



*Facultad  
de  
Ciencias*

**RESOLUCIÓN NUMÉRICA DE LAS ECUACIONES DE  
NAVIER-STOKES**  
(Numerical solution of the Navier-Stokes equations)

Trabajo de Fin de Grado  
para acceder al

**GRADO EN FÍSICA**

Autor: Adriana Cavada López-Tapia

Director: Pablo García Fernández

Septiembre - 2015

# Índice general

<b>Índice general</b>	<b>1</b>
<b>1. Introducción</b>	<b>6</b>
1.1. Propósito . . . . .	6
1.2. Motivación . . . . .	6
1.3. Alcance . . . . .	8
<b>2. Núcleo teórico</b>	<b>9</b>
2.1. Clasificación de fluidos . . . . .	10
2.2. Ecuaciones de conservación . . . . .	10
2.2.1. Conservación de la materia . . . . .	11
2.2.2. Conservación del momento . . . . .	11
2.3. Ecuaciones simplificadas . . . . .	16
2.4. Condiciones de contorno . . . . .	17
<b>3. Implementación</b>	<b>18</b>
3.1. Método de pasos fraccionados (FSM) . . . . .	19
3.1.1. Pasos del FSM . . . . .	22
3.2. Discretización espacial . . . . .	23
3.2.1. Método de Volúmenes Finitos . . . . .	23
3.2.2. Mallado . . . . .	23
3.2.3. Discretización espacial de las ecuaciones . . . . .	26
3.3. Algoritmo de Resolución . . . . .	29
<b>4. Verificación del código</b>	<b>33</b>
4.1. Presentación del caso . . . . .	33
4.2. Resultados . . . . .	35
4.2.1. $Re = 100$ . . . . .	36

4.2.2. $Re = 1000$ . . . . .	42
4.2.3. $Re = 2000$ . . . . .	43
4.3. Discusión . . . . .	44
4.4. Conclusiones . . . . .	47
<b>Apéndices</b>	<b>II</b>
<b>A. Código MatLab</b>	<b>II</b>
<b>B. Formulación matemática del teorema de descomposición de Helmholtz-Hodge aplicado a las ecuaciones de Navier-Stokes</b>	<b>VII</b>
<b>Bibliografía</b>	<b>IX</b>

# Resumen

El objetivo fundamental de este proyecto consiste en resolver numéricamente las ecuaciones de Navier-Stokes en su forma incompresible mediante la creación de un código programado con MatLab que permita resolver y estudiar la fenomenología de la mecánica de fluidos. La idea principal es profundizar en la física que hay detrás del proceso de la implementación computacional de un problema, por lo que el código resolverá casos con geometrías sencillas, y la eficiencia del mismo no será una prioridad. Se ha propuesto una discretización espacial mediante el método de volúmenes finitos y se ha utilizado el método de pasos fraccionados para resolver el acoplamiento de la velocidad y la presión.

Con el fin de validar el programa se ha utilizado un caso test denominado “driven cavity” en dos dimensiones, caso muy común en la dinámica de fluidos computacional para validar nuevos códigos que resuelven el movimiento de fluidos. Se han estudiado tres números de Reynolds en régimen laminar, 100, 1000 y 2000. Los resultados obtenidos con el código nuevo son similares a los obtenidos con el software comercial ANSYS Fluent.

***Palabras clave:** ecuaciones de Navier-Stokes, dinámica de fluidos computacional, método de pasos fraccionados, volúmenes finitos, número de Reynolds, driven cavity*

# Abstract

The main objective of this project is to numerically solve the incompressible Navier-Stokes equations by creating a code programmed with MatLab that allows solving and studying the phenomenology of fluids dynamics. The principal idea is to delve into the physics behind the process of implementing computationally a problem, thus the code will solve cases with simple geometries, and its efficiency will not be a priority. We have proposed a spatial discretization by the finite volume method and used the fractional-step-method to solve the coupling between speed and pressure.

In order to validate the program we have used a test case called “ driven-cavity“ in two dimensions, very common case in computational fluid dynamics to validate new codes that solve the movement of fluids. We have studied three Reynolds numbers in laminar regime, 100, 1000 and 2000. The results obtained with the new code are similar to those obtained with the commercial software ANSYS Fluent.

***Key words:*** *Navier-Stokes equations, computational fluid dynamics, fractional-step-method, finite volumen method, Reynolds number, driven cavity*

# Nomenclatura

$x$	Coordenada eje de abscisas
$y$	Coordenada eje de ordenadas
$\rho$	Densidad
$P$	Operador proyección
$p$	Presión absoluta
$Co$	Número de Courant
$Re$	Número de Reynolds
$\bar{\bar{\tau}}$	Tensor de tensiones asociado a fuerzas de superficie
$\bar{\bar{\tau}}'$	Tensor de tensiones de esfuerzos viscosos
$\bar{\bar{I}}$	Tensor unitario
$t$	Tiempo
$\vec{f}_v$	Vector de fuerzas volumétricas
$\vec{v}$	Velocidad del fluido en un punto del espacio y en un instante de tiempo
$u$	Velocidad en la dirección x del movimiento
$v$	Velocidad en la dirección y del movimiento
$\nu$	Viscosidad cinemática
$\eta$	Viscosidad dinámica
$\zeta$	Viscosidad volumétrica

# Capítulo 1

## Introducción

### 1.1. Propósito

El propósito fundamental de este proyecto consiste en la creación de un código programado con MatLab que permita resolver las ecuaciones de Navier-Stokes en su forma incompresible y estudiar la fenomenología de la mecánica de fluidos y las bases de la dinámica de fluidos computacional<sup>1</sup>. La idea principal es profundizar en la física que hay detrás del proceso de la implementación computacional de un problema, por lo que el código resolverá casos con geometrías sencillas, y la eficiencia del mismo no será una prioridad. Se ha propuesto una discretización espacial mediante el método de volúmenes finitos y se ha utilizado el método de pasos fraccionados para resolver el acoplamiento de la velocidad y la presión.

### 1.2. Motivación

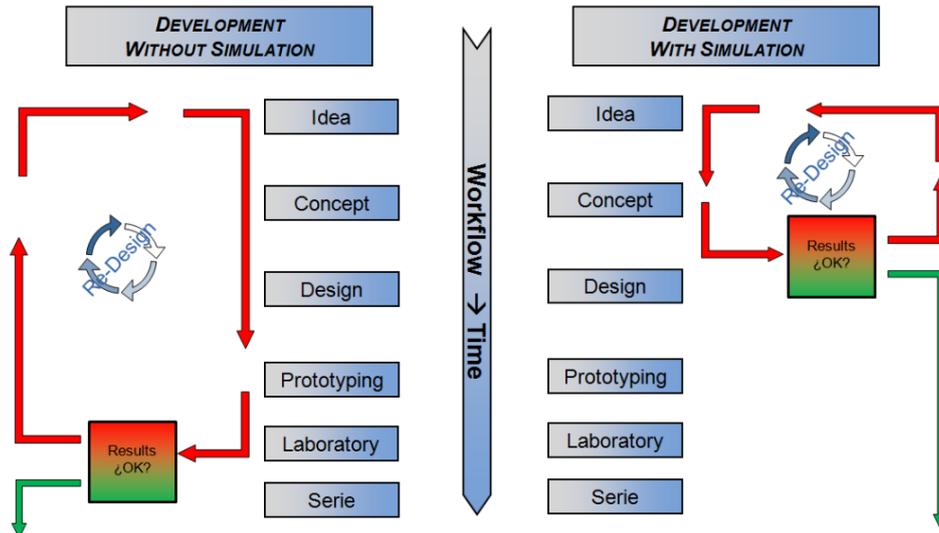
A continuación comentaremos las ventajas que presenta la simulación frente a otras herramientas de análisis y de este modo entender los motivos por los cuáles se ha elegido la simulación computacional como base de este proyecto.

Hoy en día la simulación es tanto una alternativa como un complemento a la experimentación en muchas ramas de la Ciencia ya que se trata de una herramienta barata y efectiva para predecir las propiedades y el comportamiento de un sistema físico. No sólo eso, sino que las técnicas computacionales son un complemento necesario a la teoría que permite comprobar las consecuencias que se obtienen de realizar hipótesis o modelos complejos en diferentes ramas de la física, y que de otra forma no se podrían chequear.

---

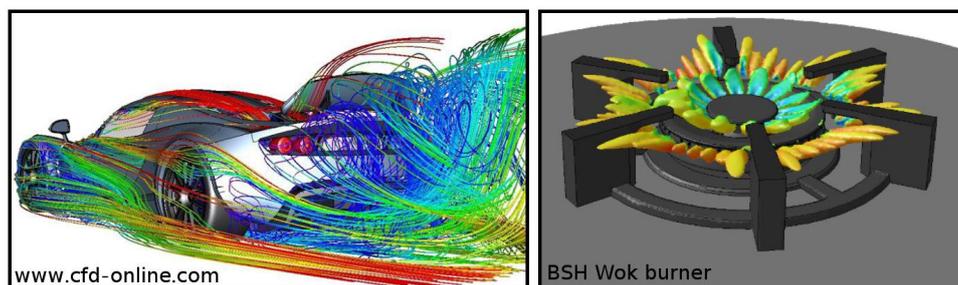
<sup>1</sup>Normalmente denominada CFD por su nombre inglés Computational Fluid Dynamics.

Más aún, la simulación nos permite mejorar de manera sistemática el diseño y las propiedades de materiales por medio de la optimización de algoritmos (ver figura 1.1). Un procedimiento así resultaría tremendamente costoso si se hiciera prototipando o por otros medios de experimentación. Así pues, podemos decir que la simulación es una herramienta que mejora la eficacia de los procesos tecnológicos, como se muestra en la siguiente figura.



**Figura 1.1** – Ventajas del uso de la simulación como herramienta en los procesos tecnológicos. Imagen obtenida de BSH Electrodomésticos España, departamento de Pre-desarrollo.

Dentro de la simulación, en concreto en el campo de la dinámica de fluidos computacional, debido a que la predicción de casos es tan complicada que no puede ser resuelta analíticamente, es donde la simulación aporta más beneficios y su uso está más que justificado.



**Figura 1.2** – Ejemplos de aplicación de CFD. Imágenes obtenidas de [www.cfd-online.com](http://www.cfd-online.com) y BSH Electrodomésticos España, departamento de Pre-desarrollo.

Por otro lado, como ya se ha mencionado, en este proyecto se pretende profundizar en la fenomenología física de la mecánica de fluidos, así como en los fundamentos computacionales de la simulación con herramientas CFD para lograr una base sólida en este campo de la física.

Es por ello que debemos entender la filosofía de este proyecto como un medio para comprender el funcionamiento de las muy complejas herramientas computacionales existentes hoy en día. Por esta razón se ha decidido crear un código propio desde cero en vez de utilizar herramientas ya en uso como pueden ser ANSYS Fluent, OpenFoam o RealFlow entre otros. Aunque dichos programas son muy útiles y espectaculares para la simulación de fluidos, su alta complejidad hace que no sean las herramientas más adecuadas para comprender en detalle como se solucionan las ecuaciones de la mecánica de fluidos.

### 1.3. Alcance

Antes de introducir los contenidos teóricos relacionados con el proyecto es necesario especificar ciertos puntos que afectan al alcance del mismo:

- Debido a que los fenómenos tratados en la dinámica de fluidos son macroscópicos, asumiremos la hipótesis de continuidad de la materia. Esto implica que cualquier elemento de volumen será suficientemente grande como para contener un número muy elevado de moléculas.
- Las propiedades físicas del fluido se considerarán siempre constantes dentro del rango de aplicación de las ecuaciones. Esto implica que se trabajará con fluidos incompresibles<sup>2</sup>.
- El régimen de trabajo será laminar, por lo que no se podrán resolver casos con flujos en estado turbulento.
- El programa trabajará con problemas bidimensionales.
- El objetivo del código es construir una herramienta de iniciación a la física de fluidos, por lo que la eficiencia del mismo no es una prioridad.

---

<sup>2</sup>Se pueden consultar las definiciones de los tipos de fluidos en la primera sección del siguiente capítulo.

## Capítulo 2

# Núcleo teórico

---

A continuación se explicarán los conceptos de la Mecánica de Fluidos necesarios para el seguimiento del proyecto. Dado que esta materia no aparece de forma detallada en el contenido de ninguna asignatura de la carrera, se ha procurado describir minuciosamente las bases de esta rama de la física. El contenido teórico de este apartado se ha descrito fundamentalmente siguiendo los siguientes libros:

- *Mecánica de Fluidos* de A. Crespo [9]
  - *Computational Methods for fluid dynamics* de J. H. Ferziger y M. Perić [10]
  - *Fundamentos de Fluidos y Procesos Fluidodinámicos* de N. Fueyo [11]
  - *Mecánica de Fluidos* de L. D. Landau [15]
  - *Computational fluid dynamics: a practical approach* de J. Tu, G. H. Yeoh y C. Liu [21]
-

## 2.1. Clasificación de fluidos

### ■ Estacionario - Transitorio

Un flujo estacionario es aquel en el que las variables que lo definen no cambian localmente con el tiempo; uno transitorio en el que sí lo hacen.

### ■ Incompresible - Compresible

Un fluido incompresible es aquel en el que podemos considerar su densidad como constante; en el caso contrario el fluido se denomina compresible.

### ■ Ideal - Viscoso

Un fluido ideal es aquel en el que el efecto de la viscosidad es tan débil que puede suponerse nula. Esto significa que podemos despreocupar la fuerza de rozamiento interna entre las diferentes capas de fluido. Por otro lado, los fluidos en los que el efecto dominante es precisamente la viscosidad, se los denomina viscosos<sup>3</sup>.

### ■ Newtonianos

Los fluidos newtonianos son aquellos en los que la viscosidad es constante, es decir, la relación entre la fuerza por unidad de superficie<sup>4</sup> que una capa de fluido ejerce sobre las vecinas y la velocidad de deformación es constante<sup>5</sup>.

## 2.2. Ecuaciones de conservación

Denominamos *Mecánica de Fluidos* a la rama de la mecánica que estudia el movimiento de los fluidos, las fuerzas que lo provocan y la interacción del mismo con el entorno que lo limita. Un fluido en movimiento puede describirse matemáticamente mediante las funciones de distribución de la velocidad y de dos magnitudes termodinámicas pertenecientes al fluido. De esta forma, usando cinco magnitudes (las tres componentes de la velocidad y por ejemplo, la presión y la densidad), el estado del fluido queda completamente determinado.

Las ecuaciones fundamentales de la mecánica de fluidos modelan el comportamiento de cualquier tipo de flujo mediante la conservación de dos cantidades, la masa y el momento.

---

<sup>3</sup>Todos los fluidos reales son viscosos, sin embargo ese término se reserva para para aquéllos en los que la viscosidad es la fuerza dominante.

<sup>4</sup>A la fuerza por unidad de superficie se le denomina *esfuerzo*.

<sup>5</sup>Dependiendo del tipo de relación que presenten los fluidos entre el esfuerzo y la velocidad de deformación existen varias clasificaciones de fluidos. Sin embargo sólo se mencionan los newtonianos por tratarse del tipo de fluidos con los que se trabajará en este proyecto.

A continuación se exponen las ecuaciones necesarias para entender la conservación de dichas magnitudes. Estas ecuaciones han sido simplificadas bajo ciertas hipótesis asumidas de forma muy común por gran parte de la comunidad científica.

Dichas hipótesis son:

- Medio continuo
- Efectos relativistas despreciables
- Ausencia de efectos cuánticos
- Validez en sistemas de referencia inerciales
- Interacción con campos magnéticos despreciable

### 2.2.1. Conservación de la materia

La *ley de conservación de la materia* se cumple con independencia de la naturaleza del fluido o de las fuerzas que actúen sobre el mismo. Esto implica que a lo largo de todo un fluido la masa no se destruye ni genera en ningún punto. Matemáticamente esta ley de conservación se expresa, en su forma diferencial, de la siguiente manera:

$$\frac{\partial \rho}{\partial t} + \vec{\nabla} \cdot (\rho \vec{v}) = 0 \quad (2.1)$$

donde  $\rho$  es la densidad del fluido y  $\vec{v}$  la velocidad del fluido en un punto  $(x, y, z)$  del espacio para un instante de tiempo  $t$ . El primer término de la expresión anterior representa la variación de masa por unidad de volumen y unidad de tiempo en cierto punto del espacio, y el segundo término hace referencia a la variación de velocidad de entrada y de salida de la materia en el sistema.

La expresión (2.1) es la denominada *ecuación de continuidad*, válida para todo tipo de fluidos. Esta expresión aparece en gran cantidad de campos de la física, como son La Mecánica Cuántica, la Teoría Electromagnética, la Mecánica Relativista etc.

### 2.2.2. Conservación del momento

La cantidad de movimiento o momento es una magnitud física vectorial que, del mismo modo que la materia, obedece a una ley de conservación. Para el caso de los fluidos, la segunda Ley de Newton afirma que la variación de la cantidad de movimiento por unidad de tiempo de una masa fluida es igual a la resultante de todas las fuerzas que actúen sobre dicha masa.

Las fuerzas que actúan sobre un fluido son de dos tipos, fuerzas de volumen y fuerzas de superficie. Si consideramos un elemento de volumen  $V_0$  rodeado por una superficie cerrada  $S_0$ , las fuerzas de volumen son las que no dependen de la interacción del fluido en  $V_0$  con el fluido que lo rodea y por tanto afectan a todo el elemento de volumen por igual. La fuerza gravitatoria, las fuerzas eléctricas y las magnéticas son ejemplos de estas fuerzas. En cambio, las fuerzas de superficie son fuerzas que dependen de la interacción del fluido en  $V_0$  con el fluido que lo rodea. Son fuerzas que se ejercen sobre el volumen  $V_0$  a través de la superficie  $S_0$ . Ejemplos de estas fuerzas son las fuerzas de presión, viscosidad o fricción. Teniendo en cuenta esto la ecuación de cantidad de movimiento se expresa como sigue:

$$\rho \frac{D\vec{v}}{Dt} = \vec{f}_v + \vec{\nabla} \cdot \bar{\bar{\tau}} \quad (2.2)$$

donde  $\vec{f}_v$  designa las fuerzas de volumen que actúan sobre el fluido y  $\bar{\bar{\tau}}$  es el tensor de tensiones asociado a las fuerzas de superficie.

En este caso el cociente  $D\vec{v}/Dt$  representa la variación con respecto al tiempo de la velocidad de una partícula fluida<sup>6</sup> moviéndose por el espacio, es decir, los ejes de coordenadas están situados sobre cada partícula fluida. La descripción del movimiento resulta de obtener las trayectorias que describen dichas partículas. Esta manera de describir los fluidos se denomina *descripción lagrangiana*, uno de los dos formalismos [9] que se utilizan habitualmente para describir macroscópicamente los fluidos. El segundo de los formalismos se denomina *descripción euleriana*, en la que los ejes se sitúan en puntos fijos del espacio, por lo que en este caso es la descripción instantánea de la velocidad en cada punto lo que se obtiene para representar el movimiento del fluido.

En lo que sigue adoptaremos un punto de vista euleriano, por lo que necesitamos conocer la relación entre ambas descripciones para obtener la ecuación (2.2) de forma que la derivada de la velocidad se refiera a puntos fijos del espacio. El concepto de *derivada sustancial*<sup>7</sup> [11] es la herramienta que nos permite pasar de un formalismo a otro. De forma intuitiva puede postularse que esa derivada refleja dos causas distintas del cambio; por un lado el hecho de que la velocidad puede estar cambiando con el tiempo de forma local; y por otro lado el hecho de que la partícula fluida pueda estar moviéndose en un campo en el que la velocidad no es uniforme. La siguiente ecuación expresa esos cambios en la velocidad que experimenta la partícula fluida

---

<sup>6</sup>Retomando la definición de medio continuo, entendemos partícula fluida como un volumen suficientemente pequeño para poder considerarlo geoméricamente un punto.

<sup>7</sup>A veces denominada también derivada total o derivada material.

a medida que se mueve por el campo de fluido.

$$\frac{D\vec{v}}{Dt} = \frac{\partial\vec{v}}{\partial t} + (\vec{v} \cdot \vec{\nabla}) \vec{v} \quad (2.3)$$

Esta ecuación es la definición [11] de derivada sustancial de la velocidad. Sustituyendo ahora en (2.2) obtenemos la expresión para la conservación del momento de un fluido viscoso referida a puntos fijos del espacio.

$$\rho \left[ \frac{\partial\vec{v}}{\partial t} + (\vec{v} \cdot \vec{\nabla}) \vec{v} \right] = \vec{f}_v + \vec{\nabla} \cdot \bar{\bar{\tau}} \quad (2.4)$$

El tensor de esfuerzos de (2.4) puede escribirse como suma de dos términos, uno que representa el tensor de esfuerzos normales producido por fuerzas externas sobre la superficie del sistema y otro el tensor de esfuerzos viscosos, esto es,

$$\bar{\bar{\tau}} = -p\bar{\bar{I}} + \bar{\bar{\tau}}' \quad (2.5)$$

donde  $p$  es la presión del fluido,  $\bar{\bar{I}}$  es el tensor unitario y  $\bar{\bar{\tau}}'$  el tensor de tensiones de esfuerzos viscosos, relacionado con el transporte molecular de cantidad de movimiento. El tensor de tensiones adopta diferentes formas dependiendo del tipo de fluido que se esté considerando. A continuación se verá la forma que adquieren los tensores cuando tratamos con fluidos ideales y con fluidos viscosos.

### Fluidos ideales

Como ya se ha mencionado, un fluido ideal es un fluido en que las fuerzas viscosas son consideradas nulas. En estas circunstancias la única fuerza que actúa sobre el fluido es la presión, de forma que la fuerza total sobre un cierto volumen es la integral  $-\oint p d\vec{S}$  de la presión a lo largo de la superficie que rodea el volumen. En este caso el tensor de tensiones (2.5) es una matriz diagonal con sus tres componentes iguales a  $-p$ , de modo que sustituyendo en (2.4) llegamos a la expresión de la conocida *ecuación de Euler* (2.6), una de las ecuaciones fundamentales de la mecánica de fluidos.

$$\rho \left[ \frac{\partial\vec{v}}{\partial t} + (\vec{v} \cdot \vec{\nabla}) \vec{v} \right] = -\vec{\nabla} p \quad (2.6)$$

Si nos encontramos en presencia de un campo gravitatorio, la ecuación (2.6) ha de ser modificada, de acuerdo a la ecuación (2.4), sumándole al gradiente de la presión un término correspondiente a la fuerza que ejerce la gravedad sobre cada elemento de volumen.

$$\frac{\partial\vec{v}}{\partial t} + (\vec{v} \cdot \vec{\nabla}) \vec{v} = -\frac{\vec{\nabla} p}{\rho} + \vec{g} \quad (2.7)$$

## Fluidos viscosos

Se pretende ahora obtener las ecuaciones del movimiento para un fluido viscoso. Para ello es necesario añadir al tensor de tensiones de un fluido ideal un término que refleje la transferencia de impulso viscoso, esto es, el término  $\bar{\tau}'$  de la ecuación (2.5). La forma más general de este tensor de rango dos se representa de la siguiente forma [15].

$$\bar{\tau}'_{ij} = \eta \left[ \frac{\partial v_j}{\partial x_i} + \frac{\partial v_i}{\partial x_j} - \frac{2}{3} (\vec{\nabla} \cdot \vec{v}) \bar{I} \right] + \zeta (\vec{\nabla} \cdot \vec{v}) \bar{I} \quad (2.8)$$

De este modo, añadiendo la divergencia del tensor (2.8) a la ecuación de Euler obtenemos la forma general de la ecuación del movimiento para un fluido viscoso.

$$\rho \left[ \frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \vec{\nabla}) \vec{v} \right] = -\vec{\nabla} p + \vec{\nabla} \cdot \left\{ \eta \left[ \frac{\partial v_j}{\partial x_i} + \frac{\partial v_i}{\partial x_j} - \frac{2}{3} (\vec{\nabla} \cdot \vec{v}) \bar{I} \right] + \zeta (\vec{\nabla} \cdot \vec{v}) \bar{I} \right\} \quad (2.9)$$

Las constantes  $\eta$  y  $\zeta$  se denominan primer y segundo coeficientes de viscosidad, o viscosidad dinámica y viscosidad volumétrica. Estas magnitudes son funciones de la presión y la temperatura, de modo que no son constantes en todo el fluido y no pueden extraerse fuera del operador divergencia. Sin embargo, en la mayoría de los casos, los coeficientes de viscosidad no varían de forma notable a lo largo del fluido y pueden considerarse constantes, obteniendo así una expresión más sencilla en forma vectorial para los fluidos viscosos.

$$\rho \left[ \frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \vec{\nabla}) \vec{v} \right] = -\vec{\nabla} p + \eta \nabla^2 \vec{v} + \left( \zeta + \frac{1}{3} \eta \right) \vec{\nabla} (\vec{\nabla} \cdot \vec{v}) \quad (2.10)$$

Si nos encontramos en un caso en el que podemos considerar el fluido incompresible, la densidad es constante a lo largo del mismo y la ecuación de continuidad (2.1) se reduce a  $\vec{\nabla} \cdot \vec{v} = 0$ . De este modo, el último término en el segundo miembro de (2.10) es cero y obtenemos la ecuación de Navier-Stokes:

$$\rho \left[ \frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \vec{\nabla}) \vec{v} \right] = -\vec{\nabla} p + \eta \nabla^2 \vec{v} \quad (2.11)$$

En este caso el tensor de esfuerzos queda reducido a una expresión sencilla ( $\bar{\tau}' = -\vec{\nabla} p + \eta \nabla^2 \vec{v}$ ) que sólo depende de la viscosidad dinámica  $\eta$ . No obstante, es frecuente encontrar las ecuaciones escritas en función del cociente  $\nu = \frac{\eta}{\rho}$ , denominado *viscosidad cinemática*.

$$\frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \vec{\nabla}) \vec{v} = -\frac{1}{\rho} \vec{\nabla} p + \nu \nabla^2 \vec{v} \quad (2.12)$$

La ecuación de Navier-Stokes (2.12) es un sistema de ecuaciones en derivadas parciales no lineales formado por tres ecuaciones, una para cada dirección del espacio. Se encuentran acopladas entre sí a través de la presión y la velocidad.

## El número de Reynolds

El número de Reynolds es un parámetro que aparece en numerosos problemas de la mecánica de fluidos debido a que permite trabajar con la ecuación de Navier-Stokes en su forma adimensional. Este número relaciona la viscosidad cinemática  $\nu$ , la velocidad principal de un problema  $v_0$ , y un tercer parámetro relacionado con las propiedades geométricas del mismo, denominado longitud característica  $l$ .

$$[\nu] = [m]^2/[s] = L^2 T^{-1}, \quad [l] = [m] = L, \quad [v_0] = [m]/[s] = L T^{-1}$$

Con estas tres magnitudes sólo podemos formar un único parámetro adimensional, a saber,  $v_0 l / \nu$ . Esta combinación, designada por  $Re$ , es lo que se conoce como número de Reynolds,

$$Re = \frac{v_0 l}{\nu} \quad (2.13)$$

y representa el ratio de las fuerzas viscosas frente a las de inercia. Números de Reynolds altos indican que las fuerzas de inercia predominan frente a las viscosas, y para números suficientemente altos,  $Re > 4000$ , el flujo será turbulento. Esto implica que el movimiento del fluido es muy sensible a cualquier perturbación, que se amplía rápidamente formando remolinos debido al carácter irregular del flujo. Por el contrario, para números de Reynolds bajos,  $Re < 2000$ , el flujo será laminar, lo que quiere decir que cualquier perturbación que el fluido pueda sufrir será amortiguada con facilidad y las capas de fluido adyacentes serán siempre paralelas. Los flujos que se encuentran entre esos dos valores de  $Re$  pertenecen a lo que se conoce como zona de transición.

El proceso de adimensionalizar la ecuación de Navier-Stokes (2.12) (y cualquier ecuación en general) consiste en suponer que cualquier variable dimensional  $\psi$  que intervenga puede escribirse como el producto de un valor típico constante,  $\psi_0$ , y una variable adimensional o variable reducida,  $\psi^*$ , de modo que:  $\psi = \psi_0 \psi^*$ . Si escribimos de este modo todas variables que intervienen en (2.12) ( $\vec{v}$ ,  $t$ ,  $\vec{r}$ ,  $\rho$ ,  $p$  y  $\nu$ ), adimensionalizamos los operadores nabla y gradiente, y sustituimos en la misma, obtenemos la ecuación de Navier-Stokes adimensionalizada<sup>8</sup>:

$$\frac{\partial \vec{v}^*}{\partial t^*} + (\vec{v}^* \cdot \vec{\nabla}^*) \vec{v}^* = \frac{1}{Re} \nabla^{*2} \vec{v}^* - \vec{\nabla}^* p^* \quad (2.14)$$

Como se acaba de mencionar, las variables marcadas con un asterisco son las nuevas variables adimensionalizadas con las que se trabajará de aquí en adelante. Sin embargo, a partir de este momento, por simplicidad en la notación prescindiremos de los asteriscos, aunque teniendo en mente que las variables que se van a resolver son las nuevas variables reducidas.

---

<sup>8</sup>Podemos encontrar el desarrollo completo del paso de (2.12) a (2.14) en la referencia [11].

## 2.3. Ecuaciones simplificadas

Como ya se ha mencionado anteriormente, es necesario adaptar las ecuaciones de continuidad y de Navier-Stokes a nuestro caso particular y realizar simplificaciones adicionales, hasta obtener las ecuaciones que finalmente se han implementado en el código.

A continuación veremos en detalle dichas simplificaciones:

- Las propiedades físicas del fluido serán constantes.

La consecuencia directa de esto es que no se podrán estudiar casos en los que el fluido sea compresible, es decir, se asume densidad constante. Por otro lado la viscosidad del fluido también será constante, por lo que la simulación queda limitada a un rango de temperaturas en las que las propiedades físicas no varíen notablemente.

- Se considerarán fluidos en régimen laminar.

Esto implica que no se podrán estudiar flujos turbulentos. Los fluidos quedan restringidos a aquellos con números de Reynolds menores de 2000.

- Se trabajará con fluidos newtonianos.

En este punto eliminamos todos aquellos fluidos en los que la relación entre el esfuerzo y la velocidad de deformación no sea constante. Por ejemplo la miel, los metales pesados, la sangre y otros fluidos “pegajosos” quedarían excluidos.

- Se tratarán problemas bidimensionales.

Aquí es donde limitamos de manera notable el número de casos que podemos simular. Sin embargo la esencia de la fenomenología es la misma en dos que en tres dimensiones, y añadir una tercera dimensión complica tanto la explicación como la visualización de resultados. En definitiva, la riqueza de descripción del movimiento que aporta el caso 3D no compensa el esfuerzo requerido en la programación y el mayor coste computacional de las simulaciones.

Teniendo en cuenta todo lo anterior, se concluye que las ecuaciones que se implementarán en el programa son las siguientes.

$$\boxed{\frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \vec{\nabla} \vec{v}) = \frac{1}{Re} \nabla^2 \vec{v} - \vec{\nabla} p} \quad (2.15)$$

$$\boxed{\vec{\nabla} \cdot \vec{v} = 0} \quad (2.16)$$

A partir de este momento consideramos que el vector velocidad tiene solamente dos dimensiones,  $\vec{v} = (u, v)$ .

## 2.4. Condiciones de contorno

En las secciones anteriores se han expuesto las ecuaciones generales que gobiernan el movimiento de los fluidos. Sin embargo, cuando se reduce el problema a un dominio espacial concreto, la información para obtener cualquier solución particular del sistema de ecuaciones (2.12) es insuficiente. Es por ello que resulta fundamental especificar las condiciones de contorno que informen del comportamiento del fluido en las fronteras.

Revisaremos ahora las condiciones de contorno más habituales para un fluido viscoso, empezando por la llamada condición de *no-deslizamiento* (no-slip). Esta condición se aplica a la superficie de sólidos y asume que la velocidad relativa entre la superficie del sólido y el fluido infinitamente próximo a ésta es cero.

Por otro lado se encuentran las condiciones de contorno tipo *Dirichlet*. La información que aportan este tipo de condiciones es directa, es decir, el valor en la frontera de una variable genérica  $\phi$  se considera conocido,  $\phi = f$ . Para el caso de un fluido, generalmente esta condición se aplica al flujo entrante e implica que una de las componentes de la velocidad es conocida.

Por último, otra de las condiciones de contorno más usuales son las llamadas condiciones *Newman*, que aportan información de tipo indirecta. De forma general se conoce el valor de la derivada de la variable  $\phi$  en una dirección normal  $n$ ,  $\partial\phi/\partial n = f$ , en la frontera. Normalmente, estas condiciones se encuentran localizadas en una zona en la que el fluido es unidireccional y por lo tanto no hay cambios en la componente de la velocidad a lo largo de esa dirección.

A lo largo de este capítulo hemos visto de forma detalla los fundamentos y conceptos necesarios para comprender la fenomenología de la mecánica de fluidos. El siguiente paso es establecer los fundamentos de la mecánica de fluidos computacional para ser capaces de traducir las ecuaciones que intervengan en cada problema al lenguaje computacional. Por tanto, en el siguiente capítulo se explicarán los pasos necesarios para llevar a cabo dicha *traducción*.

## Capítulo 3

# Implementación

---

Como se ha visto en el capítulo anterior, el conjunto de ecuaciones de Navier-Stokes y la ecuación de continuidad conforman un complejo sistema de ecuaciones diferenciales en derivadas parciales acopladas. Es por ello que, a excepción de casos muy simples, aún no se han hallado soluciones analíticas y la necesidad de una resolución numérica resulta evidente.

El proceso de obtener la solución numérica computacionalmente involucra tres etapas, desacoplar, discretizar y resolver las ecuaciones. En la primera etapa se trata de desacoplar la velocidad de la presión en las ecuaciones, esto es, buscar un método adecuado que nos permita obtener un conjunto de expresiones equivalentes a las ecuaciones de Navier-Stokes y continuidad, en las que las incógnitas ( $\vec{v}$  y  $p$ ) se puedan resolver por separado. En la segunda etapa se transforman las ecuaciones diferenciales en derivadas parciales y sus condiciones de contorno en un sistema de ecuaciones algebraicas discretas, tanto en el tiempo como en el espacio. Por último, la tercera etapa supone la implementación de métodos numéricos para resolver computacionalmente los sistemas de ecuaciones algebraicas ya discretizadas y desacopladas.

En este capítulo se estudiará, por una parte el llamado método de pasos fraccionados (Fractional-Step Method-FSM), método utilizado para desacoplar la presión de la velocidad en las ecuaciones de Navier-Stokes, y por otra parte el método de Volúmenes Finitos, utilizado para discretizar espacialmente las ecuaciones. Durante el desarrollo del FSM se explicarán también los esquemas de discretización utilizados para la parte temporal.

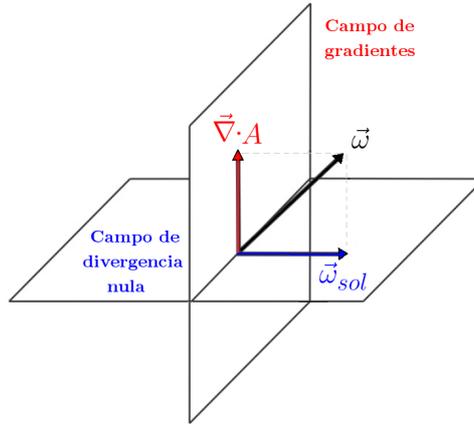
---

### 3.1. Método de pasos fraccionados (FSM)

El FSM fue primero introducido por Chorin [5] y unos años después Kim y Moin [14] realizaron una modificación para extenderlo al método de los Volúmenes Finitos. Este método utiliza el teorema de descomposición de Helmholtz-Hodge para resolver el acoplamiento de la presión y la velocidad en fluidos incompresibles. La idea es obtener dos ecuaciones separadas, una para la velocidad y otra para la presión. Para ello el método utiliza una velocidad intermedia con la que se resuelve la presión y después se calcula la velocidad final corrigiendo la velocidad intermedia con la presión. A continuación veremos de forma más detallada el funcionamiento del método, empezando por el teorema de Helmholtz-Hodge.

El teorema de Helmholtz-Hodge [6] establece que cualquier campo vectorial  $\vec{\omega}$  definido sobre un contorno cerrado puede ser descompuesto de manera única en una parte de divergencia nula o parte solenoidal  $\vec{\omega}_{sol}$  y una parte irrotacional  $\vec{\omega}_{irr}$  (ver figura 3.1), de la siguiente forma:

$$\vec{\omega} = \vec{\omega}_{sol} + \vec{\omega}_{irr} = \vec{\omega}_{sol} + \vec{\nabla}A \quad (3.1)$$



**Figura 3.1** – Descomposición única del campo vectorial  $\vec{\omega}$ .

ya que  $\vec{\nabla} \times \vec{\nabla}A = 0$  para cualquier campo escalar  $A$ . Si aplicamos el operador divergencia a ambos lados de (3.1) obtenemos la ecuación de Poisson para el campo escalar  $A$ ,

$$\vec{\nabla} \cdot \vec{\omega} = \nabla^2 A \quad (\text{ya que } \vec{\nabla} \cdot \vec{\omega}_{sol} = 0) \quad (3.2)$$

y si el campo  $\vec{\omega}$  es conocido, podemos resolver la ecuación anterior utilizando la relación siguiente:

$$\vec{\omega}_{sol} = \vec{\omega} - \vec{\nabla}A \quad (3.3)$$

Una vez explicado el teorema de Helmholtz-Hodge lo aplicamos a las ecuaciones de Navier-Stokes para resolver el acoplamiento de la presión y la velocidad y obtener dos ecuaciones resolubles, una para cada variable mencionada. Si observamos las ecuaciones simplificadas que obtuvimos en el capítulo anterior,

$$\frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \vec{\nabla}) \vec{v} = \frac{1}{Re} \nabla^2 \vec{v} - \vec{\nabla} p \quad (3.4)$$

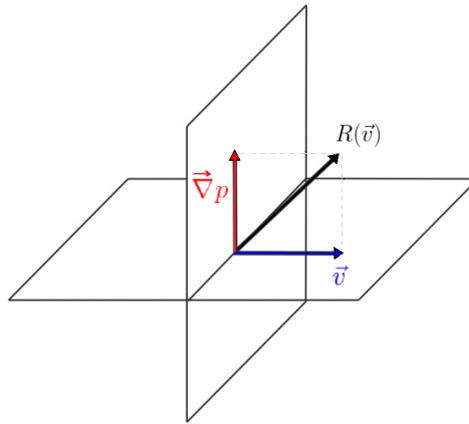
$$\vec{\nabla} \cdot \vec{v} = 0 \quad (3.5)$$

vemos que el término de la presión se corresponde con la parte irrotacional que mencionábamos ( $\vec{\nabla} p \sim \vec{\nabla} A$ ), y la velocidad con el campo de divergencia nula ( $\vec{\nabla} \cdot \vec{v} \sim \vec{\nabla} \cdot \vec{\omega}_{sol}$ ). Por tanto el método consiste en definir una velocidad intermedia que corresponde con el campo solenoidal y que es función de un vector que definiremos como  $R(\vec{v})$ <sup>9</sup> (ver ecuación (3.6)). A continuación, si despejamos el término de la presión en (3.4) y aplicamos el operador divergencia, obtenemos la ecuación de Poisson para la presión,

$$\nabla^2 p = \vec{\nabla} \cdot \underbrace{\left[ -(\vec{v} \cdot \vec{\nabla}) \vec{v} + \frac{1}{Re} \nabla^2 \vec{v} \right]}_{R(\vec{v})} \quad (3.6)$$

y utilizando  $R(\vec{v})$  obtenemos la ecuación análoga a (3.3) para las ecuaciones de Navier-Stokes (ver figura 3.2):

$$\vec{v} = R(\vec{v}) - \vec{\nabla} p \quad (3.7)$$



**Figura 3.2** – Descomposición única de las ecuaciones de Navier-Stokes en un campo irrotacional ( $\vec{\nabla} p$ ) y uno de divergencia nula ( $\vec{v}$ ).

<sup>9</sup>Este término representa los términos convectivos (primer sumando) y difusivos (segundo sumando) de la ecuación de Navier-Stokes.

De esta forma, con el campo de velocidades inicial (que es conocido), podemos calcular directamente la velocidad intermedia,  $\vec{v}^* = f(R(\vec{v}))$ , utilizarla para resolver la ecuación de Poisson, y por último corregir  $\vec{v}^*$  con el gradiente de presión para calcular la velocidad final  $\vec{v}$  del fluido. Explicado de forma sencilla esta es la idea fundamental del funcionamiento del FSM, para una información más detallada sobre el desarrollo matemático consultar el apéndice B.

### Discretización temporal

En este punto del desarrollo del FSM debemos especificar los esquemas temporales que se van a utilizar, puesto que la formulación final del método depende de los mismos. Existen numerosos esquemas temporales, pero a grandes rasgos se pueden dividir en dos grupos:

- **Esquemas explícitos.** Los esquemas explícitos son aquellos que asumen que los valores de las variables en el nuevo instante de tiempo dependen únicamente del valor de la variable en el instante anterior. Son esquemas que *aproximan* los valores nuevos a partir de los calculados, por lo que requieren un menor coste computacional, pero a su vez la estabilidad de los mismos también es menor.
- **Esquemas implícitos.** En los esquemas implícitos el valor de las variables en el nuevo instante de tiempo ya no se aproxima, se *calcula* de forma precisa a partir tanto del instante anterior como del siguiente. El coste computacional en este caso es bastante mayor en cada iteración, pero cuentan con la ventaja de que la estabilidad del método es muy alta.

Aquí utilizaremos esquemas explícitos para reducir el tiempo de computacional, y para resolver el problema de la estabilidad de utilizará el criterio de Courant para determinar el paso de tiempo:

$$Co = \frac{|u| \Delta t}{\Delta x} \leq 1 \quad (3.8)$$

donde Co es el número de Courant, también llamado número de Courant, Friedrich y Levy [8].

Siguiendo ahora con la elección de los esquemas temporales para el desarrollo del FSM, en nuestro caso hemos optado por el esquema de bajo orden CDS (Central Difference Scheme o Esquema de las Diferencias Centradas) para la discretización del término de la derivada temporal,

$$\left. \frac{\partial \vec{v}}{\partial t} \right|_{n+1/2} \approx \frac{\vec{v}^{n+1} - \vec{v}^n}{\Delta t} \quad (3.9)$$

un esquema explícito de segundo orden conocido como esquema de Adams-Bashforth [1] para los términos agrupados en  $R(\vec{v})$ ,

$$R^{n+1/2}(\vec{v}) \approx \frac{3}{2} R(\vec{v}^n) - \frac{1}{2} R(\vec{v}^{n-1}) \quad (3.10)$$

y por último un esquema de primer orden hacia atrás de Euler para el término del gradiente de presión:

$$\frac{\vec{v}^{n+1} - \vec{v}^n}{\Delta t} = \frac{3}{2} R(\vec{v}^n) - \frac{1}{2} R(\vec{v}^{n-1}) - \vec{\nabla} p^{n+1} \quad (3.11)$$

A continuación, si reorganizamos en la ecuación (3.11), obtenemos la velocidad intermedia  $\vec{v}^*$ , representada por el primer término de la siguiente expresión:

$$\vec{v}^{n+1} + \vec{\nabla} p^{n+1} \Delta t = \vec{v}^n + \Delta t \left[ \frac{3}{2} R(\vec{v}^n) - \frac{1}{2} R(\vec{v}^{n-1}) \right] \quad (3.12)$$

↓

$$\vec{v}^* = \vec{v}^{n+1} + \vec{\nabla} p^{n+1} \Delta t \quad (3.13)$$

De esta forma obtenemos una expresión para determinar la velocidad real  $\vec{v}^{n+1}$ .

El último paso consiste en aplicar a la ecuación (3.13) el operador divergencia y el criterio de incompresibilidad para obtener la ecuación de Poisson en forma discreta, concluyendo así con la formulación del método.

$$\vec{\nabla} \cdot \vec{v}^* = \Delta t \nabla^2 p^{n+1} \quad (3.14)$$

### 3.1.1. Pasos del FSM

A continuación resumiremos los pasos de resolución del FSM. De este modo veremos tanto el funcionamiento global que seguirá el programa como las ecuaciones con las que trabajaremos de aquí en adelante para concluir el proceso de discretización.

1. Evaluar los términos difusivos y convectivos mediante el cálculo de  $R(\vec{v}^n)$  utilizando el campo de velocidades inicial.

$$R(\vec{v}^n) = -(\vec{v}^n \vec{\nabla}) \vec{v}^n + \frac{1}{Re} \nabla^2 \vec{v}^n$$

2. Determinar la velocidad intermedia.

$$\vec{v}^* = \vec{v}^n + \Delta t \left[ \frac{3}{2} R(\vec{v}^n) - \frac{1}{2} R(\vec{v}^{n-1}) \right]$$

3. Resolver la ecuación de Poisson para obtener el campo de presiones.

$$\nabla^2 p^{n+1} = \frac{\vec{\nabla} \cdot \vec{v}^*}{\Delta t}$$

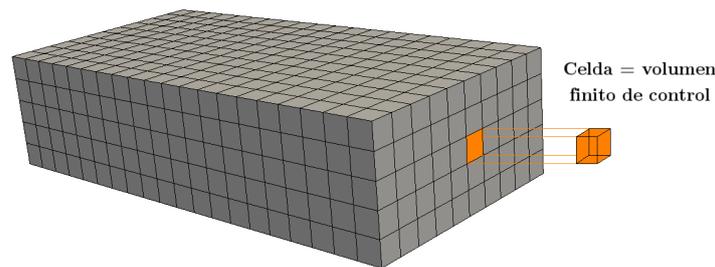
4. Corregir la velocidad intermedia con la presión del paso anterior y evaluar la velocidad real del fluido en el instante  $n + 1$ .

$$\vec{v}^{n+1} = \vec{v}^* - \vec{\nabla} p^{n+1} \Delta t$$

## 3.2. Discretización espacial

### 3.2.1. Método de Volúmenes Finitos

El *método de los volúmenes finitos* se basa en la discretización directa de las ecuaciones de conservación de la masa, cantidad de movimiento y la energía. Este método integra las ecuaciones en cada uno de volúmenes de control en que se divide el dominio de trabajo (ver figura 3.3), asociando a cada uno de ellos un punto nodal en el centro. Las integrales de superficie que aparecen en las ecuaciones se aproximan por la suma de los flujos que atraviesan cada una de caras del poliedro; el resultado es una ecuación discretizada que relaciona los valores de las variables en un grupo de puntos nodales.



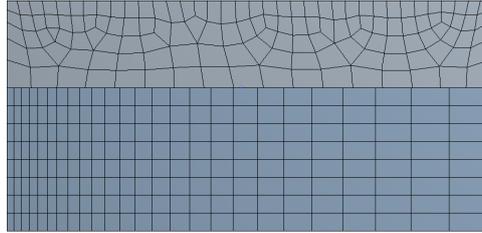
**Figura 3.3** – Representación de un dominio discretizado con el método de volúmenes finitos.

La principal ventaja del método de los volúmenes finitos es que la discretización espacial se lleva a cabo directamente en el espacio físico del problema. Por lo tanto, no hay problemas con la transformación entre sistemas de coordenadas, como ocurre en el método de las diferencias finitas. Otra ventaja del método es la gran flexibilidad que ofrece a la hora de usarlo tanto en mallas estructuradas como no estructuradas. Debido a todo esto este método es uno de los más utilizados en simulaciones CFD.

### 3.2.2. Mallado

El primer paso en la discretización espacial consiste en la discretización del dominio de trabajo, esto es, decidir el tipo de malla más adecuada a nuestro problema, teniendo en cuenta las ventajas y limitaciones de cada una de ellas.

Una malla de resolución se define como el conjunto discreto de puntos (también denominados nodos) pertenecientes al dominio continuo de trabajo y sobre los cuales se hayan las soluciones aproximadas del problema. Dependiendo de la forma y tamaño relativo de los nodos que componen una malla, se distinguen dos tipos, las mallas estructuradas y las no estructuradas. Dentro de las mallas estructuradas podemos encontrar mallas uniformes o no uniformes, dependiendo de regularidad del espaciado entre celdas. La figura 3.4 muestra un ejemplo de los tres tipos de mallas mencionados.

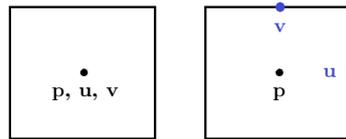


**Figura 3.4** – Ejemplo de malla uniforme no estructurada (malla inferior horizontal), uniforme estructurada (malla inferior vertical) y no estructurada (malla superior).

La ventaja de las mallas estructuradas es que su geometría es sencilla y por ende son más fáciles e intuitivas de programar. Sin embargo, no son perfectamente adaptables a las regiones de interés y no se puede densificar una zona de interés sin aumentar el número de puntos a lo largo de toda la malla. El punto fuerte de las mallas no estructuradas es precisamente que son totalmente moldeables según sea la geometría del problema, pero su implementación en el ordenador es bastante más costosa. Por ello, es necesario definir bien nuestro propósito y decidir qué tipo de malla es más conveniente, teniendo en cuenta las limitaciones de cada una.

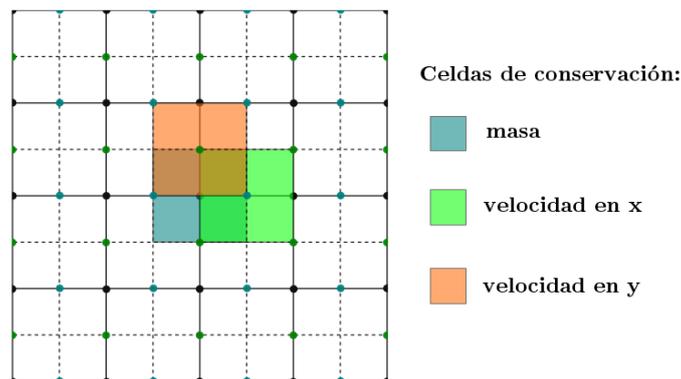
En nuestro caso sólo se tratará con geometrías sencillas y por tanto se utilizarán mallas estructuradas uniformes. La diferencia de calidad aportada por las mallas no estructuradas no compensa, para el alcance de este proyecto, el esfuerzo requerido.

Por otro lado, si nos fijamos en la situación de los nodos en las mallas, se distinguen dos tipos: las mallas centradas y las mallas desplazadas (*staggered*). Las centradas se caracterizan por trabajar con todas las variables definidas en el nodo central del volumen de control, mientras que las mallas desplazadas trabajan con la presión en el centro del volumen y las componentes de la velocidad se desplazan media celda hacia la cara de la misma, en cada dirección correspondiente.



**Figura 3.5** – Ejemplo de malla centrada (izquierda) y malla desplazada (derecha).

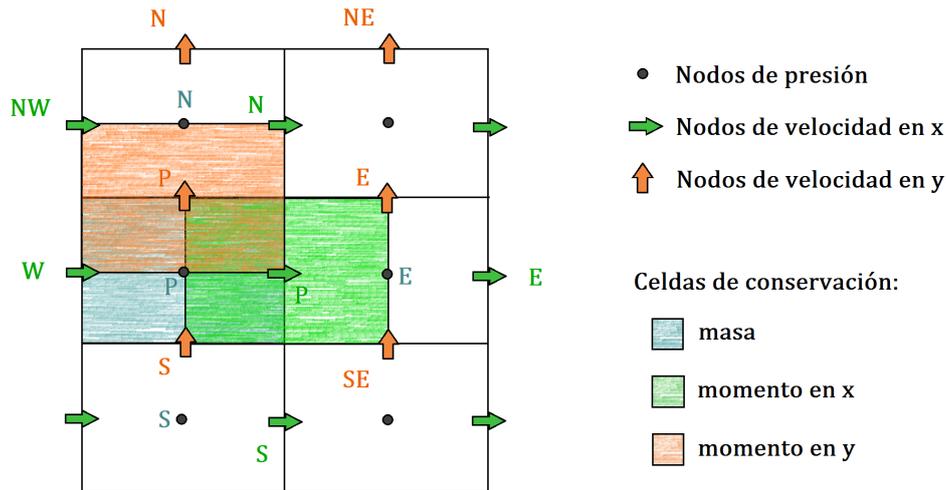
Trabajar con una malla desplazada es equivalente a trabajar con varias mallas centradas, una para cada variable, tal y como se ve en la figura 3.6.



**Figura 3.6** – Ejemplo de malla desplazada en 2D.

La elección obvia sería trabajar con una malla centrada, con todas las variables almacenadas en el mismo punto y con los mismos volúmenes de control. Sin embargo este tipo de mallas siempre ha tenido en contra las dificultades con el acoplamiento presión-velocidad en fluidos incompresibles y la aparición de oscilaciones en la presión [10]. Debido a esto, desde que se introdujeron las mallas desplazadas en 1980, la disposición centrada no ha sido apenas usada, ya que este tipo de mallas soluciona los problemas mencionados y aporta otras ventajas [10].

En concreto el uso de una disposición desplazada tal y como la que aparece en la figura 3.5 presenta la ventaja de que muchos de los términos que requieren interpolación con la malla centrada pueden ser calculados directamente sin interpolación. Podemos ver esto en la figura 3.7. Tanto la presión como los términos difusivos pueden ser aproximados por diferencias centradas sin interpolación, ya que la presión se encuentra en el centro de la celda y las derivadas de la velocidad (necesitadas para el término difusivo), ya se encuentran calculadas en las caras.

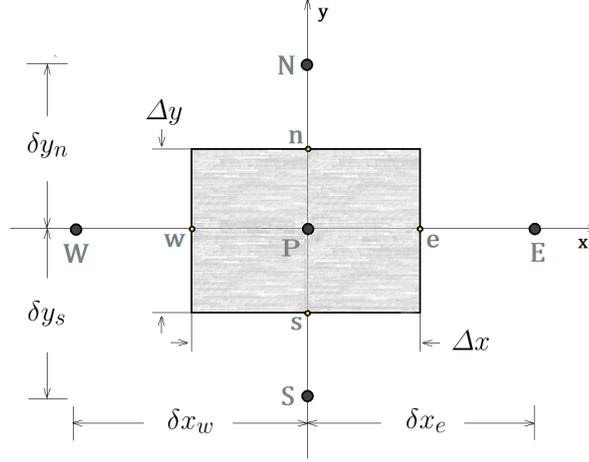


**Figura 3.7** – Esquema representativo de la malla utilizada (uniforme, estructurada y desplazada), así como la colocación de cada uno de los puntos de las diferentes variables.

Teniendo en cuenta todo lo anterior, trabajaremos con una malla uniforme estructurada y desplazada. En la figura 3.7 se muestra un esquema de la malla que se ha implementado, así como la notación utilizada (la más habitual) de las celdas y sus colindantes. Dado un punto cualquiera  $P$ , sus vecinos se denotarán según las iniciales en inglés de los puntos cardinales, N, S, E y W. Las caras de las celdas se denotan de la misma manera pero con letras minúsculas.

### 3.2.3. Discretización espacial de las ecuaciones

Una vez definida la malla de trabajo, el segundo paso consiste en discretizar las ecuaciones obtenidas en el apartado de “Pasos del FSM”, eliminando así las derivadas espaciales involucradas. Para ello aplicaremos el método de Volúmenes Finitos, de manera que integraremos las ecuaciones sobre una celda genérica como la que se muestra en la figura 3.8, válida tanto si integramos una componente  $u$  o otra de la velocidad. En un principio deberíamos integrar primero todas las ecuaciones referidas a una componente y después a la otra, cada una integrada en su correspondiente malla de trabajo (figura 3.7). Sin embargo podemos simplificar el proceso integrando las ecuaciones para una variable genérica  $\phi = \vec{v} = (u, v)$  y llegado el momento especificando qué magnitudes han de ser sustituidas u obtenidas de cada malla en particular.



**Figura 3.8** – Diagrama de una celda genérica y la nomenclatura utilizada para designar los parámetros de la misma.

Dicho esto, las ecuaciones que debemos discretizar son las siguientes:

$$R(\phi) = -(\phi \vec{\nabla}) \phi + \frac{1}{Re} \nabla^2 \phi \quad (3.15)$$

$$\nabla^2 p = \frac{\vec{\nabla} \cdot \phi^*}{\Delta t} \quad (3.16)$$

A continuación se mostrará el proceso de discretización llevado a cabo sólo para  $R(\phi)$  debido a que los operadores que intervienen en las dos ecuaciones son los mismos y la discretización es equivalente. La única diferencia, como hemos mencionado antes, es la malla de trabajo, que para la ecuación (3.16) sería la malla de los nodos de presiones.

### Discretización de $R(\phi)$ :

A) Parte convectiva  $R(\phi)_{conv} = (\phi \vec{\nabla}) \phi$  :

$$\begin{aligned} \iint_S R(\phi)_{conv} dS &= \int_s^n \int_w^e \left( u \frac{\partial \phi}{\partial x} + v \frac{\partial \phi}{\partial y} \right) dx dy = \\ &= \int_s^n \left[ (u \phi)_e - (u \phi)_w + v \frac{\partial \phi}{\partial y} \overbrace{(x_e - x_w)}^{\Delta x} \right] dy = \\ &= [(u \phi)_e - (u \phi)_w] \underbrace{(y_n - y_s)}_{\Delta y} + [(v \phi)_n - (v \phi)_s] \Delta x \end{aligned} \quad (3.17)$$

Como vemos en la expresión anterior, hemos aproximado las integrales de superficie por la suma de los flujos que atraviesan las caras,  $\phi_i = \phi_{e,w,n,s}$ . Ahora bien, los puntos  $e, w, n$  y  $s$  no son puntos pertenecientes al dominio de trabajo de la malla y necesitamos ponerlos en función de los puntos  $E, W, N$  y  $S$ . Usando de nuevo el esquema de diferencias centradas

podemos aproximar:

$$\phi_i = \frac{\phi_P + \phi_I}{2}$$

donde  $\phi_P$  y  $\phi_I = \phi_{E,W,N,S}$  ahora sí se corresponden con los valores de los flujos en puntos de la malla. Sustituyendo en (3.17) obtenemos,

$$R(\phi)_{conv} = \frac{\Delta y}{2} [u_e (\phi_P + \phi_E) - u_w (\phi_P + \phi_W)] + \frac{\Delta x}{2} [v_n (\phi_P + \phi_N) - v_s (\phi_P + \phi_S)] \quad (3.18)$$

B) Parte difusiva  $R(\phi)_{dif} = \nabla^2 \phi$  :

$$\begin{aligned} \iint_S \nabla^2 \phi \, dS &= \int_s^n \int_w^e \left( \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} \right) dx \, dy \\ &= \int_s^n \left[ \left( \frac{\partial \phi}{\partial x} \right)_e - \left( \frac{\partial \phi}{\partial x} \right)_w + \frac{\partial^2 \phi}{\partial y^2} \Delta x \right] dy = \\ &= \Delta y \left[ \left( \frac{\partial \phi}{\partial x} \right)_e - \left( \frac{\partial \phi}{\partial x} \right)_w \right] + \Delta x \left[ \left( \frac{\partial \phi}{\partial y} \right)_n - \left( \frac{\partial \phi}{\partial y} \right)_s \right] \end{aligned} \quad (3.19)$$

Esta vez para obtener la expresión en función de los puntos del dominio usamos de nuevo diferencias centradas de la siguiente forma:

$$\left( \frac{\partial \phi}{\partial x} \right)_i = \frac{\phi_I - \phi_P}{\underbrace{x_I - x_P}_{\delta x_i}} \quad \left( \frac{\partial \phi}{\partial y} \right)_i = \frac{\phi_I - \phi_P}{\underbrace{y_I - y_P}_{\delta y_i}} \quad (3.20)$$

Sustituyendo en (3.20) llegamos a la expresión para la parte difusiva,

$$R(\phi)_{dif} = \Delta y \left[ \left( \frac{\phi_E - \phi_P}{\delta x_e} \right) - \left( \frac{\phi_P - \phi_W}{\delta x_w} \right) \right] + \Delta x \left[ \left( \frac{\phi_N - \phi_P}{\delta y_n} \right) - \left( \frac{\phi_P - \phi_S}{\delta y_s} \right) \right] \quad (3.21)$$

El siguiente paso es sumar los términos convectivo y el difusivo teniendo en cuenta que  $R(\phi) = -R(\phi)_{conv} + (1/Re) R(\phi)_{dif}$ . De modo que obtenemos:

$$\begin{aligned} R(\phi) &= \frac{\Delta y}{2} [u_w (\phi_P + \phi_W) - u_e (\phi_P + \phi_E)] + \frac{\Delta x}{2} [v_s (\phi_P + \phi_S) - v_n (\phi_P + \phi_N)] + \\ &= \frac{\Delta y}{Re} \left[ \left( \frac{\phi_E - \phi_P}{\delta x_e} \right) - \left( \frac{\phi_P - \phi_W}{\delta x_w} \right) \right] + \frac{\Delta x}{Re} \left[ \left( \frac{\phi_N - \phi_P}{\delta y_n} \right) - \left( \frac{\phi_P - \phi_S}{\delta y_s} \right) \right] \end{aligned} \quad (3.22)$$

Podemos expresar la ecuación anterior de forma compacta como sigue,

$$R(\phi) = a_W \phi_W + a_E \phi_E + a_N \phi_N + a_S \phi_S - a_P \phi_P \quad (3.23)$$

siendo los coeficientes:

$$a_W = \Delta y \left( \frac{1}{Re \delta x_w} + \frac{u_w}{2} \right) \quad (3.24a)$$

$$a_E = \Delta y \left( \frac{1}{Re \delta x_e} - \frac{u_e}{2} \right) \quad (3.24b)$$

$$a_N = \Delta x \left( \frac{1}{Re \delta y_n} - \frac{v_n}{2} \right) \quad (3.24c)$$

$$a_S = \Delta x \left( \frac{1}{Re \delta y_s} + \frac{v_s}{2} \right) \quad (3.24d)$$

$$a_P = a_W + a_E + a_N + a_S \quad (3.24e)$$

Hasta este punto se ha discretizado el vector  $R(\phi)$  de forma genérica, por lo que las expresiones (3.24) y (3.25) son válidas para cualquier componente de la velocidad. Es aquí cuando debemos tener en cuenta a la hora de programar, si estamos tratando con una o con otra. A continuación, siguiendo la notación de las figuras 3.7 y 3.8, mostraremos unas tablas con los parámetros apropiados para sustituir en los coeficientes anteriores.

$\vec{v}$	$\Delta x$	$\Delta y$	$\delta x_e$	$\delta x_w$	$\delta y_n$	$\delta y_s$
$u$	$\Delta x^u$	$\Delta y$	$x_E^u - x_P^u$	$x_P^u - x_W^u$	$y_N - y_P$	$y_P - y_S$
$v$	$\Delta x$	$\Delta y^v$	$x_E - x_P$	$x_P - x_W$	$y_N^v - y_P^v$	$y_P^v - y_S^v$

$\vec{v}$	$u_e$	$u_w$	$v_n$	$v_s$	$\phi_P^{u,v}$
$u$	$(U_P + U_E)/2$	$(U_P + U_W)/2$	$(V_P + V_E)/2$	$(V_S + V_{ES})/2$	$(\phi_P + \phi_E)/2$
$v$	$(U_P + U_N)/2$	$(U_{WN} + U_W)/2$	$(V_P + V_N)/2$	$(V_P + V_S)/2$	$(\phi_P + \phi_N)/2$

**Cuadro 3.1** – Parámetros adecuados para sustituir en los coeficientes (3.25), dependiendo de la componente de la velocidad que tratemos. Los superíndices u y v significan que el parámetro se evalúa en la malla desplazada que corresponda.

### 3.3. Algoritmo de Resolución

Para concluir el capítulo de implementación veremos el proceso completo de resolución, explicando las relaciones entre los diferentes conceptos hasta ahora vistos y el orden que siguen las operaciones computacionales. La figura 3.9 muestra una perspectiva visual del algoritmo de resolución<sup>10</sup>, que podríamos dividir en cuatro partes:

1. Definición del caso.

La definición del caso es una parte esencial de todo el proceso. En este punto se transforma el problema físico elegido en un conjunto de ecuaciones discretizadas y se establecen las condiciones de contorno adecuadas. Así mismo se opta por una discretización espacial y temporal concretas, y por último se eligen los criterios de convergencia del programa.

---

<sup>10</sup>Puede consultarse el código MatLab completo en el apéndice A.

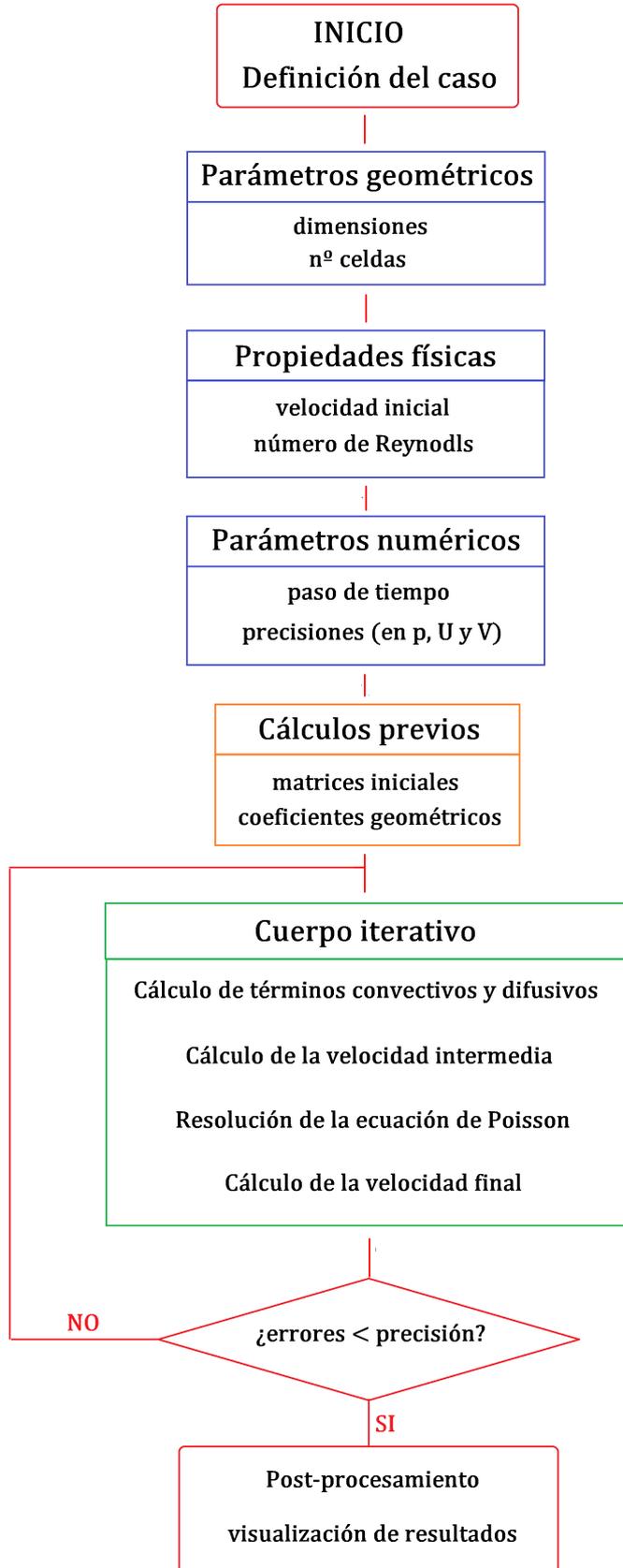


Figura 3.9 – Algoritmo completo de resolución.

## 2. Pre-procesamiento.

En esta segunda fase del proceso, una vez se ha estipulado la elección del caso, por un lado se introducen los datos físicos y parámetros numéricos relacionados con el mismo, y por otro lado se realizan los cálculos necesarios previos al cuerpo iterativo. De este modo, los pasos principales de esta fase serían los siguientes:

- introducir los parámetros geométricos necesarios para la creación de la malla, como son las dimensiones del problema y el número de celdas, obteniendo así el espaciado de la malla
- introducir los parámetros físicos, que dependerán de la elección del caso, aunque de forma general podrían ser la viscosidad, densidad, velocidad inicial, número de Reynolds etc.
- introducir los parámetros de integración numérica relacionados con la convergencia del problema, es decir, las precisiones deseadas para la presión y las componentes de la velocidad
- por último se calcula la matriz de coeficientes de la presión y se definen las matrices vacías de las componentes de la velocidad

## 3. Cuerpo iterativo.

El cuerpo iterativo es la parte más importante del algoritmo, donde se resuelven las ecuaciones del movimiento del problema dependiendo del método elegido (en nuestro caso el FSM). Esta parte se repite un gran cantidad de veces hasta que los criterios de convergencia se alcanzan. Como se ha comentado anteriormente, consta de los siguientes puntos:

- se calcula el vector  $R(\phi)$  para cada componente de la velocidad a partir del campo de velocidades inicial, obteniendo los términos convectivos y difusivos
- a partir de los términos anteriores se determina la velocidad intermedia  $\vec{v}^*$
- se resuelve la ecuación de Poisson y se determina la presión
- se corrige la velocidad intermedia con el gradiente de la presión y se obtiene la velocidad real del fluido en el instante de tiempo siguiente
- por último se establecen unos criterios de convergencia que nos aseguren que el programa termina cuando las velocidades y la presión ya no varían con el tiempo

4. Post-procesado.

El último punto del algoritmo lo denominamos post-procesado. Aquí se calculan las variables físicas que nos interesen en cada problema y se almacenan en ficheros los datos obtenidos en la simulación. Asimismo se realiza una subrutina que facilite la visualización de resultados.

## Capítulo 4

# Verificación del código

Una vez llegados a este punto está claro el proceso que hay que seguir para resolver las ecuaciones de Navier-Stokes y es el momento de validar el funcionamiento del código. Para ello se resuelve un caso de interés y se comparan los resultados con los disponibles en la bibliografía.

### 4.1. Presentación del caso

En nuestro caso se ha elegido un problema denominado “driven cavity”<sup>11</sup>. Este caso es uno de los mas importantes problemas de la dinámica de fluidos computacional, ya que se considera como el problema test por excelencia<sup>12</sup> en la valoración de nuevos métodos numéricos y en la validación de nuevos códigos para resolver Navier-Stokes. Este problema presenta las ventajas de una geometría y condiciones de contorno sencillas, una solución estable para el estado laminar y por último, la existencia de una gran cantidad de literatura sobre el desarrollo del mismo.

#### Driven cavity

El problema driven cavity consiste básicamente en una cavidad cerrada de longitud infinita<sup>13</sup>, con un fluido en su interior y una de las paredes moviéndose a velocidad constante, tal y como se ve en la figura 4.1. La masa en este sistema se conserva, por lo que no hay entrada ni salida de materia en la cavidad.

Este caso es perfectamente compatible con las especificaciones realizadas anteriormente en las ecuaciones, por lo que se trabajará con fluidos incompresibles y newtonianos en régimen

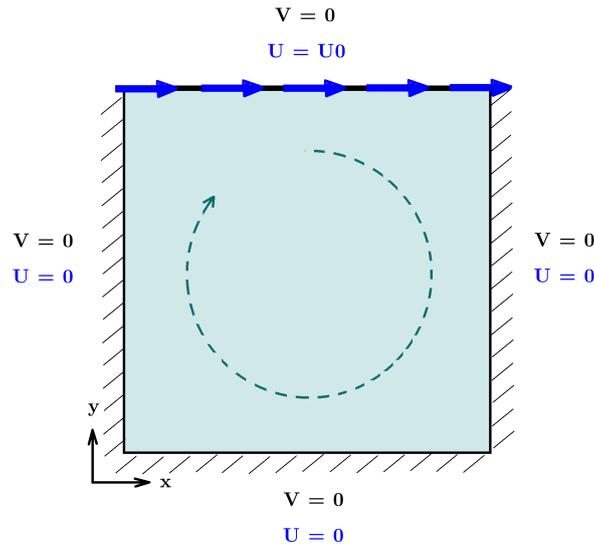
---

<sup>11</sup>En castellano podría traducirse como “cavidad con tapa deslizante”, sin embargo su nombre inglés es el más utilizado incluso en la literatura española.

<sup>12</sup>A pesar de las singularidades que se producen en dos de las esquinas del problema. Para más información consultar [13].

<sup>13</sup>lo que permite considerarlo un caso bidimensional.

laminar. Las ecuaciones del movimiento que resolveremos son, por lo tanto, las siguientes:



**Figura 4.1** – Esquema de la geometría y las condiciones de contorno del caso Driven Cavity.

$$\frac{\partial U}{\partial x} + \frac{\partial V}{\partial y} = 0 \quad (4.1)$$

$$\frac{\partial U}{\partial t} + U \frac{\partial U}{\partial x} + V \frac{\partial U}{\partial y} = \frac{1}{Re} \left( \frac{\partial^2 U}{\partial x^2} + \frac{\partial^2 U}{\partial y^2} \right) - \frac{\partial p}{\partial x} \quad (4.2)$$

$$\frac{\partial V}{\partial t} + U \frac{\partial V}{\partial x} + V \frac{\partial V}{\partial y} = \frac{1}{Re} \left( \frac{\partial^2 V}{\partial x^2} + \frac{\partial^2 V}{\partial y^2} \right) - \frac{\partial p}{\partial y} \quad (4.3)$$

$$Re = \frac{U_0 l}{\nu} \quad (4.4)$$

donde  $l$  es la longitud característica del caso, es decir, en nuestro caso la longitud de la cavidad,  $l = 1 \text{ m}$ .

Con respecto a las condiciones de contorno, como vemos en la figura 4.1, hay tres paredes fijas y una cuarta moviéndose a velocidad constante, por lo que las condiciones de contorno para la velocidad en las paredes fijas serán de tipo no-slip, es decir, todas las componentes de la velocidad iguales a cero a excepción de la componente  $U$  en la pared móvil, que tendrá una condición de tipo Dirichlet, esto es, un valor conocido  $U = 1 \text{ m/s}$ . Las condiciones de contorno para la presión son de tipo Neumann, lo que implica que la derivada de la presión en las paredes es nula.

Podemos predecir de forma cualitativa cual es la física que tiene que obedecer el problema. A medida que la pared (o tapa) se mueva, comenzará a arrastrar fluido hasta que se tope con la pared derecha de la caja, que obligará al fluido a descender. Lo mismo ocurrirá con las otras dos paredes restantes, de forma que se produzca una recirculación del flujo, como se ve en la figura 4.1. Por otra parte, cerca de la pared deberá producirse un gradiente de presiones, de forma que una presión positiva en una de las esquinas aleje el flujo de ella, y una presión negativa en la otra esquina lo acerque. Para más información detallada sobre este caso consultar [3] y [12].

## 4.2. Resultados

A continuación se muestran los resultados obtenidos con el programa creado para el problema driven cavity. Por una parte se realizará un estudio de la influencia de la densificación de la malla, y una vez elegida la malla óptima, se realiza un estudio del problema en función del número de Reynolds. Como ya se ha visto, el número de Reynolds nos indica cómo de predominantes son las fuerzas viscosas en cada fluido, por lo que realizando dicho análisis, veremos el comportamiento del fluido en la cavidad en función de la viscosidad del mismo. Se variará el número de Reynolds desde  $Re = 100$  (mayor viscosidad) hasta  $Re = 2000$  (menor viscosidad), para asegurar siempre la permanencia del fluido dentro del régimen laminar. Los resultados obtenidos con el programa creado en MatLab serán comparados con los obtenidos con un software comercial denominado ANSYS Fluent, uno de los paquetes de software más completos y eficientes en la resolución de problemas CFD que existen en el mundo hoy en día.

Se va a propagar la simulación partiendo del fluido en la cavidad en reposo hasta alcanzar el estado estacionario, es decir, un estado en el que el fluido ya no varíe explícitamente con el tiempo. El criterio de convergencia que se va a utilizar para determinar si la simulación se encuentra en el estado estacionario es el siguiente:

$$\frac{\|U^{n+1}\| - \|U^n\|}{\|U^{n+1}\|} < k \quad (4.5)$$

$$\frac{\|V^{n+1}\| - \|V^n\|}{\|V^{n+1}\|} < k \quad (4.6)$$

Es decir, estamos imponiendo que el error relativo entre cada componente de la velocidad en un instante de tiempo y en el instante anterior sea menor que  $k$ , un error que consideremos adecuado. Se ha empezado utilizando un  $k = 5 \cdot 10^{-6}$ , criterio que consideramos aceptable, y más adelante, si fuera necesario, veremos qué ocurre al utilizar un  $k$  menor. Se van a comparar

las distribuciones de velocidad y presión y el tiempo que tardan las simulaciones en alcanzar el criterio de convergencia, utilizando tanto MatLab como ANSYS.

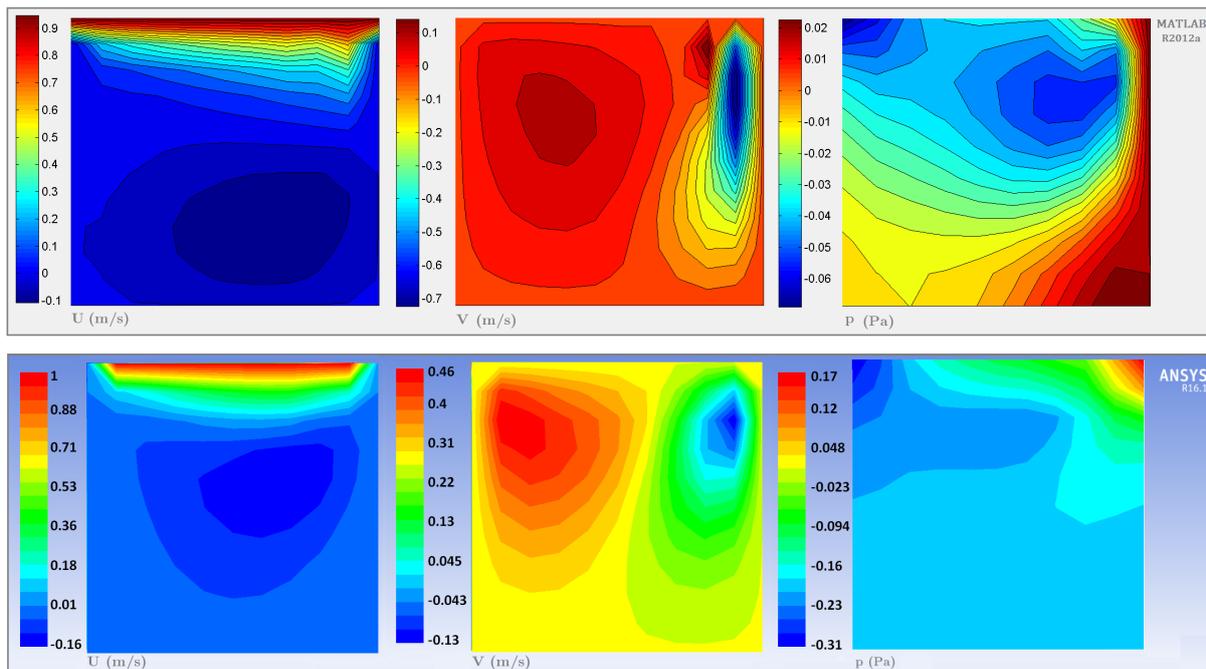
#### 4.2.1. $Re = 100$

##### Influencia de la malla

En un primer lugar se ha realizado un estudio del número de celdas en la malla para determinar qué malla se utilizará de ahora en adelante, utilizando un  $Re = 100$ . Cuando se aumenta el número de celdas la precisión de la simulación debería ser mayor, del mismo modo que el tiempo computacional, por lo que debemos llegar a un compromiso entre ambos dos.

Se han utilizado unas mallas de 10, 20, 30, 40 y 50 nodos en cada dirección. Los resultados obtenidos se comparan a continuación con los obtenidos con ANSYS.

##### ■ 10x10



**Figura 4.2** – Comparación de resultados de cada componente de la velocidad (U y V) y la presión (p) para la malla de 10x10.

En la figura anterior podemos comparar de forma visual los resultados obtenidos con los dos software. Sin embargo, hay veces que es más efectiva una comparación por gráficas, donde podamos comparar de forma numérica las dos soluciones. Por ello, a continuación compararemos los perfiles de velocidades para cada componente de la velocidad. Para la componente U se representarán el perfil de velocidad a lo largo de la recta vertical que

atraviesa el dominio, es decir, a través de la recta  $x = 0,5$ ; y para la componente V se utilizará la recta que corta horizontalmente el dominio por la mitad, es decir, la recta  $y = 0,5$ .

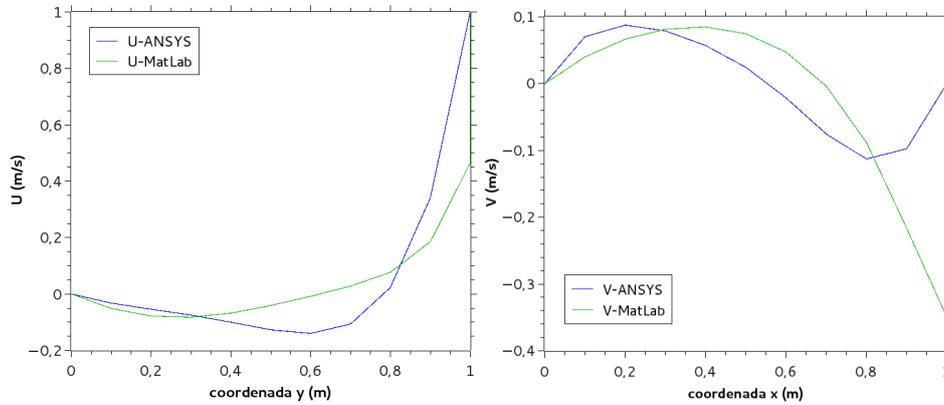


Figura 4.3 – Comparación de los perfiles de velocidad para una malla de 10x10.

■ 20x20

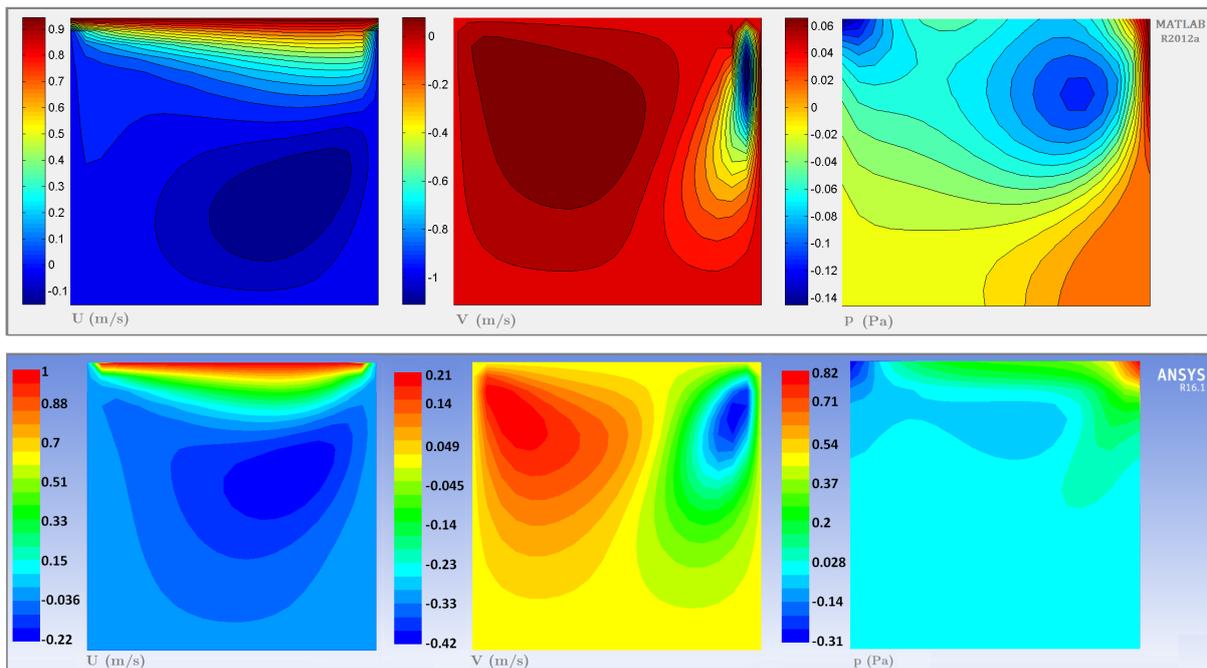


Figura 4.4 – Comparación de resultados de cada componente de la velocidad (U y V) y la presión (p) para la malla de 20x20.

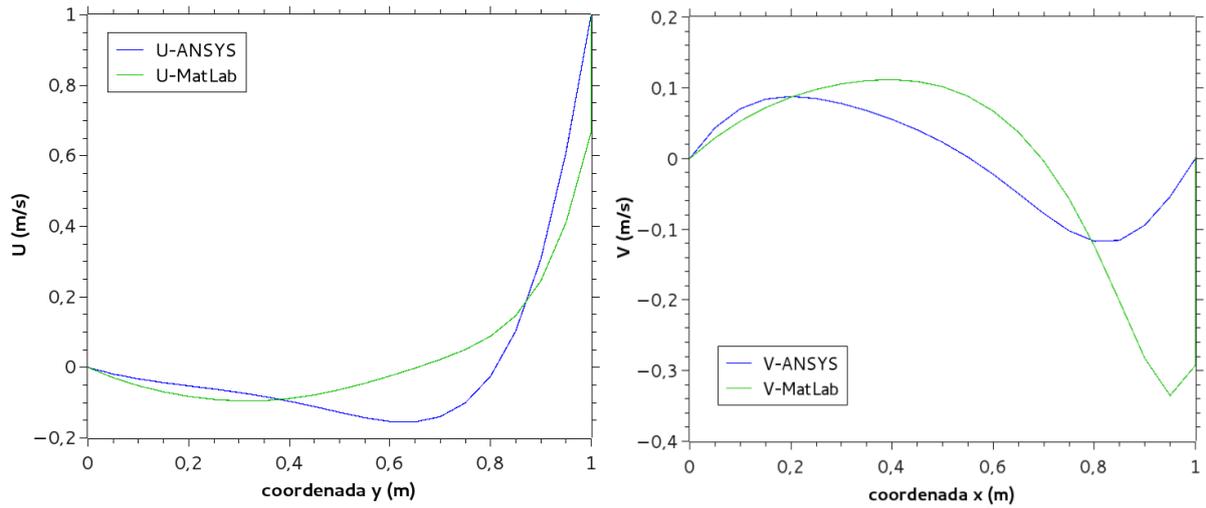


Figura 4.5 – Comparación de los perfiles de velocidad para una malla de 20x20.

■ 30x30

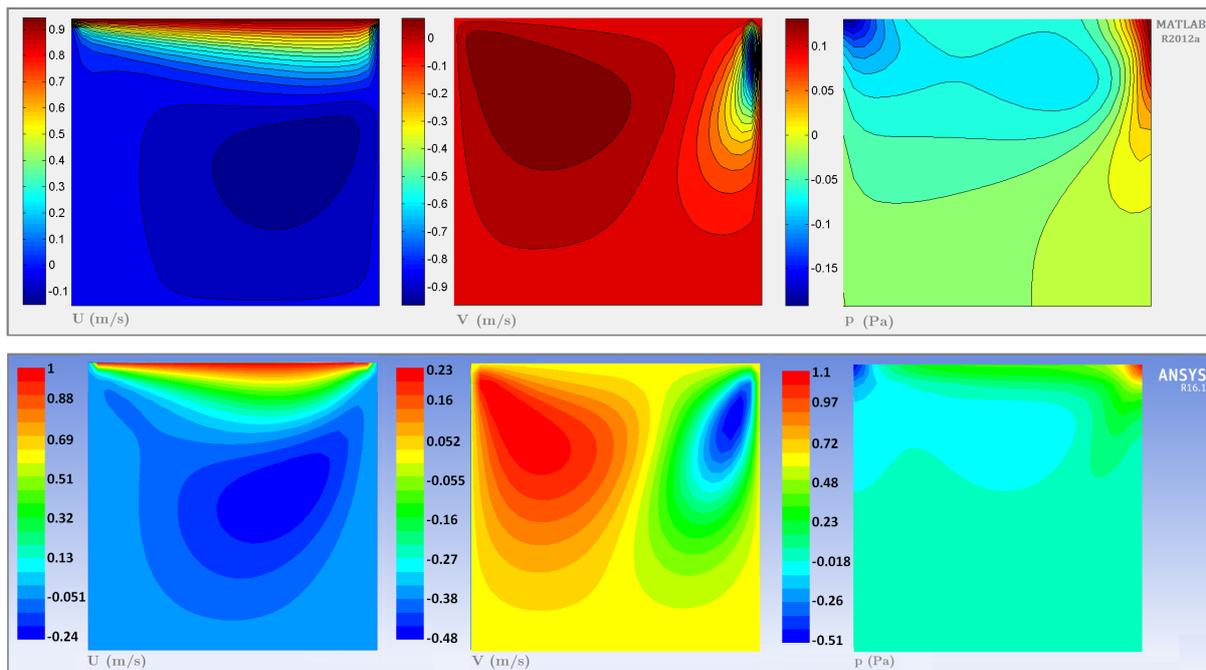


Figura 4.6 – Comparación de resultados de cada componente de la velocidad ( $U$  y  $V$ ) y la presión ( $p$ ) para la malla de 30x30.

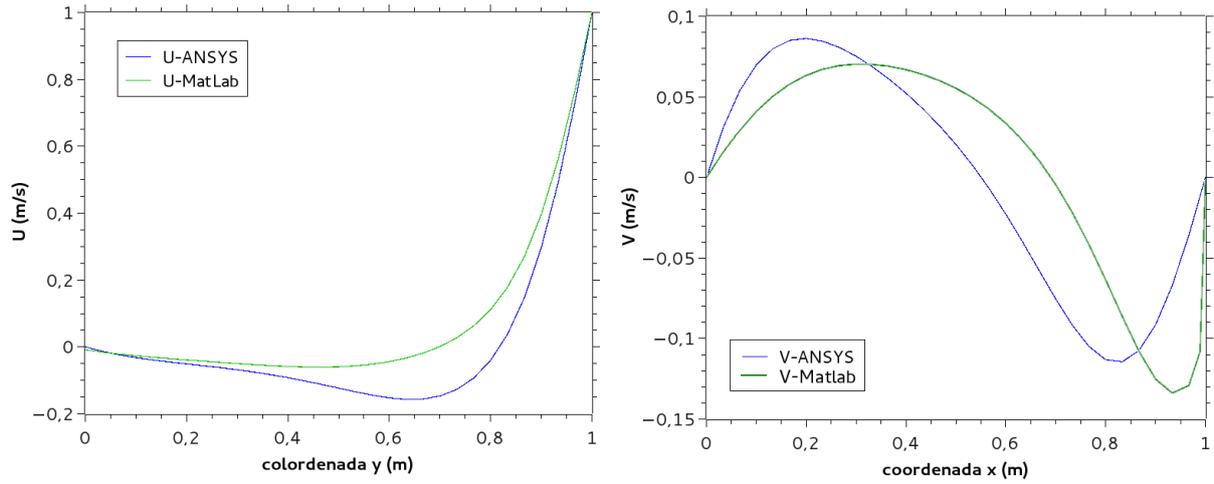


Figura 4.7 – Comparación de los perfiles de velocidad para una malla de 30x30.

■ 40x40

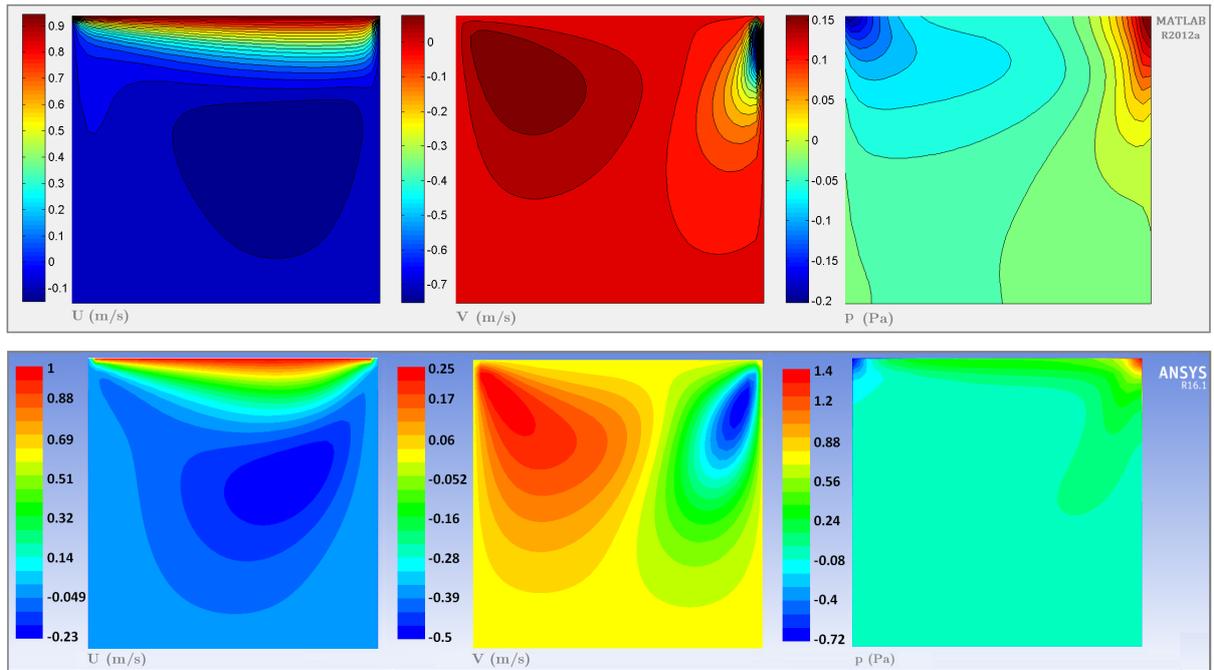


Figura 4.8 – Comparación de resultados de cada componente de la velocidad ( $U$  y  $V$ ) y la presión ( $p$ ) para la malla de 40x40.

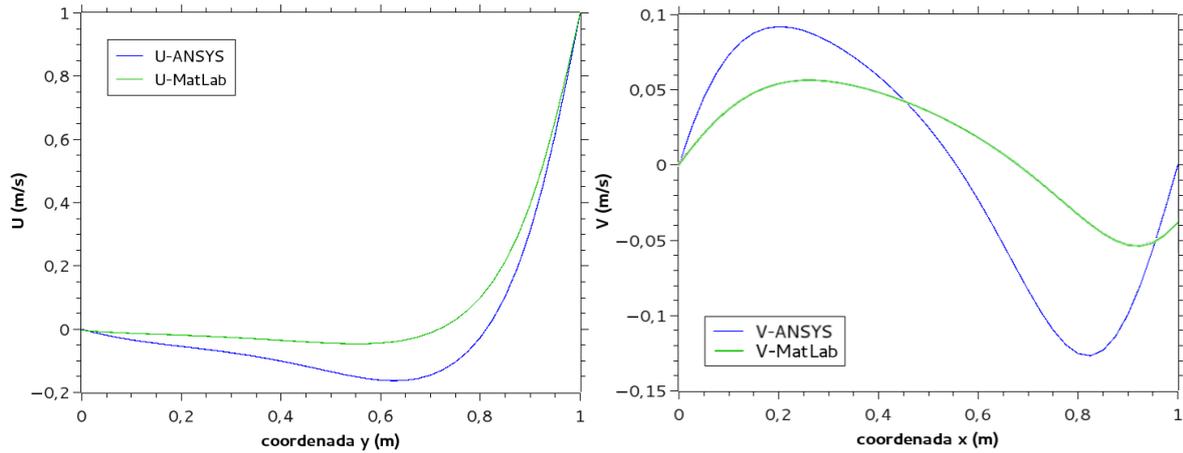


Figura 4.9 – Comparación de los perfiles de velocidad para una malla de 40x40.

■ 50x50

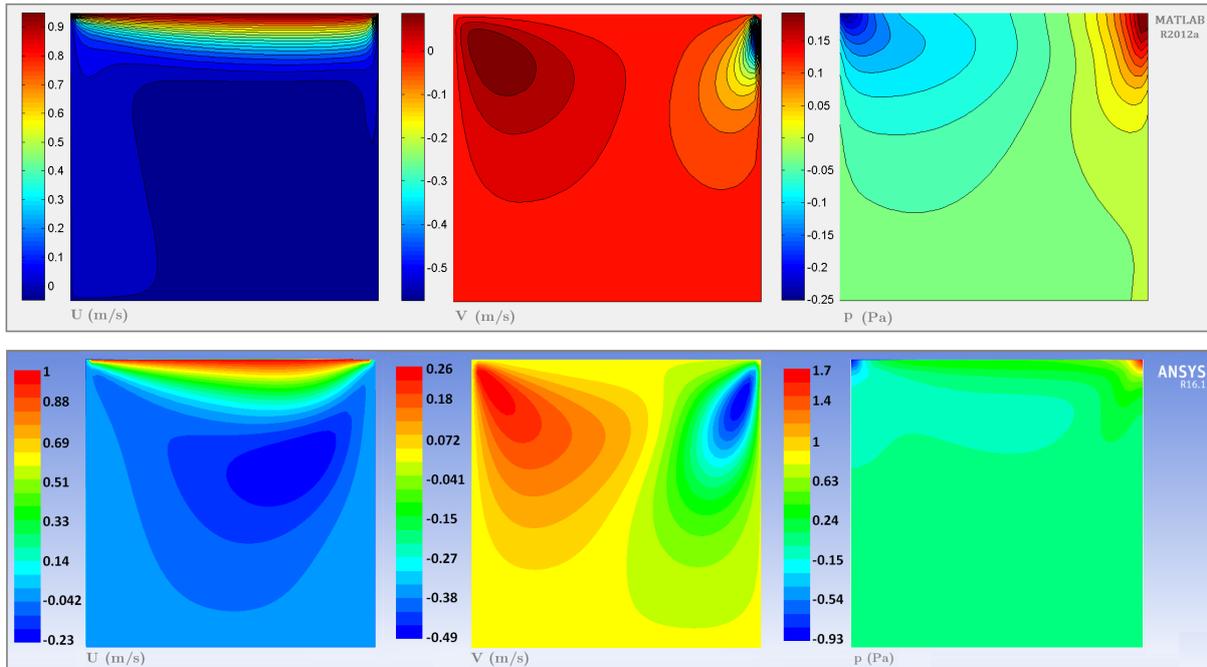


Figura 4.10 – Comparación de resultados de cada componente de la velocidad (U y V) y la presión (p) para la malla de 50x50.

A continuación hemos probado a reducir el criterio de convergencia a  $1 \cdot 10^{-6}$  en el error de la velocidad para la malla de 40, ya que vemos que los perfiles de velocidad se ajustan peor que en el caso de 30 celdas. Parece lógico que si aumentamos el número de celdas (y bajamos el paso de tiempo), el caso simule de forma más lenta y el criterio que tengamos que establecer sea menor que en los casos anteriores, por eso hemos querido comprobar si ese criterio menor mejora de forma notable los resultados teniendo en cuenta el tiempo computacional añadido.

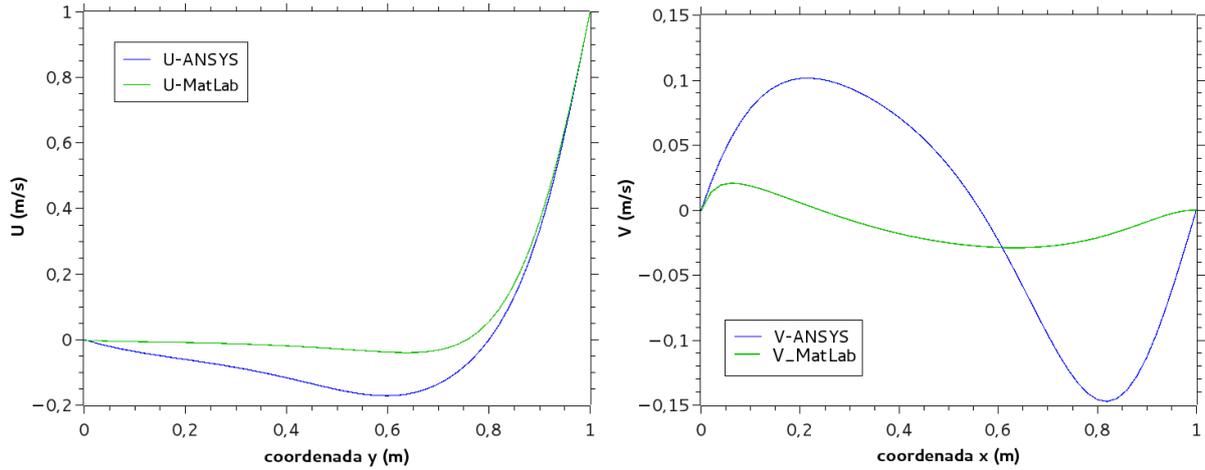


Figura 4.11 – Comparación de los perfiles de velocidad para una malla de 50x50.

Las nuevas gráficas comparativas de los perfiles de velocidad que se han obtenido son las siguientes:

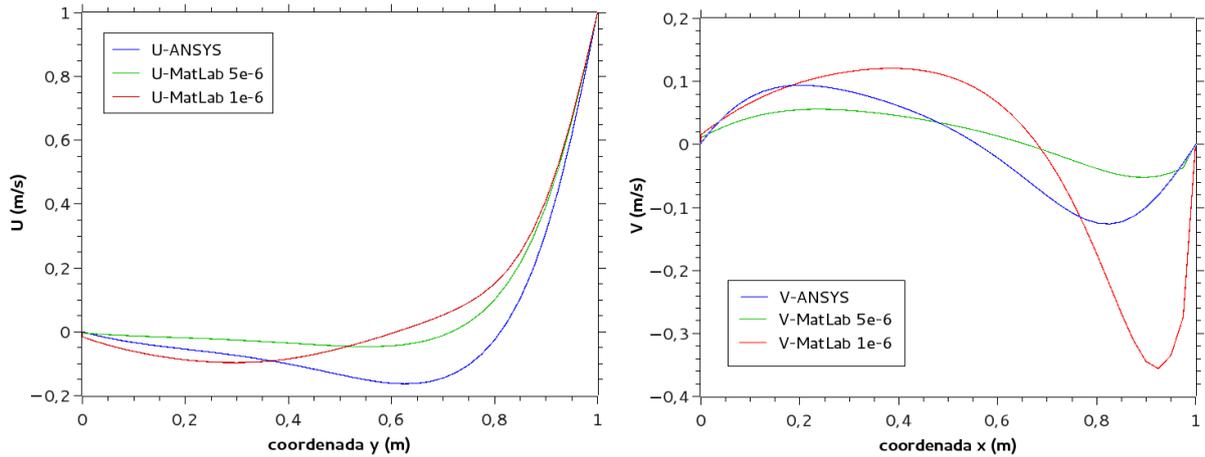


Figura 4.12 – Comparación de los perfiles de velocidad para una malla de 40x40 con un criterio de convergencia de  $5 \cdot 10^{-6}$  (línea verde) y  $1 \cdot 10^{-6}$  (línea roja).

error_ $\vec{v}$	nº pasos	t_comp
$5 \cdot 10^{-6}$	150526	1h 13min
$1 \cdot 10^{-6}$	581512	6h 9min

Cuadro 4.1 – Tabla comparativa del número de pasos y el tiempo computacional (t\_comp) requeridos con una malla de 40 celdas para los diferentes criterios de convergencia en la velocidad.

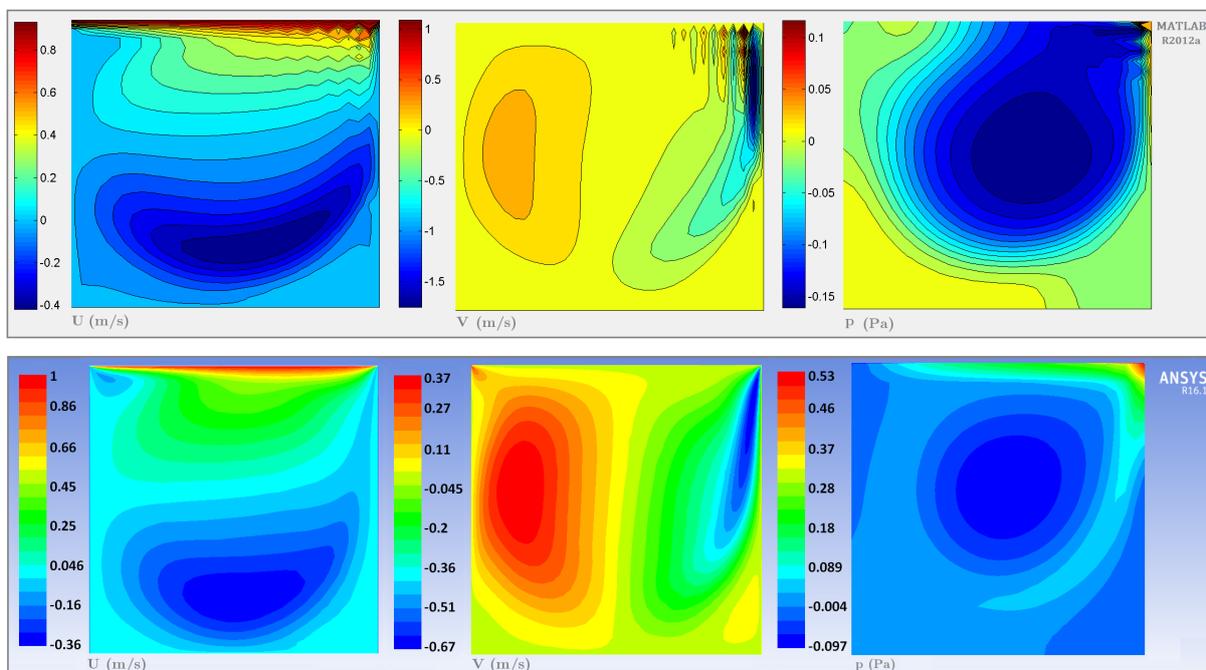
En la figura 4.12 no apreciamos de forma clara si los nuevos resultados se ajustan mejor

o peor a los anteriores, aunque sí podemos apreciar que no han mejorado con respecto a los obtenidos con la malla de 30 celdas (aún teniendo esta un error menor en la velocidad). Debido a esto y a la diferencia de tiempos computacionales que vemos en la tabla 4.1, la malla con la que se realizarán las posteriores simulaciones será la de 30x30.

Sin embargo, en cuanto a las simulaciones posteriores con números de Reynolds 1000 y 2000, el error en la velocidad que se va a utilizar sí es  $1 \cdot 10^{-6}$  (aunque eso suponga un aumento significativo del tiempo de computación), ya que se trata de dos únicas simulaciones y además son casos en los que nos vamos acercando a la zona de transición al régimen turbulento. Se trata de simulaciones más *delicadas* en las que intervienen velocidades más altas por tratarse de fluidos menos viscosos y hemos preferido optar por un criterio de convergencia menor para intentar obtener los mejores resultados posibles.

#### 4.2.2. $Re = 1000$

A continuación se presentan los resultados obtenidos para la simulación para un  $Re = 1000$ , con una malla de 30x30 y un error en la velocidad menor de  $1 \cdot 10^{-6}$ .



**Figura 4.13** – Comparación de resultados de cada componente de la velocidad (U y V) y la presión (p) para un  $Re = 1000$ .

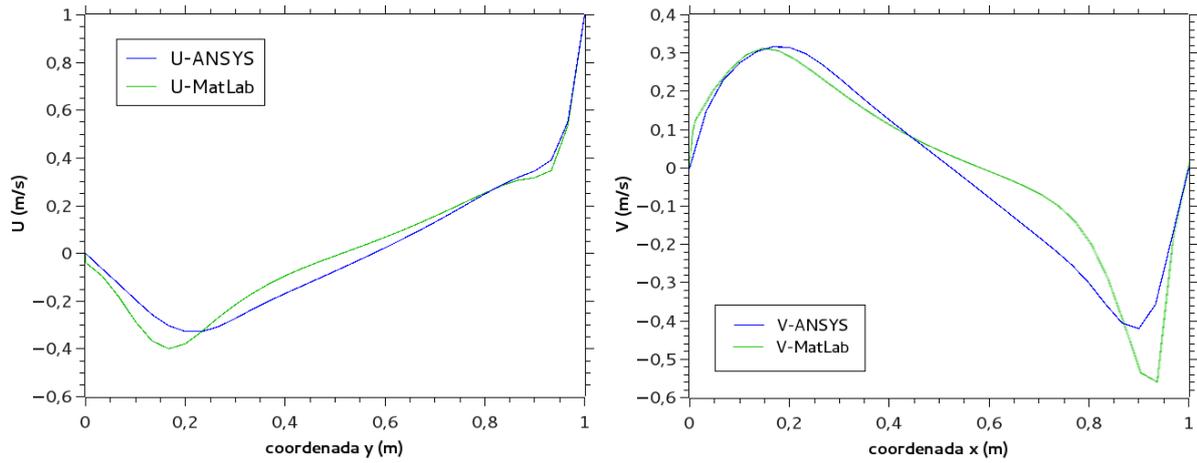


Figura 4.14 – Comparación de los perfiles de velocidad para un  $Re = 1000$ .

#### 4.2.3. $Re = 2000$

Para concluir se presentan los resultados obtenidos para la simulación para un  $Re = 2000$ , con una malla de  $30 \times 30$  y un error en la velocidad menor de  $1 \cdot 10^{-6}$ .

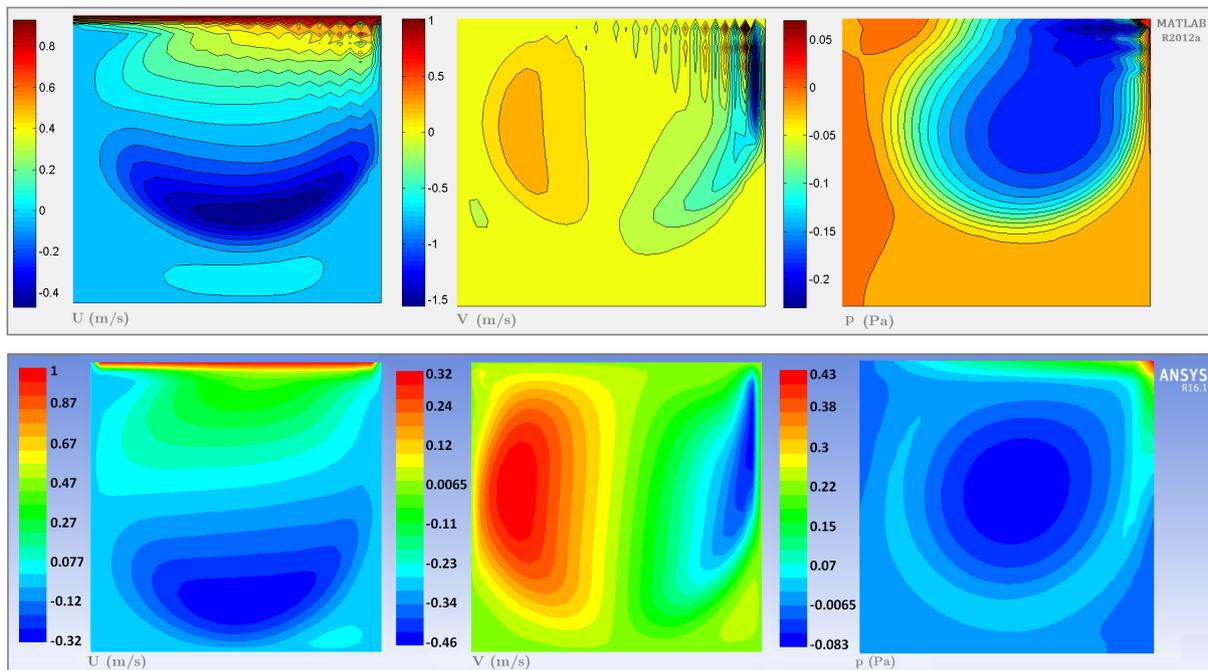


Figura 4.15 – Comparación de resultados de cada componente de la velocidad ( $U$  y  $V$ ) y la presión ( $p$ ) para un  $Re = 2000$ .

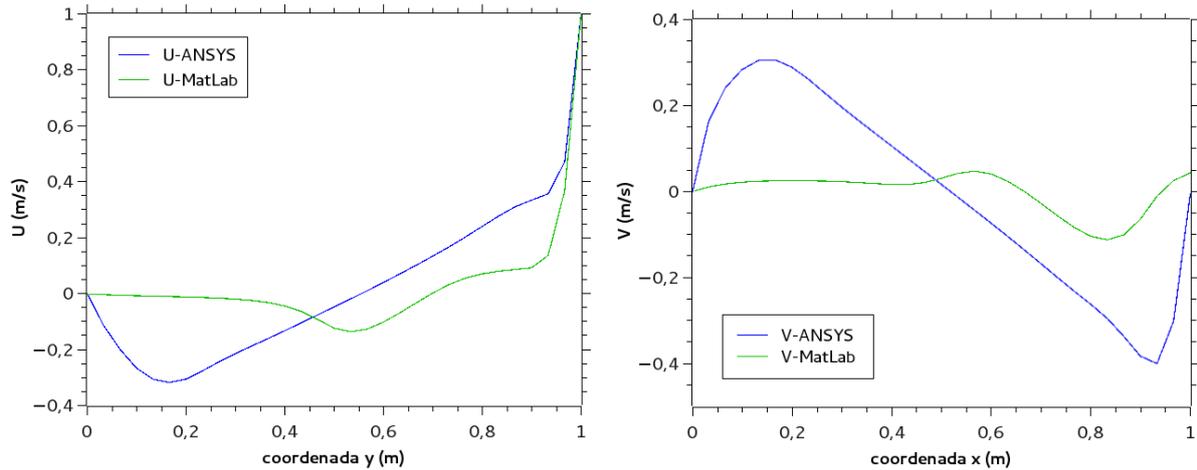


Figura 4.16 – Comparación de los perfiles de velocidad para un  $Re = 2000$ .

### 4.3. Discusión

A la vista de los resultados obtenidos diremos que de forma cualitativa el código resuelve correctamente la física del problema. Como era de esperar la pared móvil arrastra el fluido de la cavidad y se produce una recirculación del mismo en el sentido del movimiento de la pared. Al arrastrar ese fluido se forma en la pared superior un gradiente de presión, pasando de una presión negativa en la esquina izquierda que obliga al fluido a subir por la pared oeste y dirigirse a la zona de presión positiva en la esquina derecha, donde el fluido baja por la pared este.

#### Criterio de convergencia

Antes de comenzar la discusión de los resultados obtenidos hay que mencionar que las simulaciones con los dos programas han sido finalizadas bajo criterios de convergencia diferentes. Para el programa de MatLab hemos decidido que la simulación finalice cuando la diferencia relativa de la velocidad entre dos pasos de tiempo consecutivos sea menor que un cierto valor. ANSYS por su parte presenta unos criterios de convergencia más sofisticados y complicados de implementar. La idea básica de los criterios de ANSYS es calcular la diferencia de una magnitud a elegir con su valor en el instante anterior en todos los puntos del dominio, y después ponderar todas esas diferencias con la masa del fluido en cada punto. Lo ideal habría sido implementar en MatLab un criterio de convergencia igual al de ANSYS. Sin embargo se ha optado por el criterio mencionado debido a su sencillez de implementación, aún sabiendo que no se trata de un criterio ni muy efectivo ni muy utilizado en simulaciones CFD, aunque para cumplir con

los objetivos marcados del proyecto lo consideramos adecuado. Por tanto, de ahora en adelante tendremos en cuenta que una de las fuentes de error en las comparaciones de resultados es esta diferencia de criterios.

### **Influencia de la malla**

En el apartado de la influencia de la malla podemos apreciar que con las mallas menos densas (hasta 30 nodos) los resultados obtenidos cada vez se ajustan mejor a los obtenidos con el software de ANSYS. Sin embargo, con las mallas de 40 y 50 nodos, los resultados cada vez presentan mayores discrepancias en la velocidad, tal y como vemos en las gráficas 4.9 y 4.11. Además, en las figuras 4.8 y 4.10, sobretodo en la componente U de la velocidad, se observa que la recirculación en la parte inferior del dominio es menor de lo esperado, como si las simulaciones no hubiesen alcanzado el estado estacionario. Una de las causas más probables es precisamente la diferencia de criterios de convergencia en los dos programas. Es lógico pensar que para mallas menos densas en las que los pasos de tiempo permitidos son mayores y la simulación avanza más rápido, no se note demasiado la diferencia de criterios y el que hemos utilizado sea suficiente para alcanzar el estado estacionario, pero que, para mallas con un número de nodos mayor, sea necesario reducir el criterio de convergencia o incluso implementar un criterio diferente. Como ya hemos comentado, no se iba a implementar un criterio diferente y por ello el paso lógico era reducir el criterio de convergencia para comprobar si los resultados mejoraban. Con este criterio menor, como vemos en la figura 4.12, no se aprecian mejoras notables. Se podría haber seguido reduciendo el criterio de convergencia, pero a la vista del gran aumento en el tiempo computacional y de los resultados obtenidos con la malla de 30 nodos, se decidió utilizar ésta última para las posteriores simulaciones.

Para acabar este apartado, quisiéramos fijarnos en la siguiente tabla comparativa de tiempos computacionales (cuadro 4.2), donde vemos que la diferencia de tiempos entre los dos programas es muy grande. Uno de los motivos vuelve a ser el criterio de convergencia, ya que si dos simulaciones no terminan en el mismo *punto* de la simulación, es lógico que los tiempos computacionales sean diferentes. El segundo motivo de discrepancia es la diferencia de métodos de resolución, ya que ANSYS no utiliza el FSM para desacoplar las ecuaciones, y los métodos que tiene implementados llevan años optimizándose cada día para aumentar la eficiencia y eficacia del software, siendo hoy en día uno de los más utilizados en grandes empresas dedicadas a la simulación de fluidos. ANSYS dedica la mayor parte de sus esfuerzos para mejorar el tiempo de resolución, y es por esto precisamente por lo que existen unas diferencias tan grandes en el tiempo computacional. Por último, otro de los motivos es el lenguaje de programación en el que

celdas	$10 \times 10$	$20 \times 20$	$30 \times 30$	$40 \times 40$	$50 \times 50$
<b>t_comp - MatLab</b>	3min 1s	19min 55s	38min 35s	73min 6s	97min 5s
<b>t_comp - ANSYS</b>	1min 5s	1min 42s	5min 7s	6min 50s	11min

**Cuadro 4.2** – Tabla comparativa del tiempo computacional (*t\_comp*) requerido para las simulaciones con las diferentes mallas.

está escrito cada programa. MatLab se trata de un lenguaje *interpretado* en el que cada vez que ejecutamos el programa un intérprete traduce el código fuente instrucción a instrucción a un lenguaje entendible por el ordenador. Sin embargo ANSYS está escrito en C, un lenguaje *compilado* en el que un compilador traduce el código una sola vez creando un archivo ejecutable entendible para la máquina, reduciéndose así el tiempo de ejecución del programa.

### **Re = 100 / 1000 / 2000**

Por otro lado, atendiendo a la física que hay detrás del número de Reynolds, discutiremos a continuación los diferentes resultados en los apartados de  $Re = 100$ ,  $Re = 1000$  y  $Re = 2000$ .

El apartado de  $Re = 100$  es el caso con el número de Reynolds más bajo y por lo tanto el fluido más viscoso de los tres. Como podíamos prever de forma intuitiva, es el caso que presenta velocidades más bajas, ya que las fuerzas de rozamiento viscosas son más altas y al fluido le *cuesta* más desplazarse. En el caso de  $Re = 1000$  las velocidades son más altas que en el caso anterior y los perfiles de velocidad se ajustan realmente bien a los resultados de ANSYS. Sin embargo, en el caso de  $Re = 2000$ , aunque las velocidades también son más altas, la comparación con los perfiles de velocidad de ANSYS son bastante diferentes. El principal motivo puede ser que con ese número de Reynolds nos encontramos en el límite entre el régimen laminar y la zona de transición al régimen turbulento, por lo que los métodos numéricos de resolución implementados ya no son los idóneos. Sin embargo ANSYS sí que tiene implementados los métodos numéricos adecuados para resolver las ecuaciones en la zona de transición, por lo que no presenta diferencias significativas entre los casos con Reynolds más altos. Estas discrepancias en caso de  $Re = 2000$  no las consideramos un error del programa creado, al contrario, esto debía de ocurrir ya que nuestro código no está adaptado para resolver flujos en la zona de transición.

En cuanto a los casos con números de Reynolds 1000 y 2000, como vemos en las figuras 4.13 y 4.15, en una de las esquinas aparecen inestabilidades numéricas, hecho lógico si recordamos que se trata de un código sencillo en el que la eficiencia no era una prioridad. Hasta ahora no se

habían apreciado dichas inestabilidades probablemente porque el paso de tiempo que podemos permitirnos para cumplir el criterio de Courant en el caso de  $Re = 100$  es mayor que en los casos de  $Re = 1000$  y  $Re = 2000$ , por tanto el número de iteraciones necesarias para alcanzar el estado estacionario es menor y la acumulación de errores es menor también. Es por ello que se probó a reducir el paso de tiempo para estas dos simulaciones (resultados no mostrados), aunque se obtuvieron las mismas inestabilidades. Por otro lado también es posible que la malla que hemos elegido como adecuada para el caso de  $Re = 100$  no lo fuera para estos casos con Reynolds altos, pero como ya hemos visto, con una malla mayor se producía una atenuación de la recirculación en la parte inferior del dominio y por ese motivo no se ha probado a utilizar una malla con más nodos.

#### 4.4. Conclusiones

Para finalizar el proyecto comentaremos las conclusiones generales que podemos extraer de la realización de este proyecto.

Por un lado hemos creado, desde cero, un código sencillo con MatLab que resuelve las ecuaciones incompresibles de Navier-Stokes para el régimen laminar. Se ha validado la resolución de la física utilizando el caso Driven cavity y comparando de forma detallada los resultados obtenidos con nuestro programa con aquellos obtenidos con ANSYS Fluent.

Hemos comprobado que el código responde cualitativamente a la física que se espera del fluido. Dentro de la cavidad con la tapa deslizante se observa la recirculación del fluido, movido por el rozamiento con la tapa. Lo que es más, la comparación cuantitativa con programas comerciales es muy razonable, observándose como en fluidos viscosos la velocidad del flujo es pequeña y el comportamiento numérico de nuestro código robusto; y según nos acercamos al régimen turbulento se van encontrando mayores divergencias causadas por la sencillez numérica de nuestra implementación. Estos resultados pueden considerarse positivos teniendo en cuenta la enorme complejidad de las modernas técnicas computacionales de dinámica de fluidos y por lo tanto concluiremos diciendo que los objetivos marcados en un principio del proyecto han sido cumplidos con éxito.

# Apéndice

# Apéndice A

## Código MatLab

```

1 function [U,V,p_final,errorU,errorV] = navier_stokes(n_step_snst, nx, ny, dt)
2 format long
3 tic
4
5 %% Physical variables
6 rho = 933; % density
7 mu = 0.2390; % dinamic viscosity
8 D_equivalent = (Lx*Ly)*4/(Lx+Lx+Ly+Ly); % equivalent diameter for the reynolds number
9 Re = rho*D_equivalent*v_in/mu; % Reynolds number
10 Re = 100; % The reynolds number has to be less than 2000 for the laminar regime
11 v0=1; % modulus of initial velocity (m/s)
12
13 %% Grid variables
14 Lx = 1; % Lx is the domain length in x direction
15 Ly = 1; % Ly is the domain length in y direction
16 dx = Lx/nx; % dx is the cell size in x direction
17 dy = Ly/ny; % dy is the cell size in y direction
18
19 %% Plotting variables
20 makevideo=1;
21 if (makevideo==1)
22     h=figure('Position',[1 0 800 600],'MenuBar','none','ToolBar','none','resize','off','Visible','off');
23     vidobj = VideoWriter('fluid.avi');
24     open(vidobj);
25     axis tight;
26     set(gca, 'CLim',[0,1],'position',[0 0 1 1],'Visible','off');
27 end
28
29 %% Initial matrices
30 U=zeros(ny+2,nx+2);
31 V=zeros(ny+2,nx+2);
32 p=zeros(ny+2,nx+2);
33 A=coefficient_pressure_matrix(nx,ny,dx,dy); % the coefficient matrix to solve the pressure
34
35
36 function [A,aW,aE,aN,aS] = coefficient_pressure_matrix(nx,ny,dx,dy)
37 %% Create the coefficient matrix (A) to solve the pressure
38 %% Create the empty matrix
39 A=spalloc(ny*nx,ny*nx,5*(ny*nx-2*nx));
40 %% Create diagonals
41 aW=spalloc(ny,nx,ny*nx-2*nx);
42 aE=spalloc(ny,nx,ny*nx-2*nx);
43 aN=spalloc(ny,nx,ny*nx-2*nx);
44 aS=spalloc(ny,nx,ny*nx-2*nx);
45 for i=1:ny
46     for j=1:nx
47         aW(i,j)=dy/dx;
48         aE(i,j)=dy/dx;
49         aN(i,j)=dx/dy;
50         aS(i,j)=dx/dy;
51     end
52 end
53
54 aN(end,:)=0;
55 aS(1,:)=0;
56 aP=aW+aE+aN+aS;
57 aW=aW(:);
58 aE=aE(:);
59 aN=aN(:);
60 aS=aS(:);
61 aP=aP(:);
62
63 A=-aP,0+diag(aN(1:end-1),1)+diag(aS(2:end),-1)+diag(aE(1:end-ny),ny)+diag(aW(ny+1:end),-ny);
64
65 end
66
67
68 %% Boundary conditions
69 U(:,1)=0; V(:,1)=0; % east
70 U(:,end)=0; V(:,end)=0; % west
71 U(1,:)=0; V(1,:)=0; % south
72 U(end,:)=1; V(end,:)=0; % north
73
74

```

```

75 %% Time integration variables
76 tolerance_p=0.001;
77 n_steps_p=3000;
78 n_step=1; %n_step = 1 ---> in the first iteration the scheme for time discretization
79 % is explicit Euler (velocity_star = v_previous+dt*R(velocity))
80 % R(velocity) is the discretized vector of convection and
81 % diffusion parts
82 % If n_step > 1 ---> the time discretization scheme is Adam-Bashford
83 % (velocity_star = v_previous+dt*[1.5*R(velocity) - 0.5*R(velocity)])
84 errorU=1;
85 errorV=1;
86
87 %% Beginning the solving part of the code
88 while (errorV > 5e-6)
89
90 %% Calculate the mid-way velocity
91 if n_step == 1 %% Explicit Euler
92 U=avg(U);
93 V=avg(V)';
94 [U_star, V_star, Ru_old, Rv_old] = midway_velocity_Euler(U, V, Re, dx, dy, nx, ny, dt);
95
96
97 function [U_star, V_star, Ru, Rv] = midway_velocity_Euler(U, V, Re, dx, dy, nx, ny, dt)
98
99 Ru=vector_Ru(U,V,Re,dx,dy,nx,ny);
100 Rv=vector_Rv(U,V,Re,dx,dy,nx,ny);
101
102 U_star=U+dt*Ru;
103 V_star=V+dt*Rv;
104
105 end
106
107 function [Ru] = vector_Ru(U, V, Re, dx, dy, nx, ny)
108 % R(u)=aWu*UW+aEu*UE+aNu*UN+aSu*US-aPu*UP
109 % The matrices of coefficients have the same dimension as U
110 aWu=zeros(ny+2,nx+1);
111 aEu=zeros(ny+2,nx+1);
112 aNu=zeros(ny+2,nx+1);
113 aSu=zeros(ny+2,nx+1);
114 Ru=zeros(ny+2,nx+1);
115
116 for i=2:ny+1
117 for j=2:nx
118 aWu(i,j)=(dy/Re/dx)+0.25*dy.*(U(i,j)+U(i,j-1)); % UP+UW
119 aEu(i,j)=(dy/Re/dx)-0.25*dy.*(U(i,j)+U(i,j+1)); % UP+UE
120 aNu(i,j)=(dx/Re/dy)-0.25*dx.*(V(i,j+1)+V(i,j)); % VP+VE
121 aSu(i,j)=(dx/Re/dy)+0.25*dx.*(V(i-1,j)+V(i-1,j+1)); % VS+VES
122
123 end
124 end
125 aPu=aWu+aEu+aNu+aSu;
126
127 % Multiply each coefficient by its corresponding U point (UW, UE...etc)
128 for i=2:ny+1
129 for j=2:nx
130 Ru(i,j)=aWu(i,j).*U(i,j-1)+aEu(i,j).*U(i,j+1)+aNu(i,j).*U(i+1,j)+aSu(i,j).*U(i-1,j)-aPu(i,j).*U(i,j);
131
132 end
133 end
134
135 function [Rv] = vector_Rv(U, V, Re, dx, dy, nx, ny)
136 aWv=zeros(ny+1,nx+2);
137 aEv=zeros(ny+1,nx+2);
138 aNv=zeros(ny+1,nx+2);
139 aSv=zeros(ny+1,nx+2);
140 Rv=zeros(ny+1,nx+2);
141 for i=2:ny
142 for j=2:nx
143 aNv(i,j)=(dx/Re/dy)-0.25*dx.*(V(i,j)+V(i+1,j)); % VP+VN
144 aSv(i,j)=(dx/Re/dy)+0.25*dx.*(V(i,j)+V(i-1,j)); % VP+VS
145 aEv(i,j)=(dy/Re/dx)-0.25*dy.*(U(i,j)+U(i+1,j)); % UN+UP
146 aWv(i,j)=(dy/Re/dx)+0.25*dy.*(U(i+1,j-1)+U(i,j-1)); % UWN+UW
147
148 end
149 end
150 aPv=aWv+aEv+aNv+aSv;
151 for i=2:ny
152 for j=2:nx
153 Rv(i,j)=aWv(i,j).*V(i,j-1)+aEv(i,j).*V(i,j+1)+aNv(i,j).*V(i+1,j)+aSv(i,j).*V(i-1,j)-aPv(i,j).*V(i,j);
154
155 end
156 end

```

```

155 [grad_p_U,grad_p_V,p]=pressure_correction(U_star,V_star,A,nx,ny,dx,dy,dt,tolerance_p,n_steps_p,p);
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228

```

```

function [grad_p_U,grad_p_V,p]=pressure_correction(U_star,V_star,A,nx,ny,dx,dy,dt,tolerance_p,n_steps_p,p)
%% Calculate the pressure by solving the Poisson equation and
|
|   return the gradient of the calculated pressure
%% Calculate the divergence of the velocity_star
div_velocity_star=diver(U_star,V_star,nx, ny, dx, dy);
p_prima=p(2:end-1,2:end-1);
%% Calculate the pressure
p_prima=bicgstab(A,div_velocity_star(:)/dt,tolerance_p,n_steps_p);
p_prima=reshape(p_prima,ny,nx);
p(2:end-1,2:end-1)=p_prima;
%% Boundary conditions
p(:,1) = p(:,2);
p(:,end) = p(:,end-1);
p(1,:) = p(2,:);
p(end,:) = p(end-1,:);

grad_p_U=0.5*(dx*(p(1:end,2:end)-p(1:end,1:end-1)));
grad_p_V=0.5*dy*(p(2:end,1:end)-p(1:end-1,1:end));

end

```

```

grad_p_U=grad_p_U.*dt;
grad_p_V=grad_p_V.*dt;

%% Calculate the velocity at step n+1 by correcting it with the
%% previous divergence of p
U = U_star - grad_p_U;
V = V_star - grad_p_V;

%% Boundary conditions
U(:,1)=0;   V(:,1)=0;   % east
U(:,end)=0; V(:,end)=0; % west
U(1,:)=0;   V(1,:)=0;   % south
U(end,:)=1; V(end,:)=0; % north

n_step=n_step+1;
else %% Adam-Bashford
[U_star, V_star, Ru_old, Rv_old] = midway_velocity_AdamB(U,V,Re,dx,dy,nx,ny,dt,Ru_old,Rv_old);

```

```

function [U_star, V_star, Ru, Rv] = midway_velocity_AdamB(U,V,Re,dx,dy,nx,ny,dt,Ru_old,Rv_old)
Ru=vector_Ru(U,V,Re,dx,dy,nx,ny);
Rv=vector_Rv(U,V,Re,dx,dy,nx,ny);

U_star=U+dt*(1.5*Ru-0.5*Ru_old);
V_star=V+dt*(1.5*Rv-0.5*Rv_old);

end

```

```

%% Calculate the gradient of the pressure to correct the mid-way
%% velocity
[grad_p_U,grad_p_V, p]=pressure_correction(U_star,V_star,A,nx,ny,dx,dy,dt,tolerance_p,n_steps_p,p);
grad_p_U=grad_p_U.*dt;
grad_p_V=grad_p_V.*dt;
%% Calculate the velocity at step n+1 by correcting it with the
%% previous divergence of p
U = U_star - grad_p_U;
V = V_star - grad_p_V;

%% Boundary conditions
U(:,1)=0;   V(:,1)=0;   % east
U(:,end)=0; V(:,end)=0; % west
U(1,:)=0;   V(1,:)=0;   % south
U(end,:)=1; V(end,:)=0; % north

if (mod(n_step,n_step_snst)==0)
    if makevideo==1
        contourf(p(2:end-1,2:end-1),20)
        currframe=getframe;
        writeVideo(vidobj,currframe);
    end
end

```

```

229 %% Plot and store figures and matrices
230 filename = ['presion_' num2str(n_step/n_step_snst)];
231 h=dialog ( 'visible', 'off', 'windowstyle', 'normal');
232 ax=axes('parent', h, 'nextplot', 'add' );
233 colormap(h,'Jet')
234 contourf(ax,p(2:end-1,2:end-1),20);
235 hh=colorbar('peer', ax);
236 title('Contorno de presión');
237 saveas ( ax, filename, 'png' )
238 close(h)
239 filename = ['velocidadU_' num2str(n_step/n_step_snst)];
240 v=dialog ( 'visible', 'off', 'windowstyle', 'normal');
241 ax=axes('parent', v, 'nextplot', 'add' );
242 colormap(v,'Jet')
243 contourf(ax,U,20);
244 hh=colorbar('peer', ax);
245 title('Campo de velocidad');
246 saveas ( ax,filename, 'png' )
247 close(v)
248 filename = ['velocidadV_' num2str(n_step/n_step_snst)];
249 v=dialog ( 'visible', 'off', 'windowstyle', 'normal');
250 ax=axes('parent', v, 'nextplot', 'add' );
251 colormap(v,'Jet')
252 contourf(ax,V,20);
253 hh=colorbar('peer', ax);
254 title('Campo de velocidad');
255 saveas ( ax,filename, 'png' )
256 close(v)
257 end
258
259 n_step=n_step+1
260 errorU=norm(U-U0)/norm(U)
261 errorV=norm(V-V0)/norm(V)
262 p_final=p;
263 end
264 end
265
266 time=toc;
267 print_matrix(U,V,p_final,errorU,errorV,n_step,toc);
268
269 function [] = print_matrix(U,V,p,errorU, errorV,n_step)
270 [rowsU colsU]=size(U);
271 fileID= fopen('U.txt','wt');
272 for i=1:rowsU
273     for j=1:colsU
274         fprintf(fileID,'%f\t', U(i,j));
275     end
276     fprintf(fileID, '\n');
277 end
278 fclose(fileID);
279 [rowsV colsV]=size(V);
280 fileID= fopen('V.txt','wt');
281 for i=1:rowsV
282     for j=1:colsV
283         fprintf(fileID,'%f\t', V(i,j));
284     end
285     fprintf(fileID, '\n');
286 end
287 fclose(fileID);
288 [rowsp colsp]=size(p);
289 fileID= fopen('p.txt','wt');
290 for i=1:rowsp
291     for j=1:colsp
292         fprintf(fileID,'%f\t', p(i,j));
293     end
294     fprintf(fileID, '\n');
295 end
296 fclose(fileID);
297 fileID= fopen('errors.txt','wt');
298 fprintf(fileID,'%f\t', errorU);fprintf(fileID, '\n');
299 fprintf(fileID,'%f\t', errorV);fprintf(fileID, '\n');
300 fprintf(fileID,'%f\t', n_step);fprintf(fileID, '\n');
301 fclose(fileID);
302 end
303
304 if (makevideo==1)
305     close(vidobj);
306 end
307 end

```

## Apéndice B

# Formulación matemática del teorema de descomposición de Helmholtz-Hodge aplicado a las ecuaciones de Navier-Stokes

A continuación se aplica el teorema de Helmholtz-Hodge (de forma más rigurosa) a las ecuaciones simplificadas de Navier-Stokes:

$$\frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \vec{\nabla}) \vec{v} = \frac{1}{Re} \nabla^2 \vec{v} - \vec{\nabla} p \quad (\text{B.1})$$

$$\vec{\nabla} \cdot \vec{v} = 0 \quad (\text{B.2})$$

El teorema de Helmholtz-Hodge empieza proyectando<sup>14</sup> (B.1) sobre el campo de divergencia nula de manera que el término del gradiente de presión desaparezca.

$$P \left[ \frac{\partial \vec{v}}{\partial t} + (\vec{v} \cdot \vec{\nabla}) \vec{v} \right] = P \left[ \frac{1}{Re} \nabla^2 \vec{v} - \vec{\nabla} p \right] \quad (\text{B.3})$$

El término temporal del primer paréntesis permanece invariante puesto que el campo de velocidades es solenoidal y ya se encuentra en el plano de divergencia nula. Por otro lado, el término de la presión desaparece ya que se encuentra en el plano de gradientes, ortogonal a  $\vec{v}$ , y su proyección es nula, obteniendo:

---

<sup>14</sup>Denotaremos el operador proyección por  $P$ .

$$\frac{\partial \vec{v}}{\partial t} = P \left[ -(\vec{v} \cdot \vec{\nabla}) \vec{v} + \frac{1}{Re} \nabla^2 \vec{v} \right] \quad (\text{B.4})$$

El siguiente paso es despejar la presión de la ecuación de Navier-Stokes y sustituir el termino temporal por lo encontrado en (B.4).

$$\vec{\nabla} p = \underbrace{\left[ -(\vec{v} \cdot \vec{\nabla}) \vec{v} + \frac{1}{Re} \nabla^2 \vec{v} \right]}_{R(\vec{v})} - P \underbrace{\left[ -(\vec{v} \cdot \vec{\nabla}) \vec{v} + \frac{1}{Re} \nabla^2 \vec{v} \right]}_{\vec{v}^{n+1}} \quad (\text{B.5})$$

De esta forma obtenemos descompuesta la ecuación de Navier-Stokes como mencionábamos al principio de la sección. Por un lado el primer paréntesis de (B.6) representa el campo irrotacional,  $R(\vec{v})$ , y los términos del segundo paréntesis se corresponden con el campo solenoidal,  $\vec{v}$ .

$$R(\vec{v}) \equiv -(\vec{v} \cdot \vec{\nabla}) \vec{v} + \frac{1}{Re} \nabla^2 \vec{v} \quad (\text{B.6})$$

$$\vec{\nabla} p = R(\vec{v}) - \vec{v}^{n+1} \quad (\text{B.7})$$

Por último, nos falta una ecuación para cerrar el sistema de ecuaciones y poder calcular la presión, por lo que aplicamos el operador divergencia a la ecuación (B.6) y generamos la mencionada ecuación de Poisson:

$$\nabla^2 p = \vec{\nabla} \cdot \left[ -(\vec{v} \cdot \vec{\nabla}) \vec{v} + \frac{1}{Re} \nabla^2 \vec{v} \right] \quad (\text{B.8})$$

# Bibliografía

- [1] Butcher, J. C. (2000). Numerical methods for ordinary differential equations in the 20th century. *Journal of Computational and Applied Mathematics*, 125(1), 1-29.
- [2] Armfield, S. and Street, R. (1999). The fractional-step method for the Navier–Stokes equations on staggered grids: the accuracy of three variations. *Journal of Computational Physics*, 153(2), 660-665.
- [3] Botella, O. and Peyret, R. (1998). BENCHMARK SPECTRAL RESULTS ON THE LID-DRIVEN CAVITY FLOW. *Computers & Fluids*. 27(4), 421-433.
- [4] Chorin, A. J. (1997). A numerical method for solving incompressible viscous flow problems. *Journal of Computational Physics*, 135(2), 118-125.
- [5] Chorin, A. J. (1968). Numerical solution of the Navier-Stokes equations. *Mathematics of Computation*, 22(104), 745-762.
- [6] Chorin, A. J. and Marsden, J. E. (1990). A Mathematical Introduction to Fluid Mechanics. Department of Mathematics, University of California.
- [7] Costarelli, S. D., Paz, R., Dalcin, L. and Storti, M. (2011). Resolución de las ecuaciones de Navier-Stokes utilizando CUDA. *Mecánica Computacional*, 30, 2979-3008.
- [8] Courant, R., Friedrichs, K. and Lewy, H. (1967). On the Partial Difference Equations of Mathematical Physics. *IBM Journal*.
- [9] Crespo, A. (2002). Mecánica de fluidos. Universidad Politécnica de Madrid. Escuela Técnica Superior de Ingenieros Industriales. Servicio de Publicaciones.
- [10] Ferziger, J. H. and Perić, M. (2002). Computational methods for fluid dynamics (Vol. 3, pp. 196-200). Berlin: Springer.
- [11] Fueyo, N. (2010). Fundamentos de Fluidos y Procesos Fluidomecánicos. *Área de Mecánica de Fluidos, Universidad de Zaragoza*.

- [12] Guia, U., Ghia, K. N. and Shin, C. T. (1982). High-Re Solutions for Incompressible Flow using the Navier-Stokes Equations and a Multigrid Method. *Journal of Computational Physics*. 48, 387-411.
- [13] He, Q. and Wang, X-P. (2009). Numerical study of the effect of Navier slip on the driven cavity flow. *Journal of Applied Mathematics and Mechanics*, 89(10), 857-868.
- [14] Kim, J. and Moin, P. (1985). Application of a fractional-step method to incompressible Navier-Stokes equations. *Journal of computational physics*, 59(2), 308-323.
- [15] Landau, L. D. (1969). *Mecánica de Fluidos*, Ed.
- [16] Mellado, K. L. C., Ibarra, J. E. J. and Fonseca, F. R. (2013). Solución numérica de las ecuaciones de Navier-Stokes incompresibles por el método de los volúmenes finitos. Numerical solution of the incompressible Navier-Stokes equations with finite volume method.
- [17] Ochoa, J. S. (2010). Modelización de la combustión de llamas turbulentas mediante la Simulación de las Grandes Escalas (Tesis doctoral). Universidad de Zaragoza.
- [18] Sayma, A. (2009). *Computational fluid dynamics*. Bookboon.
- [19] Seibold, B. (2008). A compact and fast Matlab code solving the incompressible Navier-Stokes equations on rectangular domains. *Massachusetts Institute of Technology*.
- [20] Sohankar, A., Norberg, C. and Davidson, L. (1998). Low-Reynolds-number flow around a square cylinder at incidence: study of blockage, onset of vortex shedding and outlet boundary condition. *International Journal for numerical methods in fluids*, 26(1), 39-56.
- [21] Tu, J., Yeoh, G. H. and Liu, C. (2007). *Computational fluid dynamics: a practical approach*. Butterworth-Heinemann.
- [22] White, F. M. and Corfield, I. (2006). *Viscous fluid flow* (Vol. 3). New York: McGraw-Hill.