1.7473*a8*sin(0.1353-8*tinterp(j)*w-1.7473*z(i))+...
1.7473*a7*sin(0.1353-7*tinterp(j)*w-1.7473*z(i))+...
1.7473*a6*sin(0.1353-6*tinterp(j)*w-1.7473*z(i))+...
1.7473*a5*sin(0.1353-5*tinterp(j)*w-1.7473*z(i))+...
1.7473*a4*sin(0.1353-4*tinterp(j)*w-1.7473*z(i))+...
1.7473*a3*sin(0.1353-3*tinterp(j)*w-1.7473*z(i))+...
1.7473*a2*sin(0.1353-2*tinterp(j)*w-1.7473*z(i))+...
1.7473*a1*sin(0.1353-tinterp(j)*w-1.7473*z(i))-...
(1.989+12.548*z(i)+75.93*z(i)^2 + 105.04*z(i)^3)*...
(a0+a8*cos(0.1353-8*tinterp(j)*w -1.7473*z(i))+...
a7*cos(0.1353-7*tinterp(j)*w-1.7473 *z(i))+...
a6*cos(0.1353-6*tinterp(j)*w-1.7473*z(i))+...
a5*cos(0.1353-5*tinterp(j)*w-1.7473*z(i))+...
a4*cos(0.1353-4*tinterp(j)*w-1.7473*z(i))+...
a3*cos(0.1353-3*tinterp(j)*w-1.7473*z(i))+...
a2*cos(0.1353-2*tinterp(j)*w-1.7473*z(i))+...
a1*cos(0.1353-tinterp(j)*w-1.7473*z(i))-...
b8*sin(0.1353-8*tinterp(j)*w-1.7473*z(i))-...
b7*sin(0.1353-7*tinterp(j)*w-1.7473*z(i))-...
b6*sin(0.1353-6*tinterp(j)*w-1.7473*z(i))-...
b5*sin(0.1353-5*tinterp(j)*w-1.7473*z(i))-...
b4*sin(0.1353-4*tinterp(j)*w-1.7473*z(i))-...
b3*sin(0.1353-3*tinterp(j)*w-1.7473*z(i))-...
b2*sin(0.1353-2*tinterp(j)*w-1.7473*z(i))-...
b1*sin(0.1353-tinterp(j)*w-1.7473*z(i))));

end
end

```

```

% CODIGO POST-PROCESADO DE BARRIDO DE BETAS

tend=0.6517;
zend=-0.6;
%Representamos el caudal utilizado para obtener 'dr'

tf=0.6517;
z=linspace(-0.604,0,101);
t=linspace(0,tf,101);
Caa=26.26*z.^4 + 25.31*z.^3 + 6.274*z.^2 + 1.989*z+1.00;
phi=0.1353-1.7473*z;

for i=1:length(z)
    for j=1:length(t)
        caudal(j,i)=(6*10^-7)*Caa(i)*sin((2*pi*t(j)/tf)-phi(i));
    end
end

figure(1)
plot(t,caudal(:,end),'k')
xlabel('Tiempo (s)')

```

```

ylabel('Flujo Volumétrico (m^3/s)')
title('Flujo Volumétrico en Foramen Magnun')
axis tight
grid on

figure(2)
surf(z,t,caudal)
ylabel('Tiempo(s)')
xlabel('Coordenada z (m)')
zlabel('Flujo Volumétrico (m^3/s)')
title('Mapa de Flujo Volumétrico')
axis tight
grid on

%%%%%% BETA = 0

p1z0_beta0=dlmread('p1z0_beta0.txt');
t_p1z0_beta0=p1z0_beta0(:,1);
t_p1z0_beta0=t_p1z0_beta0-t_p1z0_beta0(1);
vz_p1z0_beta0=p1z0_beta0(:,2);

p2z0_beta0=dlmread('p2z0_beta0.txt');
t_p2z0_beta0=p2z0_beta0(:,1);
t_p2z0_beta0=t_p2z0_beta0-t_p2z0_beta0(1);
vz_p2z0_beta0=p2z0_beta0(:,2);

p1z200_beta0=dlmread('p1z200_beta0.txt');
t_p1z200_beta0=p1z200_beta0(:,1);
t_p1z200_beta0=t_p1z200_beta0-t_p1z200_beta0(1);
vz_p1z200_beta0=p1z200_beta0(:,2);

p2z200_beta0=dlmread('p2z200_beta0.txt');
t_p2z200_beta0=p2z200_beta0(:,1);
t_p2z200_beta0=t_p2z200_beta0-t_p2z200_beta0(1);
vz_p2z200_beta0=p2z200_beta0(:,2);

p1z400_beta0=dlmread('p1z400_beta0.txt');
t_p1z400_beta0=p1z400_beta0(:,1);
t_p1z400_beta0=t_p1z400_beta0-t_p1z400_beta0(1);
vz_p1z400_beta0=p1z400_beta0(:,2);

p2z400_beta0=dlmread('p2z400_beta0.txt');
t_p2z400_beta0=p2z400_beta0(:,1);
t_p2z400_beta0=t_p2z400_beta0-t_p2z400_beta0(1);
vz_p2z400_beta0=p2z400_beta0(:,2);

figure(3)
f1=subplot(2,1,1)
plot(f1,t_p1z0_beta0,vz_p1z0_beta0,'r',t_p1z200_beta0,vz_p1z200_beta0,...
'g',t_p1z400_beta0,vz_p1z400_beta0,'k')
ylabel('Velocidad Z v_z (m/s)')
title('Velocidades Canal Ancho')
legend('z=0','z=200','z=400')
grid on

f2=subplot(2,1,2)

```

```

plot(f2,t_p2z0_beta0,vz_p2z0_beta0,'r',t_p2z200_beta0,vz_p2z200_beta0,'g',.
...
t_p2z400_beta0,vz_p2z400_beta0,'k')
xlabel('Tiempo (s)')
ylabel('Velocidad Z v_z (m/s)')
title('Velocidades Canal Estrecho')
legend('z=0','z=200','z=400')
axis tight
grid on

suptitle('\beta = 0');

flujovol_foramen_beta0=dlmread('flujomasico_foramen_beta0.txt');
t_fv_foramen_beta0=flujovol_foramen_beta0(:,1);
t_fv_foramen_beta0=t_fv_foramen_beta0-t_fv_foramen_beta0(1);
fv_foramen_beta0=-flujovol_foramen_beta0(:,2)/993.3;

flujovol_pz200_beta0=dlmread('flujomasico_pz200_beta0.txt');
t_fv_pz200_beta0=flujovol_pz200_beta0(:,1);
t_fv_pz200_beta0=t_fv_pz200_beta0-t_fv_pz200_beta0(1);
fv_pz200_beta0=-flujovol_pz200_beta0(:,2)/993.3;

flujovol_pz400_beta0=dlmread('flujomasico_pz400_beta0.txt');
t_fv_pz400_beta0=flujovol_pz400_beta0(:,1);
t_fv_pz400_beta0=t_fv_pz400_beta0-t_fv_pz400_beta0(1);
fv_pz400_beta0=-flujovol_pz400_beta0(:,2)/993.3;

figure(4)
plot(t_fv_foramen_beta0,fv_foramen_beta0,'ro',t,caudal(:,end),'r',...
t_fv_pz200_beta0,fv_pz200_beta0,'go',t,caudal(:,68),'g',t_fv_pz400_beta0,...
.
fv_pz400_beta0,'ko',t,caudal(:,34),'k')
xlabel('Tiempo (s)')
ylabel('Flujo volumétrico (m^3/s)')
title('Flujo volumétrico a lo largo del canal espinal')
legend('z=0m simulación','z=0m teórico','z=0.2m simulación',...
'z=0.2m teórico','z=0.4m simulación','z=0.4 teórico')
grid on
axis tight
suptitle('\beta = 0')

%VELOCIDADES ESTACIONARIAS.

%% BETA = 0

p1b0=dlmread('velocidad_estacionaria_p1_beta0.txt')
vp1b0=p1b0(:,2);
tp1b0=p1b0(:,1);
tp1b0=tp1b0-tp1b0(1);

p2b0=dlmread('velocidad_estacionaria_p2_beta0.txt')
vp2b0=p2b0(:,2);
tp2b0=p2b0(:,1);
tp2b0=tp2b0-tp2b0(1);

```



```
p3b0=dlmread('velocidad_estacionaria_p3_beta0.txt')
vp3b0=p3b0(:,2);
tp3b0=p3b0(:,1);
tp3b0=tp3b0-tp3b0(1);

p4b0=dlmread('velocidad_estacionaria_p4_beta0.txt')
vp4b0=p4b0(:,2);
tp4b0=p4b0(:,1);
tp4b0=tp4b0-tp4b0(1);

p5b0=dlmread('velocidad_estacionaria_p5_beta0.txt')
vp5b0=p5b0(:,2);
tp5b0=p5b0(:,1);
tp5b0=tp5b0-tp5b0(1);

p6b0=dlmread('velocidad_estacionaria_p6_beta0.txt')
vp6b0=p6b0(:,2);
tp6b0=p6b0(:,1);
tp6b0=tp6b0-tp6b0(1);

figure(5)

f1=subplot(2,2,1)
plot(f1,tp1b0,vp1b0,'r',tp2b0,vp2b0,'m',tp3b0,vp3b0,'b',tp4b0,vp4b0,'y',tp5b0,vp5b0,'c',tp6b0,vp6b0,'g','LineWidth',1.5)
xlabel('Tiempo (s)');
ylabel('V_z (m/s)');
title('Velocidad oscilatoria');
axis tight
grid on

f2=subplot(2,2,2)
plot(f2,[0 tend],[2.74406e-007 -2.74406e-007],'r',[0 tend],[2.65691e-007 2.65691e-007],'m',[0 tend],[7.7121e-006 7.7121e-006],'b',[0 tend],[7.22703e-006 7.22703e-006],'y',[0 tend],[-6.81733e-007 -6.81733e-007],'c',[0 tend],[-1.5408e-006 -1.5408e-006],'g','LineWidth',1.5)
xlabel('Tiempo (s)');
ylabel('V_z (m/s)');
title('Velocidad estacionaria');
axis tight
grid on

f3=subplot(2,2,[3,4])
plot(f3,tp1b0,vp1b0,'r',tp2b0,vp2b0,'m',tp3b0,vp3b0,'b',tp4b0,vp4b0,'y',tp5b0,vp5b0,'c',tp6b0,vp6b0,'g','LineWidth',2)
xlabel('Tiempo (s)');
ylabel('V_z (m/s)');
title('Detalle de la velocidad oscilatoria');
legend('P1','P2','P3','P4','P5','P6');
grid on;
axis([0.3455 0.3475 -0.1e-03 0.1e-03])

suptitle('\beta = 0');

rbeta0=linspace(0.01344,0.01844,21);
```

```
vm_beta0_foramen=[0 -1.39031e-07 -2.74406e-07 -4.09774e-07 -5.45147e-07
2.6569e-07 2.74784e-06 5.22996e-06 7.7121e-06 1.01942e-05 9.8633e-06
7.22703e-06 4.59077e-06 1.95451e-06 -6.81733e-07 -2.30465e-06 -1.92273e-06
-1.5408e-06 -1.15888e-06 -7.7695e-07 0 ];

vm_beta0_z200=[0 -6.79416e-07 -1.34093e-06 -2.02244e-06 -2.66395e-06 -
3.30167e-06 -1.50485e-06 2.91975e-07 2.08879e-06 3.88561e-06 5.53418e-06
3.53004e-06 1.5259e-06 -4.78234e-07 -2.48235e-06 -4.5275e-06 -3.6574e-06 -
2.7873e-06 -1.91719e-06 -1.0471e-06 0];

vm_beta0_z400=[0 -4.32083e-07 -8.52776e-07 -1.27347e-06 -1.69417e-06 -
2.09169e-06 -1.55333e-06 -1.01497e-06 -4.76604e-07 6.17607e-08 5.53943e-07
-4.35298e-08 -6.40998e-07 -1.23847e-06 -1.83594e-06 -2.4103e-06 -1.93027e-
06 -1.45024e-06 -9.70213e-07 -4.90182e-07 0];

figure(6)
f1=subplot(3,1,1);
plot(f1,(rbeta0-0.01344)/0.005,vm_beta0_foramen,'bo--');
ylabel('V_{z,media} (m/s)')
title('Foramen Magnum z=0m');
grid on
axis tight

f2=subplot(3,1,2);
plot(f2,(rbeta0-0.01344)/0.005,vm_beta0_z200,'bo--');
ylabel('V_{z,media} (m/s)')
title('z=-0.2m');
grid on
axis tight

f3=subplot(3,1,3);
plot(f3,(rbeta0-0.01344)/0.005,vm_beta0_z400,'bo--');
xlabel('e=(x-d_{pia})/h')
ylabel('V_{z,media} (m/s)')
title('z=-0.4m');
grid on
axis tight
suptitle('\beta = 0');

%% Del mismo modo se procederá con todos los casos, cambiando tan solo el
valor beta en el nombre de cada una de las variables.
```

```
format long
clear all
close all

tend=0.6517;
zend=-0.6;

load('Q(t)- FM.mat');
```

```
t=t/1000;
puntos=100;
T=t(length(t));

tinterp=(0:(T/puntos):T);
%%%%
tinterp=linspace(5.80012989,6.43879604,99)
Q_interp= interp1(t,Q,tinterp)/10^6;
figure(1)
plot(tinterp,Q_interp,'k',t,Q/10^6,'go')
xlabel('Tiempo(s)');
ylabel('Flujo volumétrico (ml/s)')
title('Flujo Volumétrico en Foramen Magnum')
legend('Función interpolada','Datos paper')
axis tight

%% INTERPOLAMOS LA FUNCION ANTERIOR POR FOURIER

tstep=0.006517;
r=0.01844;

a0 = -1.274e-08;
a1 = -3.874e-08;
b1 = -2.58e-07;
a2 = 1.178e-07;
b2 = -1.303e-07;
a3 = 7.396e-08;
b3 = -3.978e-08;
a4 = 5.99e-08;
b4 = 2.393e-08;
a5 = 1.795e-08;
b5 = 3.11e-08;
a6 = 3.514e-09;
b6 = 2.454e-08;
a7 = -1.665e-08;
b7 = 1.072e-08;
a8 = -3.288e-09;
b8 = 6.413e-09;
w = 9.642;

for i=1:length(tinterp)
    Q_fourier(i)=a0 + a1*cos(tinterp(i)*w)+b1*sin(tinterp(i)*w)+...
    a2*cos(2*tinterp(i)*w)+b2*sin(2*tinterp(i)*w)+a3*cos(3*tinterp(i)*w)+...
    b3*sin(3*tinterp(i)*w)+a4*cos(4*tinterp(i)*w)+b4*sin(4*tinterp(i)*w)+...
    a5*cos(5*tinterp(i)*w)+b5*sin(5*tinterp(i)*w)+a6*cos(6*tinterp(i)*w)+...
    b6*sin(6*tinterp(i)*w)+a7*cos(7*tinterp(i)*w)+b7*sin(7*tinterp(i)*w)+...
    a8*cos(8*tinterp(i)*w)+b8*sin(8*tinterp(i)*w);
end

figure(2)
plot(tinterp,Q_fourier,'k--',tinterp,Q_interp,'go-')
```

```

%% AMORTIGUAMIENTO Y DESFASE
z=linspace(-0.6,0,101);
Ca=26.26*z.^4+ 25.31*z.^3 + 6.274*z.^2 + 1.989*z+ 1.00;

figure(3)
f1=subplot(2,1,1)
plot(f1,z,Ca,'b')
xlabel('Coordenada Longitudinal z (m)');
ylabel('Coef. de amortiguamiento');
axis tight

f2=subplot(2,1,2)
phi=0.1353-1.7473*z;
plot(f2,z,phi,'b')
xlabel('Coordenada longitudinal z (m)')
ylabel('Desfase (rad)')

axis tight

%% MAPA DE CAUDALES

for i = 1:length(z)
    for j =1:length(tinterp)

        caudal(j,i)=Ca(i)*(a0+a1*cos(tinterp(j)*w-phi(i))+...
        b1*sin(tinterp(j)*w-phi(i))+a2*cos(2*tinterp(j)*w-phi(i))+...
        b2*sin(2*tinterp(j)*w-phi(i))+a3*cos(3*tinterp(j)*w-phi(i))+...
        b3*sin(3*tinterp(j)*w-phi(i)) + a4*cos(4*tinterp(j)*w-phi(i))+ ...
        b4*sin(4*tinterp(j)*w-phi(i)) + a5*cos(5*tinterp(j)*w-phi(i))+ ...
        b5*sin(5*tinterp(j)*w-phi(i)) + a6*cos(6*tinterp(j)*w-phi(i))+...
        b6*sin(6*tinterp(j)*w-phi(i)) + a7*cos(7*tinterp(j)*w-phi(i))+...
        b7*sin(7*tinterp(j)*w-phi(i)) +a8*cos(8*tinterp(j)*w-phi(i))+...
        b8*sin(8*tinterp(j)*w-phi(i)));

    end
end

flujovol_foramen=dlmread('f_masico_foramen.txt');
t_fv_foramen=flujovol_foramen(:,1);
%t_fv_foramen=t_fv_foramen-t_fv_foramen(1);
fv_foramen=-flujovol_foramen(:,2)/993.3;

flujovol_pz200=dlmread('f_masico_z200.txt');
t_fv_pz200=flujovol_pz200(:,1);
%t_fv_pz200=t_fv_pz200-t_fv_pz200(1);
fv_pz200=-flujovol_pz200(:,2)/993.3;

flujovol_pz400=dlmread('f_masico_z400.txt');
t_fv_pz400=flujovol_pz400(:,1);
%t_fv_pz400=t_fv_pz400-t_fv_pz400(1);
fv_pz400=-flujovol_pz400(:,2)/993.3;
fv_pz400=-flujovol_pz400(:,2)/993.3;

figure(5)
plot(t_fv_foramen,fv_foramen,'ro',tinterp,caudal(:,end),'r',...
t_fv_pz200,fv_pz200,'go',tinterp,caudal(:,68),'g',t_fv_pz400,...
fv_pz400,'ko',tinterp,caudal(:,34),'k')

```

```

xlabel('Tiempo (s)')
ylabel('Flujo volumétrico (m^3/s)')
title('Flujo volumétrico a lo largo del canal espinal')
legend('z=0m simulación','z=0m teórico','z=0.2m simulación',...
'z=0.2m teórico','z=0.4m simulación','z=0.4 teórico')
grid on
axis tight

%velocidades oscilatorias.

p1z0=dlmread('p1z0.txt');
t_p1z0=p1z0(:,1);
%t_p1z0=t_p1z0-t_p1z0(1);
vz_p1z0=p1z0(:,2);

p2z0=dlmread('p2z0.txt');
t_p2z0=p2z0(:,1);
%t_p2z0=t_p2z0-t_p2z0(1);
vz_p2z0=p2z0(:,2);

p1z200=dlmread('p1z200.txt');
t_p1z200=p1z0(:,1);
%t_p1z200=t_p1z200-t_p1z200(1);
vz_p1z200=p1z200(:,2);

p2z200=dlmread('p2z200.txt');
t_p2z200=p2z200(:,1);
%t_p2z200=t_p2z200-t_p2z200(1);
vz_p2z200=p2z200(:,2);

p1z400=dlmread('p1z400.txt');
t_p1z400=p1z400(:,1);
%t_p1z400=t_p1z400-t_p1z400(1);
vz_p1z400=p1z400(:,2);

p2z400=dlmread('p2z400.txt');
t_p2z400=p2z400(:,1);
%t_p2z400=t_p2z400-t_p2z400(1);
vz_p2z400=p2z400(:,2);

figure(11)
f1=subplot(2,1,1)
plot(f1,t_p1z0,vz_p1z0,'r',t_p1z200,vz_p1z200,'g',t_p1z400,vz_p1z400,'k')
ylabel('Velocidad Z v_z (m/s)')
title('Velocidad Canal Ancho')
legend('z=0','z=200','z=400')
axis tight
grid on
f2=subplot(2,1,2)
plot(f2,t_p2z0,vz_p2z0,'r',t_p2z200,vz_p2z200,'g',t_p2z400,vz_p2z400,'k')
xlabel('Tiempo (s)')
ylabel('Velocidad Z v_z (m/s)')
title('Velocidad Canal Estrecho')
legend('z=0','z=200','z=400')
axis tight

```

```

grid on
suptitle('Caudal real Geometría real')

%% BETA 0,5 GEOMETRÍA SIMPLIFICADA.
ribeta05=linspace(-0.01844,-0.01094,21);
vmi_beta05_z0=[0 -6.48178e-06 -1.28298e-05 -1.91778e-05 -2.55258e-05 -
3.18737e-05 -3.43402e-05 -3.68067e-05 -3.92732e-05 -4.17397e-05 -4.42062e-
05 -4.2243e-05 -4.02424e-05 -3.82605e-05 -3.62786e-05 -3.42966e-05 -
2.7437e-05 -2.0578e-05 -1.37187e-05 -6.85933e-06 0];
vmi_beta05_z200=[0 -4.46033e-06 -8.85037e-06 -1.32404e-05 -1.76304e-05 -
2.2020e-05 -2.39942e-05 -2.59679e-05 -2.79416e-05 -2.99153e-05 -3.1889e-05
-3.02195e-05 -2.85501e-05 -2.68806e-05 -2.52112e-05 -2.35417e-05 -1.88334e-
05 -1.4215e-05 -9.4167e-06 -4.70835e-06 0];
vmi_beta05_z400=[0 -1.5937e-06 -3.16562e-06 -4.73754e-06 -6.30946e-06 -
7.88138e-06 -8.61239e-06 -9.3434e-06 -1.00744e-05 -1.08054e-05 -1.15364e-05
-1.09177e-05 -1.0299e-05 -9.68026e-06 -9.06155e-06 -8.44283e-06 -6.75426e-
06 -5.0657e-06 -3.37713e-06 -1.168857e-06 0];
figure(14)
f1=subplot(3,1,1);
plot(f1,((-0.01844-ribeta05)/7.25e-03),vmi_beta05_z0,'bo-')
ylabel('V_{z,media} (m/s)')
axis tight
grid on
f2=subplot(3,1,2);
plot(f2,((-0.01844-ribeta05)/7.25e-03),vmi_beta05_z200,'bo-')
ylabel('V_{z,media} (m/s)')
axis tight
grid on
f3=subplot(3,1,3);
plot(f3,((-0.01844-ribeta05)/7.25e-03),vmi_beta05_z400,'bo-')
xlabel('(x-D_{pía})/h_1')
ylabel('V_{z,media} (m/s)')
axis tight
grid on
suptitle('\beta = 0.5 Canal Ancho')

rdbeta05=linspace(0.01594,0.01844,21);
vmd_beta05_z0=[0 -6.19549e-07 -6.40733e-07 -6.61913e-07 -6.83101e-07 -
7.04285e-07 -7.25466e-07 -7.46646e-07 -7.67835e-07 -7.89012e-07 -7.98544e-
07 -7.34827e-07 -6.71114e-07 -6.07382e-07 -5.43684e-07 -4.79968e-07 -
4.16241e-07 -3.52547e-07 -2.88821e-07 -2.25073e-07 0];
vmd_beta05_z200=[0 1.14174e-06 1.17032e-06 1.19891e-06 1.2275e-06 1.25608e-
06 1.28467e-06 1.31326e-06 1.34184e-06 1.37043e-06 1.37495e-06 1.22827e-06
1.0816e-06 9.34926e-07 7.88258e-07 6.41581e-07 4.9489e-07 3.48245e-07
2.01551e-07 5.48864e-08 0];
vmd_beta05_z400=[0 7.43509e-07 7.55101e-07 7.66689e-07 7.7828e-07 7.89868e-
07 8.0146e-07 8.1305e-07 8.24638e-07 8.36229e-07 8.34636e-07 7.50209e-07
6.65782e-07 5.81354e-07 4.96929e-07 4.12504e-07 3.28076e-07 2.43653e-07
1.59225e-07 7.48012e-08 0];
figure(13)
f1=subplot(3,1,1);
plot(f1,(rdbeta05-0.01594)/2.5e-03),vmd_beta05_z0,'bo-');
title('z=0m')
axis tight
grid on

```

```
f2=subplot(3,1,2);
plot(f2, ((rdbeta05-0.01594)/2.5e-03), vmd_beta05_z200, 'bo-')
title('z=-0.2m')
axis tight
grid on
f3=subplot(3,1,3);
plot(f3, ((rdbeta05-0.01594)/2.5e-03), vmd_beta05_z400, 'bo-')
title('z=-0.4m')
axis tight
grid on
suptitle('\beta = 0.5 Canal Estrecho')

%velocidades oscilatorias.

plz0_beta05=dlmread('plz0_beta05.txt');
t_plz0_beta05=plz0_beta05(:,1);
%t_plz0_beta05=t_plz0_beta05-t_plz0_beta05(1);
vz_plz0_beta05=plz0_beta05(:,2);

p2z0_beta05=dlmread('p2z0_beta05.txt');
t_p2z0_beta05=p2z0_beta05(:,1);
%t_p2z0_beta05=t_p2z0_beta05-t_p2z0_beta05(1);
vz_p2z0_beta05=p2z0_beta05(:,2);

plz200_beta05=dlmread('plz200_beta05.txt');
t_plz200_beta05=plz200_beta05(:,1);
%t_plz200_beta05=t_plz200_beta05-t_plz200_beta05(1);
vz_plz200_beta05=plz200_beta05(:,2);

p2z200_beta05=dlmread('p2z200_beta05.txt');
t_p2z200_beta05=p2z200_beta05(:,1);
%t_p2z200_beta05=t_p2z200_beta05-t_p2z200_beta05(1);
vz_p2z200_beta05=p2z200_beta05(:,2);

plz400_beta05=dlmread('plz400_beta05.txt');
t_plz400_beta05=plz400_beta05(:,1);
%t_plz400_beta05=t_plz400_beta05-t_plz400_beta05(1);
vz_plz400_beta05=plz400_beta05(:,2);

p2z400_beta05=dlmread('p2z400_beta05.txt');
t_p2z400_beta05=p2z400_beta05(:,1);
%t_p2z400_beta05=t_p2z400_beta05-t_p2z400_beta05(1);
vz_p2z400_beta05=p2z400_beta05(:,2);

figure(15)
f1=subplot(2,1,1)
plot(f1,t_plz0_beta05,vz_plz0_beta05,'r',t_plz200_beta05,vz_plz200_beta05,'g',t_plz400_beta05,vz_plz400_beta05,'k')
ylabel('Velocidad Z v_z (m/s)')
title('Velocidad Canal Ancho')
legend('z=0', 'z=200', 'z=400')
axis tight
grid on
f2=subplot(2,1,2)
plot(f2,t_p2z0_beta05,vz_p2z0_beta05,'r',t_p2z200_beta05,vz_p2z200_beta05,'g',t_p2z400_beta05,vz_p2z400_beta05,'k')
xlabel('Tiempo (s)')
```

```

ylabel('Velocidad Z v_z (m/s)')
title('Velocidad Canal Estrecho')
legend('z=0','z=200','z=400')
axis tight
grid on

flujovol_foramen_beta05=dlmread('f_masico_foramen_beta05.txt');
t_fv_foramen_beta05=flujovol_foramen_beta05(:,1);
t_fv_foramen_beta05=t_fv_foramen_beta05-t_fv_foramen_beta05(1);
fv_foramen_beta05=-flujovol_foramen_beta05(:,2)/993.3;

flujovol_pz200_beta05=dlmread('f_masico_z200_beta05.txt');
t_fv_pz200_beta05=flujovol_pz200_beta05(:,1);
t_fv_pz200_beta05=t_fv_pz200_beta05-t_fv_pz200_beta05(1);
fv_pz200_beta05=-flujovol_pz200_beta05(:,2)/993.3;

flujovol_pz400_beta05=dlmread('f_masico_z400_beta05.txt');
t_fv_pz400_beta05=flujovol_pz400_beta05(:,1);
t_fv_pz400_beta05=t_fv_pz400_beta05-t_fv_pz400_beta05(1);
fv_pz400_beta05=-flujovol_pz400_beta05(:,2)/993.3;

tinterp=(0:(T/puntos):T);
for i = 1:length(z)
    for j =1:length(tinterp)

        caudal(j,i)=Ca(i)*(a0+a1*cos(tinterp(j)*w-phi(i))+...
        b1*sin(tinterp(j)*w-phi(i))+a2*cos(2*tinterp(j)*w-phi(i))+...
        b2*sin(2*tinterp(j)*w-phi(i))+a3*cos(3*tinterp(j)*w-phi(i))+...
        b3*sin(3*tinterp(j)*w-phi(i)) + a4*cos(4*tinterp(j)*w-phi(i))+ ...
        b4*sin(4*tinterp(j)*w-phi(i)) + a5*cos(5*tinterp(j)*w-phi(i))+ ...
        b5*sin(5*tinterp(j)*w-phi(i)) + a6*cos(6*tinterp(j)*w-phi(i))+...
        b6*sin(6*tinterp(j)*w-phi(i)) + a7*cos(7*tinterp(j)*w-phi(i))+...
        b7*sin(7*tinterp(j)*w-phi(i)) +a8*cos(8*tinterp(j)*w-phi(i))+...
        b8*sin(8*tinterp(j)*w-phi(i)));
    end
end

figure(18)
plot(t_fv_foramen_beta05,fv_foramen_beta05,'ro',tinterp,caudal(:,end),'r',...
...
t_fv_pz200_beta05,fv_pz200_beta05,'go',tinterp,caudal(:,68),'g',...
t_fv_pz400_beta05,fv_pz400_beta05,'ko',tinterp,caudal(:,34),'k')
xlabel('Tiempo (s)')
ylabel('Flujo volumétrico (m^3/s)')
title('Flujo volumétrico a lo largo del canal espinal')
legend('z=0m simulación','z=0m teórico','z=0.2m simulación',...
'z=0.2m teórico','z=0.4m simulación','z=0.4m teórico')
grid on
axis tight

```