

Trabajo Fin de Máster Máster en Ingeniería Aeronáutica

Exposición y revisión del método SIMPLE con aplicaciones a flujos de geometría sencilla

Autor: Aitor Robles Corpa

Tutor: Miguel Pérez-Saborid Sánchez-Pastor

**Dpto. Ingeniería Aeroespacial y Mecánica de Fluidos
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla**

Sevilla, 2024



Trabajo Fin de Máster
Máster en Ingeniería Aeronáutica

Exposición y revisión del método SIMPLE con aplicaciones a flujos de geometría sencilla

Autor:

Aitor Robles Corpa

Tutor:

Miguel Pérez-Saborid Sánchez-Pastor

Profesor Titular

Dpto. Ingeniería Aeroespacial y Mecánica de Fluidos
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2024

Trabajo Fin de Máster: Exposición y revisión del método SIMPLE con aplicaciones a flujos de geometría sencilla

Autor: Aitor Robles Corpa

Tutor: Miguel Pérez-Saborid Sánchez-Pastor

El tribunal nombrado para juzgar el trabajo arriba indicado, compuesto por los siguientes profesores:

Presidente:

Vocal/es:

Secretario:

acuerdan otorgarle la calificación de:

El Secretario del Tribunal

Fecha:

Agradecimientos

Este documento supone para mí el fin de otra etapa muy importante en mi vida, quizás una de las más complicadas y duras. Supone, además, la reafirmación de los valores con los que me educaron. La prueba de que, si se quiere algo, con esfuerzo, ganas y sacrificio, se pueden alcanzar todas las metas que uno se proponga en esta vida.

Por supuesto, no habría llegado hasta aquí solo, y son muchas las personas a las que me gustaría agradecer su apoyo, sus palabras y los buenos consejos.

Empezando por mi familia, por los que están y por los que se fueron, y en especial a mi madre. Sin su paciencia, cariño, comprensión y valores vitales, no sería quien soy.

A mi tutor, Don Miguel Pérez-Saborid Sánchez-Pastor, por haber confiado en mí una vez más para desarrollar este estudio. Gracias por su orientación y dedicación.

A mis amigos, que son pocos, pero son mi segunda familia. En especial a Loly y Desirée, que pese a la distancia que nos separa, siempre os he sentido cerca para cualquier cosa. Seguro que nos esperan muchas cosas buenas en el futuro que seguir disfrutando juntos. Y no me puedo olvidar de A.R., sin su consejo, quizás no habría comenzado esta aventura que estoy a punto de terminar.

A todos, muchas gracias.

Aitor Robles Corpa

Sevilla, 2024

Resumen

El campo de la Dinámica de Fluidos Computacional (CFD) ha experimentado un crecimiento significativo, siendo vital en la ingeniería y en particular en la industria aeronáutica. Este trabajo investiga el método SIMPLE (Semi-Implicit Method for Pressure-Linked Equations), como procedimiento base de las simulaciones de dominios fluidos. Se presenta una revisión exhaustiva de la literatura de CFD, seguida de la formulación matemática de las ecuaciones de Navier-Stokes y la implementación del algoritmo SIMPLE. Se resuelven casos de geometría sencilla y flujo incompresible, evidenciando su eficacia y versatilidad. Se discuten los resultados en comparación con la literatura existente, seguido de conclusiones y sugerencias para futuras investigaciones. Los anexos detallan el método line-by-line, las ecuaciones implementadas y los códigos en Matlab. Este estudio no solo profundiza en el método SIMPLE, sino que también sirve como recurso educativo introductorio en CFD, proporcionando las herramientas básicas necesarias para comprender y aplicar algoritmos de simulación de fluidos.

Abstract

The field of Computational Fluid Dynamics (CFD) has experienced a significant growth, being vital in engineering and in particular in the aeronautical industry. This paper examines the SIMPLE method (Semi-Implicit Method for Pressure-Linked Equations), as a basic procedure for fluid domain simulations. A exhaustive review of the CFD literature is presented, followed by the mathematical formulation of the Navier-Stokes equations and the implementation of the SIMPLE algorithm. Cases of simple geometry and incompressible flow are solved, demonstrating its efficiency and versatility. The results are discussed in comparison with available literature, followed by conclusions and suggestions for future research. The appendices detail the line-by-line method, the implemented equations and the Matlab codes. This study not only delves into the SIMPLE method, but also serves as an introductory educational resource in CFD, providing the basic tools necessary to understand and apply fluid simulation algorithms.

Índice

<i>Resumen</i>	III
<i>Abstract</i>	V
<i>Índice de Figuras</i>	XI
<i>Índice de Tablas</i>	XV
<i>Introducción</i>	XVII
1 Revisión bibliográfica	1
1.1 Historia sobre el desarrollo del CFD	2
1.2 Fases del CFD	3
1.3 Métodos de discretización	5
1.4 Método SIMPLE y sus variantes	7
1.4.1 SIMPLER	8
1.4.2 SIMPLEC	8
1.4.3 PISO	8
1.5 Tipos de mallado	9
1.5.1 Mallado estructurado: escalonado vs co-ubicado	12
2 Formulación matemática	17
2.1 Ecuación general de Advección-Difusión	17
2.1.1 Discretización de la ecuación general de advección-difusión para un volumen finito	18
2.2 Ecuaciones de Navier-Stokes para flujos incompresibles	23
2.2.1 Discretización y linealización de la ecuación de cantidad de movimiento para un volumen finito	25
2.2.2 Discretización y linealización de la ecuación de continuidad para un volumen finito	27
2.3 Desarrollo del método SIMPLE	28
2.3.1 Fundamentos del método SIMPLE	28
2.3.2 Algoritmo SIMPLE	28
2.3.2.1 Primera iteración	30
2.3.2.2 Cálculo de los coeficientes y fuentes para las ecuaciones de cantidad de movimiento	30
2.3.2.3 Resolución de las ecuaciones de cantidad de movimiento	31
2.3.2.4 Método PWIM para la interpolación de las velocidades en las caras	33
2.3.2.5 Cálculo de los coeficientes de enlace y fuentes para la ecuación de continuidad	35
2.3.2.6 Resolución de la ecuación de continuidad	38

2.3.2.7	Actualización de presiones y velocidades	39
2.3.2.8	Evaluación de la convergencia de los residuos	40
2.3.3	Implementación del algoritmo SIMPLE en Matlab	41
2.4	Particularización de las ecuaciones para las condiciones de contorno	41
2.4.1	Pared sin deslizamiento	42
2.4.2	Pared con deslizamiento	48
2.4.3	Entrada a velocidad uniforme	50
2.4.4	Salida a presión uniforme	52
3	Resolución computacional de dominios fluidos	57
3.1	Flujo en canal de sección constante	57
3.1.1	Descripción del problema	57
3.1.2	Condiciones de contorno	58
3.1.3	Resultados y análisis	59
3.2	Problema de cavidad impulsada	66
3.2.1	Descripción del problema	66
3.2.2	Condiciones de contorno	67
3.2.3	Resultados y análisis	68
3.3	Flujo en canal con cambio de sección	75
3.3.1	Descripción del problema	75
3.3.2	Condiciones de contorno	76
3.3.3	Resultados y análisis	77
4	Discusión de los resultados	83
4.1	Flujo en canal de sección constante	84
4.2	Problema de cavidad impulsada	90
4.3	Flujo en canal con cambio de sección	94
5	Conclusiones	103
Apéndice A	Método Line-by-line	107
Apéndice B	Condiciones de contorno	111
B.1	Ecuaciones generales para celdas interiores	112
B.2	Pared izquierda sin deslizamiento	115
B.3	Pared derecha sin deslizamiento	117
B.4	Pared inferior sin deslizamiento	119
B.5	Pared superior sin deslizamiento	121
B.6	Pared superior con deslizamiento	124
B.7	Entrada a velocidad constante	127
B.8	Entrada a presión constante	129
B.9	Salida a presión constante	131
Apéndice C	Códigos desarrollados	133
C.1	Código principal del algoritmo SIMPLE	134
C.2	Función de cálculo de los coeficientes de enlace y fuentes para las ecuaciones de cantidad de movimiento	140
C.3	Función para resolución de ecuación de cantidad de movimiento en X	148
C.4	Función para resolución de ecuación de cantidad de movimiento en Y	156

C.5	Función PWIM de interpolación de velocidades en las caras	163
C.6	Función de cálculo de los coeficientes de enlace y fuentes para la ecuación de continuidad	172
C.7	Función para resolución de ecuación de continuidad	176
C.8	Función de actualización de velocidades y presiones	182
C.9	Función de post-procesado	186
C.10	Función auxiliar: <i>TDMA</i>	190
	<i>Bibliografía</i>	191

Índice de Figuras

1.1	Ejemplo de mallas estructuradas [12]	9
1.2	Ejemplo de mallas estructuradas tipo O y C [12]	10
1.3	Ejemplo de malla estructurada multi-bloque [12]	10
1.4	Ejemplo de mallas no estructuradas [12]	11
1.5	Mallado estructurado co-ubicado	12
1.6	Ejemplo de problema 1D	13
1.7	Ilustración del <i>Checkerboard pressure oscillation</i>	13
1.8	Mallado estructurado escalonado	14
2.1	Teorema de Gauss	19
2.2	Caras y normales de la celda de referencia	20
2.3	Algoritmo SIMPLE	29
2.4	Ilustración del sistema de ecuaciones de la fila j	32
2.5	Aproximación de la interpolación de \hat{u}_e	34
2.6	Pared izquierda sin deslizamiento	42
2.7	Capa límite	44
2.8	Adaptación de velocidades u_e y v_n en condiciones cercanas a la pared	46
2.9	Pared superior con deslizamiento	48
2.10	Entrada del fluido a velocidad uniforme	50
2.11	Salida del fluido a presión uniforme	52
2.12	Celda cerca de la salida del fluido a presión constante	53
3.1	Canal de sección constante	58
3.2	Condiciones de contorno del canal de sección constante	59
3.3	Residuos y campo de presiones $Re = 100$	60
3.4	Residuos y campo de presiones $Re = 200$	60
3.5	Residuos y campo de presiones $Re = 300$	60
3.6	Residuos y campo de presiones $Re = 400$	61
3.7	Residuos y campo de presiones $Re = 450$	61
3.8	Campo de velocidades y perfiles, $Re = 100$	63
3.9	Campo de velocidades y perfiles, $Re = 200$	63
3.10	Campo de velocidades y perfiles, $Re = 300$	63
3.11	Campo de velocidades y perfiles, $Re = 400$	64
3.12	Campo de velocidades y perfiles, $Re = 450$	64
3.13	Comparación de perfiles de velocidad horizontal, $x = 0.0175m$	65
3.14	Comparación de perfiles de velocidad horizontal, $x = 0.025m$	65

3.15	Comparación de perfiles de velocidad horizontal, $x = 0.0375m$	65
3.16	Problema de cavidad impulsada	66
3.17	Condiciones de contorno del problema de cavidad impulsada	67
3.18	Residuos y líneas de corriente $Re = 100$	69
3.19	Residuos y líneas de corriente $Re = 200$	69
3.20	Residuos y líneas de corriente $Re = 500$	69
3.21	Residuos y líneas de corriente $Re = 1000$	70
3.22	Residuos y líneas de corriente $Re = 1500$	70
3.23	Residuos y líneas de corriente $Re = 2000$	70
3.24	Campo de presiones y velocidades $Re = 100$	72
3.25	Campo de presiones y velocidades $Re = 200$	72
3.26	Campo de presiones y velocidades $Re = 500$	72
3.27	Campo de presiones y velocidades $Re = 1000$	73
3.28	Campo de presiones y velocidades $Re = 1500$	73
3.29	Campo de presiones y velocidades $Re = 2000$	73
3.30	Comparación de perfiles de velocidad horizontal, $x = 0.05 m$	74
3.31	Flujo en canal con cambio de sección	75
3.32	Condiciones de contorno del canal con cambio de sección	76
3.33	Residuos y líneas de corriente $Re = 50$	78
3.34	Residuos y líneas de corriente $Re = 100$	78
3.35	Residuos y líneas de corriente $Re = 150$	78
3.36	Residuos y líneas de corriente $Re = 200$	79
3.37	Residuos y líneas de corriente $Re = 300$	79
3.38	Campo de velocidades y perfiles, $Re = 50$	81
3.39	Campo de velocidades y perfiles, $Re = 100$	81
3.40	Campo de velocidades y perfiles, $Re = 150$	81
3.41	Campo de velocidades y perfiles, $Re = 200$	82
3.42	Campo de velocidades y perfiles, $Re = 300$	82
4.1	Comparación de perfiles de velocidad a la salida con el perfil teórico de Poiseuille	85
4.2	Comparación de longitudes de entrada	86
4.3	Nuevos resultados para $Re = 300$	87
4.4	Determinación de la longitud de entrada para $Re = 300$	88
4.5	Comparación con perfil de Poiseuille para $Re = 300$	89
4.6	Comparación de la longitud de entrada para $Re = 300$	89
4.9	Comparación de resultados entre el método MacCormack y Hou et al [10]	92
4.10	Comparación de perfiles de velocidad para $Re = 100$	93
4.11	Comparación de perfiles de velocidad para $Re = 400$	93
4.12	Esquema de separación del flujo [15]	94
4.13	Validación de datos, Hossain et al. [15]	95
4.14	Resultados para $Re = 350$	96
4.15	Resultados para $Re = 400$	97
4.16	Resultados para $Re = 500$	98
4.17	Resultados de Hossain et al. [15] para $Re = 100$	99
4.18	Resultados de Hossain et al. [15] para $Re = 350$	99
4.19	Resultados de Hossain et al. [15] para $Re = 500$	99
A.1	Mallado para resolución mediante el método line-by-line con barrido vertical	107
B.1	Celda interior de referencia	112
B.2	Pared izquierda sin deslizamiento	115

B.3	Pared derecha sin deslizamiento	117
B.4	Celda cerca de la pared derecha	118
B.5	Pared inferior sin deslizamiento	119
B.6	Pared superior sin deslizamiento	121
B.7	Celda cerca de la pared superior	122
B.8	Pared superior con deslizamiento	124
B.9	Celda cerca de la pared superior con deslizamiento	125
B.10	Entrada del fluido a velocidad constante	127
B.11	Entrada del fluido a presión constante	129
B.12	Salida del fluido a presión constante	131
B.13	Celda cerca de la salida del fluido a presión constante	132

Índice de Tablas

3.1	Canal longitudinal constante: iteraciones y residuos en función de Re	61
3.2	Cavidad impulsada: iteraciones y residuos en función de Re	68
3.3	Cambio de sección: iteraciones y residuos en función de Re	77
4.1	Error relativo de la velocidad máxima	85
4.2	Error relativo de la longitud de entrada	86
4.3	Error relativo de la velocidad máxima para $Re = 300$	88
4.4	Error relativo de la longitud de entrada para $Re = 300$	88
4.5	Errores relativos de la posición del centro de rotación	91

Introducción

El campo de la Dinámica de Fluidos Computacional (CFD, por sus siglas en inglés) ha experimentado un crecimiento exponencial en su relevancia y aplicación en la industria aeronáutica y en diversos campos de la ingeniería. Este crecimiento se ha visto impulsado por la capacidad de los métodos numéricos para simular y predecir el comportamiento de dominios fluidos con una precisión cada vez mayor, lo que ha permitido optimizar diseños, reducir costos y acelerar el desarrollo de productos.

El presente Trabajo Fin de Máster surge de la motivación de explorar y comprender a fondo cómo funcionan códigos comerciales de simulación de dominios fluidos como pueden ser ANSYS Fluent, Autodesk CFD o CFD++. En este sentido uno de los primeros métodos desarrollados fue el método SIMPLE (Semi-Implicit Method for Pressure-Linked Equations). Este método sirve como base para muchos de los programas comerciales disponibles en la actualidad. Por ende, este estudio se centra en analizar en detalle este algoritmo y comprender su implementación en la práctica de la simulación computacional de fluidos.

El objetivo principal de este estudio es proporcionar una comprensión detallada de las ecuaciones y algoritmos subyacentes al método SIMPLE, así como su aplicación práctica mediante la implementación de un código computacional. Este código se utiliza para resolver tres casos de estudio representativos: un canal longitudinal de sección constante, un problema de cavidad impulsada y un canal con cambio de sección, con el fin de ilustrar la eficacia y versatilidad del método en diferentes configuraciones.

Es importante destacar que, si bien existen numerosos recursos y herramientas comerciales disponibles para la simulación de flujos fluidos, la comprensión profunda de los fundamentos teóricos y la capacidad para desarrollar y modificar códigos personalizados son habilidades esenciales para cualquier estudiante o investigador interesado en este campo. Este trabajo pretende proporcionar una síntesis clara y concisa de la literatura existente sobre los aspectos teóricos y prácticos del CFD, con el propósito de orientar y facilitar el aprendizaje y desarrollo de habilidades en este ámbito. Es de esperar que el estudiante o investigador interesado pueda utilizar este proyecto para desarrollar sus propios códigos sin necesidad de recurrir a cajas negras como son los códigos comerciales.

El documento se ha estructurado en seis capítulos que tratan distintos aspectos del estudio realizado. En el primer capítulo, se lleva a cabo una revisión exhaustiva de la literatura sobre Dinámica de Fluidos Computacional (CFD), explorando la historia de su desarrollo, clasificando los métodos utilizados y proporcionando una explicación conceptual del algoritmo SIMPLE y sus variantes, así como una discusión sobre el mallado del dominio. El segundo capítulo se centra en la formulación matemática fundamental, partiendo de la ecuación general de advección-difusión y desarrollando

las ecuaciones de Navier-Stokes, junto con su discretización. Este capítulo es crucial para comprender los fundamentos del método SIMPLE, detallando cada paso del algoritmo y adaptando las ecuaciones a diversas condiciones de contorno. En el tercer capítulo, se presentan los tres problemas seleccionados para ser resueltos con el código implementado, describiendo sus características y condiciones de contorno, y analizando los resultados obtenidos. El cuarto capítulo se dedica a la discusión de estos resultados, comparándolos con los hallazgos de la literatura actual. Finalmente, el quinto capítulo recoge las conclusiones derivadas del estudio y propone líneas futuras de investigación. Además, el documento incluye tres anexos: el Anexo A aborda los fundamentos del método line-by-line necesario para resolver los sistemas de ecuaciones, el Anexo B proporciona un resumen de las ecuaciones implementadas del método SIMPLE y un compendio de los ajustes requeridos según las condiciones de contorno, y el Anexo C contiene los códigos desarrollados en Matlab.

En conclusión, este estudio no solo se centra en profundizar en el método SIMPLE y su aplicación práctica, sino que también aspira a servir como un recurso introductorio y educativo en el fascinante campo de la dinámica de fluidos computacional (CFD). Más allá de simplemente presentar los fundamentos teóricos y prácticos del método SIMPLE, este trabajo tiene como objetivo proporcionar una base sólida para aquellos que se inician en el campo de la CFD, brindándoles las herramientas necesarias para comprender, implementar y adaptar algoritmos y técnicas de simulación de fluidos.

1 Revisión bibliográfica

Antes de abordar cualquier proyecto en ingeniería, es esencial realizar una investigación preliminar sobre el tema en cuestión, familiarizándose con los conceptos y terminología empleada en dicho campo. El propósito de este primer capítulo es hacer una revisión de la bibliografía actual sobre la Dinámica de Fluidos Computacional (CFD), para proporcionar al lector una base sobre los fundamentos de esta disciplina, asegurando así una mejor comprensión del presente estudio. Se comenzará por un breve repaso histórico sobre el desarrollo de los métodos CFD, para después pasar a clasificarlos según el tipo de metodología empleada. Posteriormente, se centrará la explicación en el método SIMPLE, que es el que se desarrollará más en profundidad durante todo este estudio, y se abordarán sus variantes. Finalmente, se dará una visión sobre los tipos de mallados y se justificará el uso del mallado co-ubicado frente al mallado escalonado.

La dinámica de fluidos computacional (CFD) es una herramienta de análisis utilizada por los ingenieros desde hace casi cincuenta años. Como explica Charesworth [5], en sus inicios, el CFD se aplicó a problemas de aerodinámica, pero pronto se comprendieron sus ventajas y se empezó a emplear en todas las disciplinas de la ingeniería para resolver una gran variedad de problemas. El gran logro del CFD es su capacidad para predecir, con una precisión razonable, los detalles de complejos patrones de flujo que afectan a problemas prácticos de ingeniería. Esto se consigue sin realizar pruebas con modelos en túneles de viento, tanques de agua o instalaciones experimentales similares. En su lugar, el CFD utiliza la capacidad de los ordenadores para resolver las ecuaciones que rigen la dinámica de fluidos mediante técnicas numéricas. Resulta evidente que los cálculos numéricos pueden realizarse en un ordenador más rápidamente y con menos esfuerzo que los tradicionales ensayos experimentales con modelos o maquetas. Sin embargo, la fiabilidad de estos cálculos está condicionada por su validación experimental dentro de ciertos parámetros y condiciones. Aun así, tal como recalca Charesworth [5], el CFD ofrece una herramienta de análisis mucho más rentable que las pruebas experimentales. Esta eficiencia permite una reducción significativa del tiempo necesario para analizar un problema de flujo de fluidos, lo que facilita un estudio más profundo y una mejor optimización del diseño técnico. Aunque los ensayos con modelos siguen y seguirán siendo una parte importante de la ingeniería, el CFD se ha convertido cada vez más en una herramienta más accesible y esencial que ya forma parte de los procesos normales de diseño y optimización.

Como es lógico, el desarrollo de estas herramientas está vinculado al avance de la capacidad computacional de los ordenadores. En los próximos años, es de esperar que, con la incorporación de la inteligencia artificial y la computación cuántica, el CFD experimente un avance significativo tanto en precisión como en reducción de tiempos de cálculo. Este crecimiento proyectado en las capacidades del CFD promete transformar aún más la ingeniería, ofreciendo soluciones más rápidas, precisas y económicas para el análisis y diseño de sistemas complejos de flujo de fluidos.

1.1 Historia sobre el desarrollo del CFD

La Dinámica de Fluidos Computacional (CFD) ha experimentado una evolución impresionante desde sus modestos comienzos hasta, como ya se ha mencionado en varias ocasiones, convertirse hoy en día en una herramienta indispensable en la ingeniería moderna. A lo largo de décadas de investigación y desarrollo, el CFD ha revolucionado la forma en que se comprenden y diseñan sistemas que involucran flujo de fluidos, desde aviones y automóviles hasta procesos industriales y aplicaciones biomédicas.

Los comienzos del CFD se remonta a los años 60, cuando los primeros investigadores comenzaron a aplicar métodos numéricos para resolver las ecuaciones que describen el movimiento de los fluidos. Durante este periodo, es cuando se produce el desarrollo de las bases iniciales del CFD, ya que se desarrollan el método de Volumen Finito y el método de Diferencias Finitas por Harlow y Welch en 1965, que permitieron discretizar el dominio del fluido y aproximar las soluciones de las ecuaciones diferenciales. Sin embargo, uno de los hitos más significativos en el campo del CFD se produce cuando Richard Courant, en colaboración con IBM, empezaron a implementar estos métodos en los primeros computadores de aquella época para resolver flujos muy sencillos.

Más adelante, en la década de los años 70, es cuando se produce otro de los mayores avances en los primeros pasos del CFD, puesto que Brian Patankar y Suhas Spalding, como parte de su tesis doctoral en el Imperial College de Londres, desarrollan el método SIMPLE (Semi-Implicit Method for Pressure-Linked Equations) [13]. Fue el primer algoritmo implícito desarrollado para la resolución de las ecuaciones de Navier-Stokes, aportando una gran estabilidad numérica, eficiencia y convergencia de las soluciones para flujos de fluidos incompresibles, hasta entonces planteados. De hecho, para validar su metodología, simulaban los problemas de cavidad impulsada y estudio de la estela tras un escalón.

El método SIMPLE, marcó un punto de inflexión en el desarrollo del CFD, ya que permitió a los ingenieros y científicos abordar problemas más complejos y realistas [5]. Con el tiempo, se han desarrollado variantes más sofisticadas del método SIMPLE, como PISO (Pressure-Implicit with Splitting of Operators) y SIMPLEC (SIMPLE-Consistent), que han mejorado aún más la eficiencia y la precisión de las simulaciones, siendo hoy en día, la base de la mayoría de los softwares comerciales CFD disponibles en el mercado (ANSYS Fluent, Star-CCM+, CFX, CDF-ACE).

Paralelamente al desarrollo de los métodos numéricos, también fueron creados modelos de turbulencia para intentar reflejar, de la manera más fiel posible, los efectos turbulentos en los flujos. Algunos de estos modelos fueron el modelo $k - \epsilon$ y el modelo de esfuerzo de Reynolds. Estos modelos han sido fundamentales para simular flujos turbulentos, porque permitieron ampliar el rango de aplicación de los códigos basados en SIMPLE, ya que añaden más ecuaciones para cambiar las condiciones del problema y tratar los efectos de la turbulencia. Gracias a estos avances, fue posible simular aplicaciones industriales cada vez más reales, como la aerodinámica de aviones y automóviles, así como en la ingeniería de procesos.

Durante los años 80, los avances en el campo del CFD fueron dirigidos a encontrar soluciones para casos con geometrías más complejas [9]. Esta necesidad surgió debido a las limitaciones de los métodos tradicionales, que utilizaban mallados estructurados basados en formas ortogonales, los cuales eran inadecuados para representar con precisión configuraciones geométricas complicadas. Para superar estas limitaciones, se desarrollaron técnicas avanzadas de generación de mallados no estructurados, que permitían una mejor adaptación a las formas irregulares. Estos mallados, basados en elementos triangulares y tetraédricos, ofrecían una flexibilidad superior, permitiendo modelar

con mayor precisión las características complejas de los fluidos y las superficies.

En los años 90 y principios de los 2000, marcaron un periodo crucial para el CFD, caracterizado por una revolución informática. La llegada y proliferación de ordenadores personales más rápidos y con mayor capacidad de cálculo transformaron el campo. Estas mejoras en hardware permitieron realizar simulaciones más detalladas y precisas, reduciendo significativamente los tiempos de procesamiento. La disponibilidad de recursos computacionales más potentes facilitó el desarrollo y la implementación de métodos más sofisticados, basados en las técnicas de volumen finito, elementos finitos y diferencias finitas.

En la actualidad, el desarrollo de software de CFD ha avanzado notablemente, enfocándose no solo en la capacidad de simulación, sino también en la accesibilidad y usabilidad para los usuarios. Las interfaces gráficas de usuario han sido clave en la mejora de estos softwares, haciendo que los programas de CFD sean más intuitivos y accesibles incluso para aquellos con menos experiencia en el campo. Estas interfaces permiten a los usuarios configurar simulaciones complejas de manera más sencilla y visual, reduciendo la barrera de entrada y facilitando el uso de estas herramientas. Además, la integración de capacidades de pre y post-procesamiento dentro de los softwares ha simplificado el flujo de trabajo, permitiendo una visualización y análisis más eficiente de los resultados.

La influencia del método SIMPLE en los modelos actuales de CFD es innegable. Muchos de los softwares comerciales y herramientas de simulación utilizadas en la industria se basan en algoritmos derivados del método SIMPLE. Su capacidad para resolver problemas de flujo de manera eficiente y precisa ha permitido avances significativos en áreas como el diseño de vehículos, la aerodinámica, la ingeniería de procesos y la biomecánica.

Además de los avances en métodos numéricos y modelos de turbulencia, el desarrollo del hardware y el software ha sido crucial para el crecimiento del CFD. El aumento en la capacidad de computación ha permitido resolver problemas más grandes y complejos, mientras que el desarrollo de software especializado ha hecho que las simulaciones sean más accesibles y fáciles de usar para una amplia gama de usuarios. El desarrollo histórico y futuro del CFD ha sido y será impulsado por una combinación de avances en los métodos numéricos, modelos de turbulencia, hardware y software.

1.2 Fases del CFD

De forma general, el proceso de resolución de cualquier dominio fluido mediante CFD siguen el mismo flujo de trabajo, el cual se puede dividir en cuatro fases:

1. Modelado del Problema

El primer paso en una simulación CFD es modelar el problema. Esto implica dos partes: un modelo geométrico del dominio fluido y, un modelo matemático que capture el comportamiento físico del fluido. Para el modelo geométrico, basta con definir el espacio físico donde se llevará a cabo la simulación, creando la geometría mediante un modelo 2D o 3D. En cuanto al modelo matemático, consiste en identificar las ecuaciones que gobiernan el fenómeno que se desea estudiar y establecer las condiciones de contorno adecuadas.

Las ecuaciones fundamentales que se deben tener en cuenta para aplicar o no, son las siguientes:

- **Ecuación de Cantidad de Movimiento:** describe cómo cambia el movimiento del fluido debido a las fuerzas aplicadas.
- **Ecuación de Continuidad:** garantiza la conservación de la masa en el sistema, indicando que la masa que entra y sale de un volumen de control debe estar equilibrada.
- **Ecuación de Energía:** considera la conservación de la energía dentro del fluido, importante para problemas donde los efectos térmicos son significativos y también para flujos compresibles.
- **Ecuación de Estado:** relaciona propiedades termodinámicas del fluido, como presión, densidad y temperatura.
- **Ecuaciones del Modelo Turbulento:** en problemas de flujo turbulento, se requieren ecuaciones adicionales para capturar los efectos de la turbulencia, como por ejemplo el modelo $k - \epsilon$ o $k - \omega$, que tratan de modelar la viscosidad asociada a la turbulencia.

2. Mallado del dominio fluido

El siguiente paso es el mallado del dominio fluido. Este proceso implica dividir la geometría del dominio en celdas o volúmenes de control más pequeños, donde se ubicarán los nodos para calcular los valores de las incógnitas. Los nodos pueden ubicarse en el centro de la celda o volumen o en los vértices. Los tipos de mallado pueden ser:

- **Estructurado:** las celdas o volúmenes siguen un patrón regular, facilitando la implementación y el cálculo, pero limitando la flexibilidad en geometrías complejas.
- **No Estructurado:** las celdas o volúmenes no siguen un patrón regular, permitiendo mayor flexibilidad y adaptación a geometrías complicadas, aunque con un mayor costo computacional.

Como se explicará más adelante, este paso es fundamental, pues dependiendo de la calidad y densidad del mallado afectan directamente a la precisión y estabilidad de las simulaciones.

3. Resolución del dominio fluido

Una vez mallado el dominio, el próximo paso es discretizar las ecuaciones del problema en el mallado para plantear y resolver los sistemas algebraicos de forma numérica. En la próxima sección, se detallarán los tipos de métodos numéricos que se emplean para la resolución de los sistemas de ecuaciones. Dentro de los softwares comerciales, este paso se correspondería con la configuración de los métodos de resolución, criterios de convergencia, número de iteraciones y simulación del dominio.

4. Post-procesado

El post-procesado implica el análisis y la evaluación de los resultados obtenidos de la simulación. Esto incluye:

- **Visualización de Resultados:** a través de gráficas y representaciones visuales, como líneas de corriente, mapas de velocidad, distribuciones de presión y fuerzas.
- **Análisis de Datos:** evaluación cuantitativa y cualitativa de los resultados para verificar su coherencia con el fenómeno físico modelado y con los datos experimentales disponibles.

Esta última fase permite obtener una comprensión detallada del comportamiento del fluido dentro del dominio y es fundamental para la validación y optimización del diseño o del modelo estudiado.

En conclusión, una simulación CFD abarca desde la definición precisa del problema y su modelado, pasando por la generación de un mallado adecuado, la resolución de las ecuaciones discretizadas mediante un método numérico y culmina con el análisis exhaustivo de los resultados representados en la fase de post-procesado. Cada uno de estos pasos es esencial para garantizar la precisión y la utilidad de la simulación en aplicaciones prácticas.

1.3 Métodos de discretización

Si se pone el foco en el paso de la resolución del dominio fluido, las ecuaciones fundamentales que deben ser resueltas, son las ecuaciones de Navier-Stokes (NSE) que son las que representan el comportamiento de un fluido. Las NSE son la ecuación de cantidad de movimiento y la ecuación de continuidad, a las que, dependiendo del tipo de problema, se debe incluir la ecuación de energía. Todas estas ecuaciones derivan de la ecuación general de convección-difusión que es una ecuación diferencial que contiene cuatro términos principales: un término temporal, un término convectivo, un término difusivo y términos de fuentes externas. Los términos convectivo y difusivo son particularmente complicados de resolver debido a que involucran derivadas espaciales [2].

Por este motivo, la estrategia general para abordar estos términos convectivo y difusivo, consiste en discretizar las derivadas espaciales, reemplazándolas con expresiones algebraicas que faciliten su resolución computacional. Las técnicas de discretización empleadas en la dinámica de fluidos computacional (CFD) y otras disciplinas se pueden resumir en los siguientes enfoques[4]:

Diferencias finitas

Las derivadas espaciales se aproximan mediante operadores algebraicos obtenidos de series truncadas de Taylor. Esto implica que, en cada nodo de la malla, el valor de su derivada, dependerá del valor de los nodos adyacentes [11].

$$f'(x_i) \approx \frac{f(x_{i+1}) - f(x_i)}{\Delta x} \quad (1.1)$$

Volúmenes finitos

En lugar de formular la ecuación en forma diferencial, se plantea en forma integral. Se define un conjunto de volúmenes finitos en el dominio fluido sobre los cuales se impone el cumplimiento de la ecuación integral. Para cada volumen finito se calcula el flujo a través de las caras y este se utiliza como condición de contorno para los volúmenes adyacentes para cumplir con la continuidad. De esta manera se crea así un sistema de ecuaciones algebraico que poder resolver más fácilmente [5].

Elementos finitos

Este método intenta expresar la solución como la suma ponderada de funciones de forma. Dependiendo de la forma de estas funciones, se obtienen las distintas variables del método de los elementos finitos. Generalmente, estas funciones tienen carácter nodal y son nulas en todo el dominio excepto en las proximidades del nodo que representan. Es un método muy usado en mecánica de sólidos por su considerable flexibilidad discretizando una geometría complicada en elementos mucho más simples [11].

Diferencias finitas compactas

Son una extensión de los métodos de diferencias finitas utilizados en la solución numérica de ecuaciones diferenciales. Relaciona los valores de la derivada en varios nodos con el valor de la variable en los mismos nodos, de manera que, para obtener la derivada, se debe resolver un sistema n -diagonal de ecuaciones algebraicas. Estos métodos proporcionan mejoras considerables en cuanto a precisión y estabilidad.

$$\alpha f'(x_{i-1}) + f'(x_i) + \alpha f'(x_{i+1}) \approx \frac{\beta}{2\Delta x} (f(x_{i+1}) - f(x_{i-1})) \quad (1.2)$$

Métodos espectrales

Estos métodos aproximan la solución mediante su desarrollo en una base de un espacio funcional específico. Esto quiere decir, que, mediante muestras experimentales, intentan aproximar mediante una función que represente dicho comportamiento, por ejemplo, usando series de Fourier, series de Chebyshev o incluso polinomios de Lagrange. La principal ventaja de estos métodos es su precisión espectral. Una vez aproximada la función, sus derivadas se obtienen de forma exacta. Sin embargo, los puntos de discretización están determinados por el método de aproximación que se use, y las geometrías a resolver deben ser extremadamente simples [4].

Estas técnicas de discretización permiten resolver las complejas ecuaciones involucradas en la dinámica de fluidos computacional, facilitando la obtención de soluciones precisas y eficientes.

Tradicionalmente en CFD se ha usado el método de volúmenes finitos debido a que se trata de un método intuitivo y coherente con la física del problema. La cantidad de materia contenida en cada volumen cambia porque entra o sale materia de ese volumen de control debido a las acciones de las fuentes exteriores o al cambio temporal de las variables. Por ello se dice que es un método conservativo. Además, tiene relación directa con las magnitudes físicas (velocidad, presión, flujos, densidad...) y es flexible para adaptarse a cualquier geometría. Todas estas ventajas, han hecho que, durante años, se haya estado usado este enfoque en la resolución de los campos fluidos, aunque como se ha presentado, han surgido nuevos métodos con otros enfoques diferentes. En este estudio, se seguirá el enfoque tradicional, usando la integración en volúmenes finitos para representar el dominio fluido.

1.4 Método SIMPLE y sus variantes

Llegados a este punto, es claro que las ecuaciones de Navier-Stokes (NSE) y la ecuación de energía, deben ser discretizadas para poder resolverse de forma iterativa en el dominio fluido. Esta discretización se realizará usando el enfoque tradicional de los volúmenes finitos. Sin embargo, esto no es suficiente para poder resolver completamente los dominios fluidos, pues si se intenta resolver la ecuación de cantidad de movimiento para conocer las velocidades, primero se debe conocer la presión, ya que actúa como una fuente en dicha ecuación. De las ecuaciones presentadas, no existe ninguna relación directa, es decir, explícita, para obtener el campo de presiones. No obstante, sí que se dispone de la ecuación de continuidad, con la cual, mediante desarrollos matemáticos, se puede lograr llegar a una relación directa con el campo de presiones del dominio. De este hecho, se dieron cuenta Brian Patankar y Suhas Spalding, logrando desarrollar el método SIMPLE (*Semi-Implicit Method for Pressure-Linked Equations*) [5]. Con este método lograron, de manera implícita, compatibilizar el campo de presiones expresado en la ecuación de continuidad, con las velocidades de la ecuación de cantidad de movimiento.

Este método se enmarca dentro de los métodos basados en volúmenes finitos y proporciona una forma efectiva de compatibilizar las ecuaciones de cantidad de movimiento y continuidad mediante una estrategia iterativa de corrección de presión. Es importante entender que, no se trata de un método que resuelva las ecuaciones del campo fluido, si no de un método que compatibiliza la ecuación de cantidad de movimiento y la de continuidad. El motivo es porque dado un campo de presión, no hay dificultad en resolver las ecuaciones de cantidad de movimiento. La clave está en que si se logra un campo de presiones que satisface la ecuación de continuidad, el campo de velocidades derivado de la resolución de la cantidad de movimiento será un campo de velocidades físicamente consistente y que también cumplirá con la continuidad.

El algoritmo SIMPLE aborda este problema mediante un proceso de corrección iterativo del campo de presión. Aunque en el próximo capítulo se desarrollarán más en detalle cada uno de los pasos del proceso junto con su desarrollo matemático, es importante conocer los pasos básicos para poder compararlo con las variantes que surgieron a posteriori:

1. Estimación Inicial del Campo de Presión y velocidades
2. Solución de las Ecuaciones de Momento
3. Corrección de la Presión
4. Actualización del Campo de Presión y Velocidades
5. Evaluación de la convergencia
6. Si no converge la solución, repetir desde el paso 2 con los nuevos valores de presión y velocidad.

Por lo tanto, como se ha visto, la clave del método SIMPLE reside en la reformulación de la ecuación de continuidad como una ecuación de corrección de presión. Este enfoque permite que, al aplicar la corrección de presión, el campo de velocidades resultante satisfaga tanto las ecuaciones de cantidad de movimiento como la ecuación de continuidad. De esta manera, el algoritmo SIMPLE transforma la especificación indirecta de la presión en un procedimiento directo e iterativo que asegura la compatibilidad de las ecuaciones fundamentales del flujo de fluidos.

Como ya se ha mencionado en varias ocasiones, el método SIMPLE es ampliamente utilizado debido a su robustez y simplicidad relativa, sin embargo, con el tiempo, surgieron variantes y mejoras del algoritmo, como SIMPLER, SIMPLEC y PISO.

A continuación, se revisan y explican las diferencias fundamentales de cada una de las variantes.

1.4.1 SIMPLER

El método SIMPLER (*SIMPLE Revised*) es una mejora respecto del método SIMPLE. Fue desarrollado también por Brian Patankar para aumentar la tasa de convergencia de su método predecesor. La principal diferencia es que se comienza por calcular unas pseudo-velocidades iniciales en base a un campo de presiones preliminar calculado de manera directa, en lugar de empezar por una suposición de ambos campos. Esto hace que la estimación inicial sea mucho mejor y, por tanto, el número de iteraciones hasta llegar a la solución, disminuye, consiguiendo una mejor convergencia del método.

Los pasos simplificados del método SIMPLER:

1. Cálculo de pseudo-velocidades:
2. Resolución de la ecuación de presión
3. Resolución de ecuaciones de momento
4. Corrección de presión
5. Actualización del Campo de velocidades
6. Evaluación de la convergencia.
7. Si la solución no converge, se repite el proceso desde el paso 2.

1.4.2 SIMPLEC

El método SIMPLEC (*SIMPLE Consistent*) es la segunda variante del método SIMPLE, que fue desarrollada por Van Doormal y Raithby [2]. Los pasos del algoritmo SIMPLEC son esencialmente los mismos que SIMPLE. La diferencia entre ambos métodos está en la forma que tienen de discretizar las ecuaciones de momento. Como se verá durante el desarrollo matemático, en un momento dado es necesario aproximar las correcciones de velocidades. Es en ese punto, donde ambos métodos difieren. En el caso de SIMPLEC, la expresión de aproximación se simplifica mucho más mediante la eliminación de los términos menos significativos. Con esta simplificación, se consigue un algoritmo más eficiente en ciertos casos, logrando una convergencia más rápida que SIMPLE.

1.4.3 PISO

Por último, el método PISO (Pressure Implicit with Splitting of Operators) se considera una extensión del método SIMPLE. Se basa en un método que originalmente estaba desarrollado para el cálculo no iterativo de flujos compresibles no estacionarios. Issa lo adaptó y convirtió en un método iterativo para flujos estacionarios incompresibles. Se diferencia principalmente en que incluye 2 pasos de corrección para la presión, añadiendo pasos adicionales al algoritmo para mejorar la precisión y la convergencia.

1. Estimación Inicial del Campo de Presión y velocidades
2. Solución de las Ecuaciones de Momento
3. Corrección de la Presión
4. Actualización del Campo de velocidades

5. Corrección de la Presión
6. Actualización del Campo de velocidades
7. Evaluación de la convergencia
8. Si no converge la solución, repetir desde el paso 2 con los nuevos valores de presión y velocidad.

El algoritmo SIMPLE es muy sencillo conceptualmente y es la base de muchos códigos CFDs actuales. Sus variantes están enfocadas a mejorar la convergencia y ahorrar tiempo de computación. Sin embargo, Charlesworth [5] señala que las comparaciones han demostrado que el rendimiento de cada algoritmo depende de las condiciones de flujo, del grado de acoplamiento entre la ecuación de momento y las ecuaciones escalares, y de la cantidad de sub-relajación utilizada. Por lo tanto, dadas todas estas variables, es complicado afirmar que un método sea mejor que otro y no queda más remedio que elegir un algoritmo en particular según el caso.

1.5 Tipos de mallado

Por último, pero no menos importante, es necesario echar un vistazo a los tipos de mallas usados en CFD. Como se ha explicado en esta revisión bibliográfica, el uso de una malla es fundamental para poder resolver el dominio fluido, pues es la forma en la que se debe subdividir el dominio en una serie de pequeñas celdas o volúmenes. Esta división es la que permite discretizar las ecuaciones que rigen el flujo de fluidos y convertirlas en un gran sistema de ecuaciones algebraicas simultáneas que pueden resolverse numéricamente [5]. En este sentido, como ya se adelantó en apartados anteriores, las mallas se suelen dividir en mallados estructurados y no estructurados.

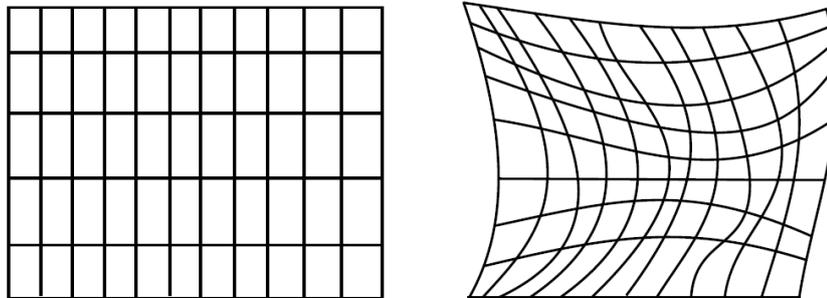


Figura 1.1 Ejemplo de mallas estructuradas [12].

Por su sencillez, las primeras técnicas CFD que se desarrollaron utilizaban mallas ortogonales estructuradas. Estas mallas se construyen típicamente a partir de cuadrados o rectángulos en dos dimensiones y cubos o prismas en tres dimensiones y, obviamente, son relativamente fáciles de construir. Su principal ventaja es que su estructura lineal facilita mucho su implementación en códigos informáticos para resolver las ecuaciones algebraicas resultantes del proceso de discretización. Sin embargo, aunque las mallas ortogonales estructuradas presentan ventajas significativas, éstas se ven mitigadas por el número limitado de dominios que pueden mallarse utilizando cuadrados o cubos. En la mayoría de las aplicaciones de ingeniería de interés práctico, los fluidos se encuentran en geometrías complejas que requieren técnicas de mallado más sofisticadas.

Los primeros intentos de encontrar técnicas de generación de mallas capaces de mallar geometrías más complejas seguían girando en torno a las mallas estructuradas. La clave fue que se eliminó la restricción ortogonal de la malla, lo que dio lugar a técnicas de generación de mallas libres de ajustarse a formas más complicadas. Los tipos de mallas que surgieron entonces se clasifican como mallas O, C y H. Algunos ejemplos se ilustran en la figura 1.2. Otra técnica que surgió basada también en mallas estructuradas, son las mallas multi-bloque, donde hay uno o más niveles de subdivisión. Normalmente en el nivel exterior, hay bloques generalmente grandes y se reserva el nivel más fino para las zonas de acoplamiento entre regiones o de especial interés en el estudio.

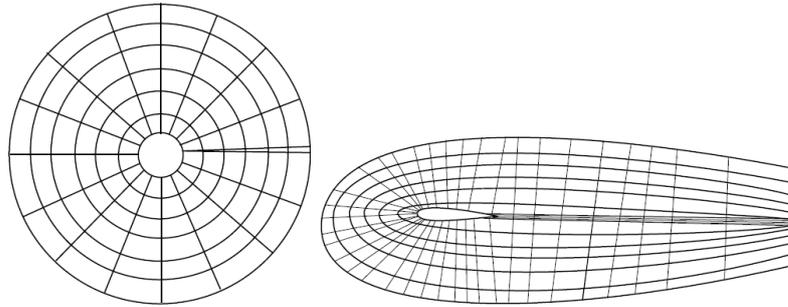


Figura 1.2 Ejemplo de mallas estructuradas tipo O y C [12].

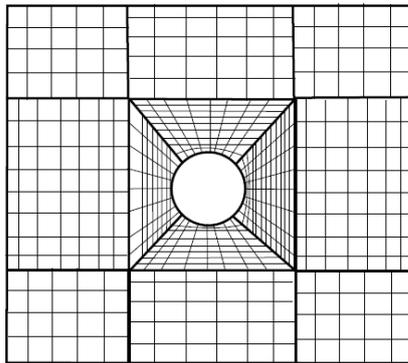


Figura 1.3 Ejemplo de malla estructurada multi-bloque [12].

Este tipo de mallas son más flexibles que las estructuradas y permite usar mayor resolución en aquellas regiones donde es necesario, aunque por contrapartida son más complejas de programar [12].

Para lograrlo, fue necesario replantearse por completo la forma de generar las mallas. El resultado fue el desarrollo de técnicas de generación de mallas no estructuradas que permiten mallar casi cualquier configuración geométrica actual. La mayoría de estas técnicas utilizan triángulos y tetraedros, en dos y tres dimensiones respectivamente, como elemento básico para construir las mallas. Así surgieron las mallas no estructuradas, como por ejemplo las mostradas en la figura 1.4.

Evidentemente, este cambio de elemento básico obliga a desestructurar la malla, ya que mantener cualquier estructura lineal es imposible, a no ser que usen triángulos muy regulares, por supuesto, no es posible en absoluto con tetraedros. Sin embargo, este cambio es el que permite a las mallas no estructuradas discretizar geometrías muy complejas. Además, las mallas no estructuradas también

facilitan enormemente el refinamiento adaptativo como parte del proceso de solución. Esta técnica consiste en utilizar información de la solución encontrada en una malla para crear una nueva malla con áreas localizadas más refinadas donde existen patrones de flujo complejos. Es evidente, que este método iterativo, permite que las ecuaciones de gobierno controlen lo fina o gruesa que debe ser una malla en una región determinada y, por lo tanto, conducen a soluciones más precisas. Tanto la flexibilidad geométrica, como la posibilidad del refinamiento adaptativo favorecen otra ventaja más sobre las mallas estructuradas, que es una reducción significativa del tiempo y el esfuerzo necesarios para generar las mallas. Según señala Charlesworth [5], a menudo, en los cálculos CFD basados en mallas estructuradas, el proceso de generación de la malla constituye en más de un cincuenta por ciento al tiempo total necesario para alcanzar una solución. Por lo tanto, cualquier técnica que reduzca el tiempo, supone una ventaja importante, puesto que eso permite bien realizar más simulaciones de un problema determinado o bien, reducir el coste de realización de un estudio específico.

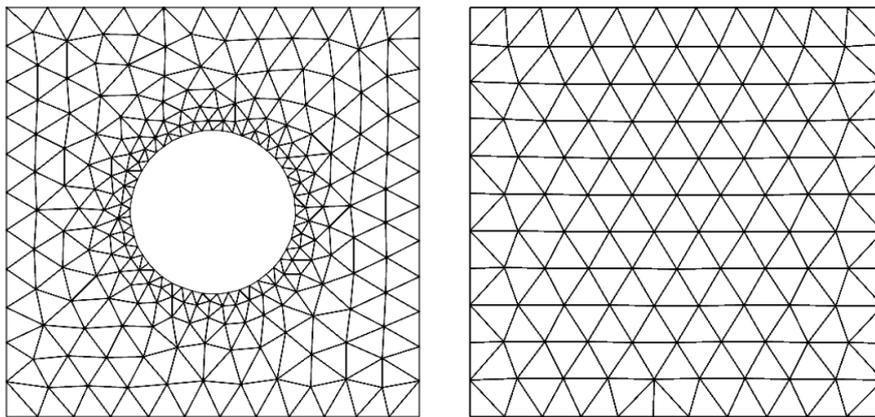


Figura 1.4 Ejemplo de mallas no estructuradas [12].

A primera vista, las mallas no estructuradas ofrecen muchas ventajas para una buena estrategia de generación de mallas, pero inevitablemente también presentan desventajas. El principal inconveniente es la falta de estructura lineal en las mallas, lo que suponen una mayor carga computacional por tener que almacenar los datos topológicos de la malla. Debido a esta falta de estructura, la discretización de las ecuaciones se vuelve mucho más compleja y requiere de nuevas técnicas de discretización para poder lograr los mismos sistemas algebraicos. Sin embargo, pese a ello, y gracias a los avances informáticos, las mallas no estructuradas son las que generalmente se emplean en la actualidad en cualquier programa CFD [9].

En conclusión, para cualquier estudio CFD, la elección de un mallado adecuado es fundamental a la hora de obtener una solución lo más exacta posible y, sobre todo, en términos de ahorro de tiempo, ya que no solo importa el tiempo empleado en la construcción del mallado sino también el tiempo de cálculo que depende del mallado seleccionado. En este sentido y dados los objetivos del presente estudio, se ha optado por un mallado estructurado, ya que al tratarse de ser un manual introductorio a CFD, la sencillez de la discretización permite mostrar perfectamente el proceso de discretización de las ecuaciones NSE en volúmenes finitos.

1.5.1 Mallado estructurado: escalonado vs co-ubicado

En este punto, el lector debe tener claro que este estudio desarrollará el método de volúmenes finitos junto con el método SIMPLE, que permite el acoplamiento de las velocidades y presiones, para resolver las ecuaciones que gobiernan los dominios fluidos. Para este desarrollo, se empleará un mallado estructurado que permitirá discretizar las ecuaciones NSE en los volúmenes de control definidos, para posteriormente, crear sistemas de ecuaciones algebraicos que resolver computacionalmente.

Sin embargo, antes de poder realizar esta discretización, y definir los nodos donde se establecerán las variables del dominio fluido, queda por abordar un problema que surge precisamente por desacoplar las velocidades y presiones. Es lo que se conoce como problema de oscilación de presiones en tablero de ajedrez, o en inglés, *Checkerboard pressure oscillation*.

Los métodos de discretización en volúmenes finitos, empiezan por dividir mediante un mallado el dominio fluido en un conjunto de volúmenes discretos. Las variables de flujo como las velocidades, presiones, densidades, temperaturas, se vinculan bien en el centro de estos volúmenes o en sus vértices, y las ecuaciones del problema se expresan en términos de relaciones algebraicas que conectan estos puntos discretos entre sí. Es intuitivo asociar todas las variables de flujo en los mismos nodos en todo el dominio computacional. Este tipo de mallados como el de la figura 1.5 se conocen como mallados co-ubicados. Sin embargo, como apunta Charlesworth [5], aquí es donde surge el problema. Si las velocidades y las presiones se calculan en las mismas ubicaciones, se puede dar el caso de que un campo de presiones no uniforme pueda actuar como si de un campo uniforme se tratase en las ecuaciones de momento discretizadas.

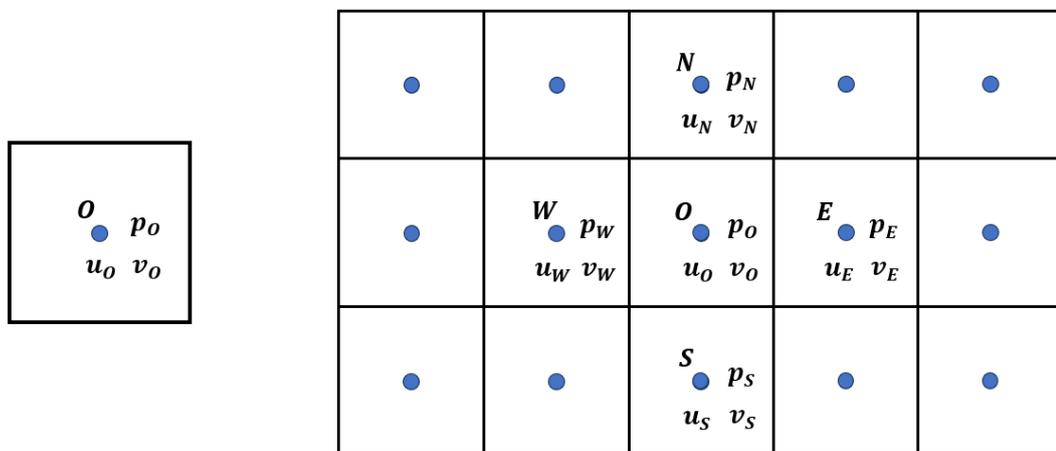


Figura 1.5 Mallado estructurado co-ubicado.

El origen de este problema está en que, para resolver la ecuación de continuidad, es necesario interpolar las velocidades en las caras de los volúmenes y es esa interpolación la que provoca las oscilaciones de presión en forma de tablero de ajedrez. Para ilustrar este concepto, se empleará el mismo ejemplo que usa Mazumder [13]: un flujo unidimensional, de densidad constante con una velocidad de entrada igual a 1.

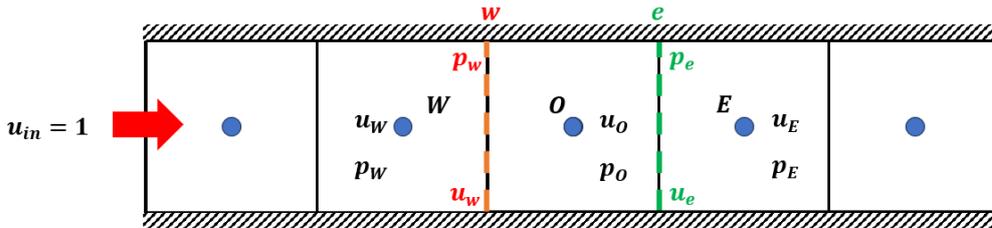


Figura 1.6 Ejemplo de problema 1D.

En este caso, la ecuación de continuidad para la celda O quedaría expresada como 1.5, donde se expresa que el flujo que entra por la cara w debe ser igual al flujo que sale por la cara e .

$$(\rho u_e - \rho u_w) \Delta y = 0 \tag{1.3}$$

Pero las velocidades en las caras u_w y u_e , son desconocidas, puesto que las velocidades están discretizadas en los nodos centrales. Si se aproximan de manera lineal según las expresiones:

$$u_w = \frac{u_O + u_W}{2} \quad u_e = \frac{u_O + u_E}{2}$$

si se sustituye en la expresión de continuidad y simplificamos, quedaría como:

$$u_E = u_W \tag{1.4}$$

Como se observa, el propio valor de la velocidad en O, no aparece en la ecuación, esto indica que matemáticamente, cualquier combinación en forma de tablero de ajedrez como la de la figura 1.7 sería una posible solución de la continuidad. Y dado que la continuidad es la que está asociada a la corrección de las presiones mediante el método SIMPLE, lo que se obtiene en realidad son oscilaciones de presiones para intentar converger a las infinitas soluciones posibles. Por lo tanto, el algoritmo no llegaría a converger y de hacerlo, no se puede garantizar que sea una solución con verdadero sentido físico.

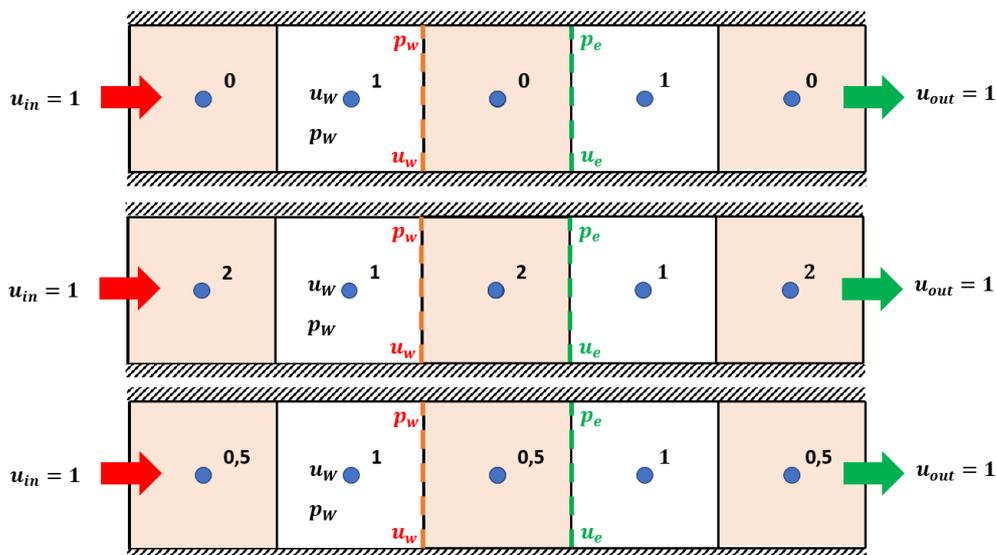


Figura 1.7 Ilustración del *Checkerboard pressure oscillation*.

Una de las primeras soluciones para remediar este problema de la interpolación debido al desacoplamiento, fue sugerida por Harlow y Welch, fue el uso de un mallado escalonado. El concepto era sencillo, se trataba de vincular las distintas variables de velocidad, presión y escalares en puntos distintos del dominio fluido, tal como muestra la figura 1.8.

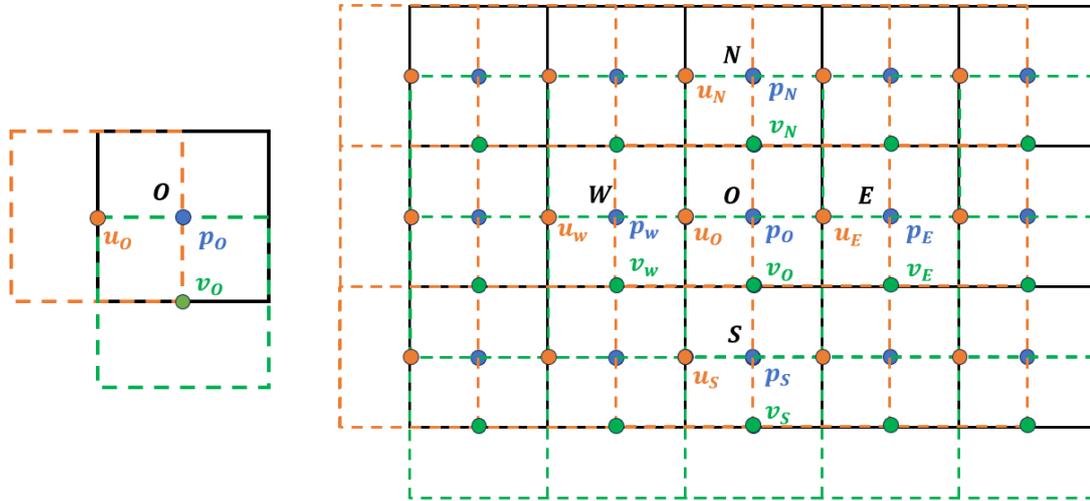


Figura 1.8 Mallado estructurado escalonado.

Al escalar estos nodos, se provoca que los volúmenes finitos empleados para discretizar las ecuaciones de cantidad de movimiento y continuidad, sean diferentes. Por lo tanto, si de nuevo se evalúa la continuidad en el volumen O, la expresión resultante es:

$$(\rho u_e - \rho u_w)\Delta y = 0 \quad (1.5)$$

$$u_w = u_O \quad u_e = u_E$$

sustituyendo y simplificando:

$$u_O = u_E \quad (1.6)$$

donde ahora sí se tiene en cuenta el valor de la velocidad de la propia celda O y, se elimina la necesidad de interpolación de las velocidades en las caras, consiguiendo una vinculación de continuidad real entre ambos volúmenes adyacentes.

Es cierto que el mallado escalonado, consigue solucionar el problema del acoplamiento de presión y velocidad, pero también introduce una mayor complejidad en el procedimiento de discretización. Deja de ser intuitivo en cuanto a la interpolación de las variables y por lo tanto complica las expresiones y el algoritmo computacional resultante. Además, si se quisiera aplicar a mallas no estructuradas, sería aún más complejo.

A raíz de esas complicaciones, surgió un nuevo enfoque para tratar de mantener todas las variables en el mismo nodo, y por tanto seguir usando los mallados co-ubicados. Rhie y Chow [13] desarrollaron en 1983 el método PWIM (Pressure-Weight Interpolation Method). Se trata de un nuevo método de interpolación para obtener las velocidades en las caras de los volúmenes, pero que tiene en cuenta las presiones del resto de volúmenes adyacentes, por lo tanto, deja de ser una ponderación lineal que es la que provoca el problema de oscilaciones de presiones en tablero de

ajedrez. Los detalles de este método serán desarrollados durante el capítulo 2 y se entenderá mejor por qué se tiene en cuenta la presión para la ponderación en la interpolación.

En conclusión, PWIM es la última pieza que se necesita para comenzar a discretizar las ecuaciones NSE en una malla estructurada co-ubicada, y desarrollar el algoritmo SIMPLE para la resolución de dominios fluidos en CFD, alcanzando la convergencia hacia la solución.

2 Formulación matemática

En este capítulo, se presenta la base teórica y matemática sobre la cual se fundamentará el desarrollo computacional del proyecto. Se inicia con la ecuación general de advección-difusión, la cual constituye un modelo fundamental para la descripción de procesos de transporte en fluidos de cualquier magnitud que se quiera estudiar. Partiendo de esta ecuación, se realizará su discretización para un volumen finito, lo que permitirá observar las claves necesarias para su implementación en un marco computacional.

Posteriormente, se realizará una particularización de esta formulación general para el caso específico de las ecuaciones de Navier-Stokes, que describen el comportamiento de fluidos viscosos e incompresibles. Estas ecuaciones, al ser de suma importancia en la mecánica de fluidos, servirán como base para el desarrollo del método de resolución computacional de dominios fluidos que se utilizará en el proyecto para los diferentes casos que se han propuesto.

A continuación, se explicará en detalle el método SIMPLE (*Semi-Implicit Method for Pressure Linked Equations*), el cual es ampliamente utilizado para la resolución numérica de las ecuaciones de Navier-Stokes en problemas de flujo laminar e incompresible. Se expondrán los fundamentos teóricos de este método, así como cada uno de los pasos necesarios para obtener expresiones matemáticas que puedan ser implementadas en un código de programación.

Finalmente, se abordará la revisión de las expresiones obtenidas para las diferentes condiciones de contorno que son comunes en problemas prácticos de flujo de fluidos. Se considerarán condiciones como paredes sin deslizamiento, velocidades y presiones de entrada y salida, con el objetivo de asegurar una adecuada modelización de los fenómenos físicos involucrados en el problema.

Por lo tanto, este apartado es de fundamental interés pues es donde se sientan las bases teóricas y matemáticas necesarias para el desarrollo y la implementación computacional del modelo fluidodinámico propuesto en el proyecto.

2.1 Ecuación general de Advección-Difusión

La ecuación de advección-difusión (2.1) describe la evolución en el tiempo y el espacio de una magnitud física ϕ en un medio continuo. En su forma más general, se compone de cuatro términos principales que representan diferentes fenómenos físicos presentes en el sistema:

$$\frac{\partial}{\partial t}(\rho\phi) + \nabla \cdot (\rho\mathbf{u}\phi - \Gamma\nabla\phi) = S_\phi \quad (2.1)$$

Término de almacenamiento $\frac{\partial}{\partial t}(\rho\phi)$

Este término describe la tasa de cambio en el tiempo de la magnitud ϕ en un volumen de control debido a procesos de almacenamiento o acumulación. Puede surgir de fenómenos como la generación o destrucción de la magnitud ϕ en el sistema. También es conocido como el término transitorio, es decir, su presencia es destacable durante fases de cambio en el sistema. Una vez que el sistema alcanza un estado estacionario, este término será nulo .

Término de flujo de advección $(\rho\mathbf{u} \cdot \nabla\phi)$

El flujo de advección representa el transporte de la magnitud ϕ debido al movimiento del medio. Se calcula como el producto del vector de velocidad del fluido \mathbf{u} y el gradiente espacial de ϕ . Este término modela el transporte convectivo de ϕ a través del dominio.

Término de flujo difusivo $\nabla \cdot (\Gamma\nabla\phi)$

El término de flujo difusivo modela la difusión molecular de la magnitud ϕ en el medio. Se calcula como el producto de la difusividad Γ y el laplaciano de ϕ . Este término describe cómo la magnitud ϕ se disemina desde regiones de mayor concentración hacia regiones de menor concentración debido a gradientes de concentración.

Término fuente S_ϕ

El término fuente representa cualquier fuente o sumidero de la magnitud ϕ en el dominio. Puede surgir de procesos físicos como reacciones químicas, generación de calor, o cualquier otro proceso que añada o elimine ϕ en el sistema.

Revisando la ecuación general (2.1), se puede escribir de forma más compacta si se combinan el término advectivo y difusivo, como un único término de flujo J que entra o sale de la siguiente forma:

$$J = J_A + J_D = \rho\mathbf{u}\phi - \Gamma\nabla\phi \quad (2.2)$$

Así, la ecuación general compacta (2.3), solo presenta tres términos:

$$\frac{\partial}{\partial t}(\rho\phi) + \nabla \cdot J = S_\phi \quad (2.3)$$

- Término de almacenamiento $\frac{\partial}{\partial t}(\rho\phi)$
- Término de flujos $\nabla \cdot J$
- Término fuente S_ϕ

2.1.1 Discretización de la ecuación general de advección-difusión para un volumen finito

Avanzando hacia una formulación apta para la implementación en código, el siguiente paso implica la integración de esta ecuación (2.3) dentro de un volumen de control finito. Este proceso implica un análisis minucioso de cada término de manera individual, desentrañando su contribución al sistema en su totalidad, hasta alcanzar la expresión completa que refleje de manera precisa el comportamiento de la magnitud ϕ en el contexto del problema abordado

Término de almacenamiento

Comenzando por el término transitorio de la expresión, como ya se comentó en apartados anteriores, se usará un mallado co-ubicado, por lo tanto, la evaluación de la expresión debe hacerse en el centro O del volumen finito de control V_O . Integrada la expresión en el volumen, ahora sí, puede ser discretizada la derivada temporal en serie de Taylor.

$$\int_{V_O} \frac{\partial}{\partial t}(\rho\phi)dV = \left[\frac{\partial}{\partial t}(\rho\phi) \right]_O V_O \approx \left[\frac{(\rho\phi)_O^{n+1} - (\rho\phi)_O^n}{\Delta t} \right] V_O \quad (2.4)$$

Sin embargo, este término resultará insignificante para esta revisión, dado que los dominios que se analizarán estarán en régimen estacionario. En esta condición, las variables del fluido ya no estarán sujetas al efecto transitorio, lo que hace que este término no tenga impacto en el estudio.

Término de flujos

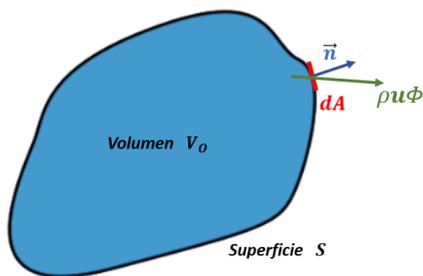
El siguiente aspecto a analizar son los flujos, recordando que estos están compuestos por dos componentes distintos: el flujo advectivo, que refleja los efectos de la advección, y el flujo difusivo, que representa los efectos de la difusión.

$$\int_{V_O} \nabla \cdot J dV = \int_{V_O} \nabla \cdot (\rho\mathbf{u}\phi - \Gamma\nabla\phi) dV \quad (2.5)$$

Es apropiado examinar este término una vez más, descomponiendo cada uno de los flujos y adaptándolos al caso particular de este estudio. En este contexto, el volumen de control se define como una celda cuadrada, lo que nos facilita estudiar los flujos que entran y salen a través de las superficies de dicha celda debido a los efectos advectivos y difusivos mencionados anteriormente.

Flujo Advectivo J_A

Revisando la expresión de la integral de volumen, al aplicar el Teorema de Gauss, se logra transformarla en una integral de superficie. Esto tiene sentido físico, ya que se busca evaluar el flujo de la magnitud ϕ que entra o sale a través de la superficie S



$$J_A = \int_{V_O} \nabla \cdot (\rho\mathbf{u}\phi) dV = \int_S \rho\mathbf{u}\phi \cdot \vec{n} dA \quad (2.6)$$

Figura 2.1 Teorema de Gauss.

Si se extiende este razonamiento a la superficie de una celda cuadrada típica utilizada en la dinámica de fluidos computacional (CFD), es necesario proyectar esta integral en las superficies Oeste (w), Este (e), Sur (s) y Norte (n).

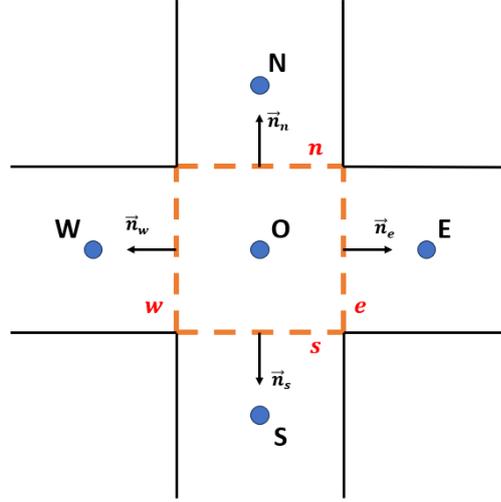


Figura 2.2 Caras y normales de la celda de referencia.

$$J_A = \int_S \rho \mathbf{u} \phi \cdot \vec{n} dA = \rho_e (\mathbf{u} \cdot \vec{n})_e A_e \phi_e + \rho_w (\mathbf{u} \cdot \vec{n})_w A_w \phi_w + \rho_n (\mathbf{u} \cdot \vec{n})_n A_n \phi_n + \rho_s (\mathbf{u} \cdot \vec{n})_s A_s \phi_s \quad (2.7)$$

Si se toma, por ejemplo, la superficie e y se evalúa el valor del vector velocidad \mathbf{u} y la normal a la superficie \vec{n}_e , el flujo atravesado en dicha cara es:

$$\mathbf{u}_e = u_e \cdot \vec{i} + v_e \cdot \vec{j} \quad (2.8)$$

$$\vec{n}_e = \vec{i} \quad (2.9)$$

$$(\mathbf{u} \cdot \vec{n})_e = u_e \quad (2.10)$$

$$\rho_e (\mathbf{u} \cdot \vec{n})_e A_e \phi_e = \rho_e u_e A_e \phi_e \quad (2.11)$$

Repetiendo este proceso para las caras restantes w , s y n , se obtiene la expresión completa del flujo advectivo. No obstante, analizando los términos obtenidos, se observa que se pueden simplificar, ya que realmente esta expresión lo que representa es el gasto \dot{m} de cada una de las caras de las celdas por el valor de la magnitud ϕ en cada una de las caras.

$$\begin{aligned} J_A &= \rho_e u_e A_e \phi_e - \rho_w u_w A_w \phi_w + \rho_n v_n A_n \phi_n - \rho_s v_s A_s \phi_s = \\ &= (\rho u)_e \Delta y \phi_e - (\rho u)_w \Delta y \phi_w + (\rho u)_n \Delta x \phi_n - (\rho u)_s \Delta x \phi_s = \\ &= \dot{m}_e \phi_e - \dot{m}_w \phi_w + \dot{m}_n \phi_n - \dot{m}_s \phi_s \end{aligned} \quad (2.12)$$

Sin embargo, en el contexto del campo fluido que se ha planteado, los valores de ϕ se conocen en el centro de las celdas O y no en las caras. Por lo tanto, se debe idear una aproximación para los valores en las caras. Una aproximación común es utilizar la diferencia central mediante una aproximación de Taylor de segundo orden de las celdas adyacentes. Considerando un ejemplo concreto para ϕ_e , esta se puede aproximar mediante ϕ_O y ϕ_E de la siguiente manera:

$$\phi_E = \phi_e + \frac{\Delta x}{2} \frac{d\phi}{dx} \Big|_e + \frac{1}{2} \left(\frac{\Delta x}{2} \right)^2 \frac{d^2\phi}{dx^2} \Big|_e + \frac{1}{6} \left(\frac{\Delta x}{2} \right)^3 \frac{d^3\phi}{dx^3} \Big|_e \quad (2.13)$$

$$\phi_O = \phi_e - \frac{\Delta x}{2} \frac{d\phi}{dx} \Big|_e + \frac{1}{2} \left(\frac{\Delta x}{2} \right)^2 \frac{d^2\phi}{dx^2} \Big|_e - \frac{1}{6} \left(\frac{\Delta x}{2} \right)^3 \frac{d^3\phi}{dx^3} \Big|_e \quad (2.14)$$

$$\phi_e = \frac{\phi_E + \phi_O}{2} - \frac{1}{2} \left(\frac{\Delta x}{2} \right)^2 \frac{d^2\phi}{dx^2} \Big|_e \quad (2.15)$$

Una vez obtenida la expresión para ϕ_e (2.15) como una diferencia central de segundo orden en las expansiones de Taylor, surge la pregunta: ¿es esta expresión válida? Sus ventajas son dos:

- Es una expresión de segundo orden muy intuitiva para mallas uniformes,
- Proporciona información tanto aguas arriba como aguas abajo del punto de interés.

Sin embargo, esta última característica no es típica de estos procesos, ya que la advección sigue la dirección del flujo. En conclusión, esta expresión no representa adecuadamente la física del fenómeno de advección. Para corregir esto, se debe recurrir a una expresión más simple de primer orden, que tenga en cuenta la información aguas arriba, la cual se transmitirá aguas abajo. Ahora bien, es crucial determinar la dirección del flujo en cada caso. Esto se logra mediante la expresión (2.16) que indica que si la velocidad es positiva, el flujo proviene de la izquierda, mientras que si la velocidad es negativa, el flujo proviene de la derecha.

$$\phi_e = \begin{cases} \phi_O & \text{if } (\rho u)_{e,O} > 0 \\ \phi_E & \text{if } (\rho u)_{e,O} < 0 \end{cases} \quad (2.16)$$

Para simplificar y obtener un resultado más compacto, Mazumder [13] recomienda representarlo como una función de conmutación (2.17). Esto permite evitar la necesidad de implementar bucles condicionales en la programación de las expresiones.

$$(\rho u)_e \phi_e = \frac{|(\rho u)_e| + (\rho u)_e}{2} \phi_O - \frac{|(\rho u)_e| - (\rho u)_e}{2} \phi_E \quad (2.17)$$

Simplemente se debe repetir este proceso para las caras restantes: oeste, norte y sur. No obstante, es crucial tener en cuenta que la orientación de estas caras determinará el signo positivo o negativo en la expresión, dependiendo de la dirección del flujo del fluido. Por lo tanto, esta orientación debe reflejarse correctamente en las expresiones equivalentes resultantes.

Sintetizando los conceptos previamente discutidos, la expresión final discretizada del flujo advectivo es la siguiente:

$$\begin{aligned}
J_A = & \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} \phi_O - \frac{|(\rho u)_e| - (\rho u)_e}{2} \phi_E \right] \Delta y - \\
& - \left[\frac{|(\rho u)_w| + (\rho u)_w}{2} \phi_W - \frac{|(\rho u)_w| - (\rho u)_w}{2} \phi_O \right] \Delta y + \\
& + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} \phi_O - \frac{|(\rho v)_n| - (\rho v)_n}{2} \phi_N \right] \Delta x - \\
& - \left[\frac{|(\rho v)_s| + (\rho v)_s}{2} \phi_S - \frac{|(\rho v)_s| - (\rho v)_s}{2} \phi_O \right] \Delta x
\end{aligned} \tag{2.18}$$

Flujo Difusivo J_D

Al abordar el término difusivo, se debe seguir el mismo procedimiento de discretización que para el flujo advectivo, con una excepción crucial: en este caso, el flujo difusivo afecta tanto aguas arriba como aguas abajo de la corriente, y no solo en la dirección del flujo. Por lo tanto, en este caso, es apropiado utilizar la diferencia central para la aproximación de la magnitud ϕ_e .

De manera similar, al aplicar el teorema de Gauss, la integral de volumen se transforma en una integral de superficie que se proyecta sobre las caras de la celda cuadrada típica en la dinámica de fluidos computacional (CFD).

$$\begin{aligned}
J_D &= - \int_{V_O} \nabla \cdot (\Gamma \nabla \phi) dV = - \int_S (\Gamma \nabla \phi) \cdot \vec{n} dA = \\
&= - [\Gamma_e (\nabla \phi \cdot \vec{n})_e A_e + \Gamma_w (\nabla \phi \cdot \vec{n})_w A_w + \Gamma_n (\nabla \phi \cdot \vec{n})_n A_n + \Gamma_s (\nabla \phi \cdot \vec{n})_s A_s]
\end{aligned} \tag{2.19}$$

Tomando nuevamente como ejemplo la cara Este (e) de la celda, simplemente basta con proyectar la divergencia de ϕ a lo largo de su vector normal n_e en la cara.

$$\nabla \phi = \frac{\partial \phi}{\partial x} \cdot \vec{i} + \frac{\partial \phi}{\partial y} \cdot \vec{j} \tag{2.20}$$

$$\vec{n}_e = \vec{i} \tag{2.21}$$

$$J_{D,e} = -\Gamma_e (\nabla \phi \cdot \vec{n})_e A_e = -\Gamma_e \left. \frac{\partial \phi}{\partial x} \right|_e \Delta y \tag{2.22}$$

Finalmente, como se mencionó anteriormente, esta derivada parcial puede aproximarse mediante la diferencia central, lo que proporciona la expresión.

$$J_{D,e} = -\Gamma_e \left. \frac{\partial \phi}{\partial x} \right|_e \Delta y \approx -\Gamma_e \left(\frac{\phi_E - \phi_O}{\Delta x} \right)_e \Delta y \tag{2.23}$$

Repitiendo este proceso para las demás caras de la celda, se logra la expresión final para el flujo difusivo:

$$J_D = - \left[\Gamma_e \left(\frac{\phi_E - \phi_O}{\Delta x} \right)_e \Delta y + \Gamma_w \left(\frac{\phi_O - \phi_W}{\Delta x} \right)_w \Delta y + \Gamma_n \left(\frac{\phi_N - \phi_O}{\Delta y} \right)_n \Delta x + \Gamma_s \left(\frac{\phi_O - \phi_S}{\Delta y} \right)_s \Delta x \right] \quad (2.24)$$

Término fuente

En cuanto al término fuente en la ecuación general de advección-difusión, su discretización no puede abordarse hasta que se reemplace por las fuentes o sumideros que afectan a la magnitud ϕ . Es decir, la discretización de este término depende de las condiciones específicas del problema que se esté resolviendo, por lo que se dejará esta tarea para secciones posteriores. Por ahora, solo se puede dejar indicado que este término deberá integrarse en el volumen de la celda cuadrada previamente definida para los flujos mencionados anteriormente.

$$S_F = \int_{V_O} S_\phi dV = \int_s^n \int_w^e S_\phi dx dy \quad (2.25)$$

Para concluir este apartado, donde se ha trabajado en la obtención de una expresión discretizada de la ecuación general de advección-difusión, es importante destacar que, dado que se está analizando un régimen estacionario, la expresión general discretizada se obtiene al combinar las ecuaciones (2.18), (2.24) y (2.25), resultando en lo siguiente:

$$\begin{aligned} & \left[\frac{|\rho u|_e + (\rho u)_e}{2} \phi_O - \frac{|\rho u|_e - (\rho u)_e}{2} \phi_E - \Gamma_e \left(\frac{\phi_E - \phi_O}{\Delta x} \right) \right] \Delta y - \\ & - \left[\frac{|\rho u|_w + (\rho u)_w}{2} \phi_W - \frac{|\rho u|_w - (\rho u)_w}{2} \phi_O - \Gamma_w \left(\frac{\phi_O - \phi_W}{\Delta x} \right) \right] \Delta y + \\ & + \left[\frac{|\rho v|_n + (\rho v)_n}{2} \phi_O - \frac{|\rho v|_n - (\rho v)_n}{2} \phi_N - \Gamma_n \left(\frac{\phi_N - \phi_O}{\Delta y} \right) \right] \Delta x - \\ & - \left[\frac{|\rho v|_s + (\rho v)_s}{2} \phi_S - \frac{|\rho v|_s - (\rho v)_s}{2} \phi_O - \Gamma_s \left(\frac{\phi_O - \phi_S}{\Delta y} \right) \right] \Delta x = \int_s^n \int_w^e S_\phi dx dy \end{aligned} \quad (2.26)$$

2.2 Ecuaciones de Navier-Stokes para flujos incompresibles

Las ecuaciones de Navier-Stokes, fundamentales en la mecánica de fluidos, se derivan como un caso particular de la ecuación general de advección-difusión. Estas ecuaciones describen el comportamiento de un flujo de fluido en un dominio dado. En particular, como muestra Mazumder [13], si se realizan las siguientes sustituciones en la ecuación general de advección-difusión (2.1), se consigue obtener las ecuaciones de continuidad y cantidad de movimiento de las ecuaciones de Navier-Stokes:

$$\text{Ec. Continuidad } \phi = 1 ; \Gamma = 0 ; S_\phi = 0 \quad (2.27)$$

$$\frac{\partial \rho}{\partial t} + \nabla(\rho \vec{u}) = 0 \quad (2.28)$$

$$\text{Ec. Cant. Movimiento } \phi = \vec{u} ; \Gamma = \mu ; S_\phi = +\nabla p + \rho \vec{f}_m \quad (2.29)$$

$$\frac{\partial(\rho \vec{u})}{\partial t} + \nabla \cdot (\rho \vec{u} \vec{u} - \mu \nabla \vec{u}) = -\nabla p + \rho \vec{f}_m \quad (2.30)$$

A medida que se analiza su formulación, es crucial considerar las simplificaciones y suposiciones necesarias que permiten reducir la complejidad del problema y llegar a una descripción más manejable y aplicable a problemas específicos como los que aplican a esta revisión.

En primer lugar, se considerará un flujo bidimensional isotérmico, lo que implica que las propiedades del fluido, como la densidad y la viscosidad, no varían en el tiempo ni en el espacio. Además, es lógico asumir que la gravedad es despreciable en comparación con otros efectos, lo que permite centrarse únicamente en las fuerzas inerciales y viscosas del fluido.

Otra hipótesis fundamental típica es que el flujo se encuentra en régimen estacionario, lo que significa que las propiedades del fluido en cualquier punto del dominio no cambian con el tiempo, es decir, se puede despreciar el término transitorio de ambas ecuaciones. Si a esto se le agrega, que solo se van a estudiar flujos incompresibles, donde la densidad del fluido permanece constante en todo el dominio, implica que por lo tanto, las variaciones de velocidad se producen únicamente por cambios en la dirección y magnitud del flujo.

Bajo estas simplificaciones, las ecuaciones de Navier-Stokes se reducen a tres ecuaciones principales: una de continuidad (2.31) y dos de cantidad de movimiento (2.32 y 2.33).

$$\nabla(\rho \vec{u}) = 0 \quad (2.31)$$

$$\nabla \cdot (\rho \vec{u} u - \mu \nabla u) = -\frac{\partial p}{\partial x} \quad (2.32)$$

$$\nabla \cdot (\rho \vec{u} v - \mu \nabla v) = -\frac{\partial p}{\partial y} \quad (2.33)$$

La ecuación de continuidad representa la conservación de la masa y establece que la divergencia del campo de velocidad es cero. Por otro lado, las ecuaciones de cantidad de movimiento, derivadas de la segunda ley de Newton para fluidos, relacionan las variaciones de velocidad con las fuerzas que actúan sobre el fluido, como la presión y la viscosidad.

Estas ecuaciones forman un sistema acoplado de tres incógnitas: las dos componentes de velocidad del fluido ($\vec{u} = u \vec{i} + v \vec{j}$) y la presión p . Resolver este sistema permite determinar el campo de velocidades y la distribución de presiones en el dominio, lo que es crucial para comprender y predecir el comportamiento del flujo en cualquier dominio fluido de interés para aplicaciones en ingeniería.

Al revisar este sistema de ecuaciones, se observa que en las ecuaciones de cantidad de movimiento, las tres incógnitas u , v y p aparecen de manera explícita. Sin embargo, en la ecuación de continuidad,

solo se hace referencia a la velocidad. Más adelante, en este estudio, se explicará cómo manipular la ecuación de continuidad para introducir la presión p , lo que permitirá resolver el sistema de ecuaciones utilizando el algoritmo SIMPLE.

El próximo paso es discretizar estas ecuaciones en un volumen finito, utilizando las expresiones deducidas previamente de la ecuación general de advección-difusión.

2.2.1 Discretización y linealización de la ecuación de cantidad de movimiento para un volumen finito

En primer lugar, se debe abordar la discretización de las ecuaciones de cantidad de movimiento en un volumen finito. Para ello, basta con tomar la expresión general discretizada previamente obtenida (2.26) y sustituir los mismos parámetros que se han utilizado anteriormente en (2.32 y 2.33) para las dos dimensiones. Esto permite expresar las ecuaciones de cantidad de movimiento en términos discretos (2.34 y 2.35). Donde como se puede observar, en el sistema de ecuaciones, siguen apareciendo las incógnitas \hat{u} , \hat{v} y p .

$$\begin{aligned}
 & \left[\frac{|\rho u|_e + (\rho u)_e}{2} \hat{u}_O - \frac{|\rho u|_e - (\rho u)_e}{2} \hat{u}_E - \mu_e \frac{\hat{u}_E - \hat{u}_O}{\Delta x} \right] \Delta y - \\
 & - \left[\frac{|\rho u|_w + (\rho u)_w}{2} \hat{u}_W - \frac{|\rho u|_w - (\rho u)_w}{2} \hat{u}_O - \mu_w \frac{\hat{u}_O - \hat{u}_W}{\Delta x} \right] \Delta y + \\
 & + \left[\frac{|\rho v|_n + (\rho v)_n}{2} \hat{u}_O - \frac{|\rho v|_n - (\rho v)_n}{2} \hat{u}_N - \mu_n \frac{\hat{u}_N - \hat{u}_O}{\Delta y} \right] \Delta x - \\
 & - \left[\frac{|\rho v|_s + (\rho v)_s}{2} \hat{u}_S - \frac{|\rho v|_s - (\rho v)_s}{2} \hat{u}_O - \mu_s \frac{\hat{u}_O - \hat{u}_S}{\Delta y} \right] \Delta x = - \int_s^n \int_w^e \frac{\partial p}{\partial x} dx dy
 \end{aligned} \tag{2.34}$$

$$\begin{aligned}
 & \left[\frac{|\rho u|_e + (\rho u)_e}{2} \hat{v}_O - \frac{|\rho u|_e - (\rho u)_e}{2} \hat{v}_E - \mu_e \frac{\hat{v}_E - \hat{v}_O}{\Delta x} \right] \Delta y - \\
 & - \left[\frac{|\rho u|_w + (\rho u)_w}{2} \hat{v}_W - \frac{|\rho u|_w - (\rho u)_w}{2} \hat{v}_O - \mu_w \frac{\hat{v}_O - \hat{v}_W}{\Delta x} \right] \Delta y + \\
 & + \left[\frac{|\rho v|_n + (\rho v)_n}{2} \hat{v}_O - \frac{|\rho v|_n - (\rho v)_n}{2} \hat{v}_N - \mu_n \frac{\hat{v}_N - \hat{v}_O}{\Delta y} \right] \Delta x - \\
 & - \left[\frac{|\rho v|_s + (\rho v)_s}{2} \hat{v}_S - \frac{|\rho v|_s - (\rho v)_s}{2} \hat{v}_O - \mu_s \frac{\hat{v}_O - \hat{v}_S}{\Delta y} \right] \Delta x = - \int_s^n \int_w^e \frac{\partial p}{\partial y} dx dy
 \end{aligned} \tag{2.35}$$

Si se reorganizan los términos de la ecuación agrupando todos los términos que multiplican a cada una de las incógnitas, se consigue obtener los coeficientes de enlace, y una expresión mucho más compacta. Estos coeficientes A_i^X y A_i^Y son fundamentales en la resolución de las ecuaciones discretizadas, ya que representan las contribuciones de los nodos vecinos al nodo central, es decir, determinan la forma en que las variables están relacionadas entre sí.

Ahora sí, como se indicó anteriormente, es posible integrar el término fuente de estas expresiones. En el caso de las ecuaciones de Navier-Stokes y con las simplificaciones que se han realizado, este término fuente se reduce al gradiente de presiones, que es responsable de impulsar el flujo del fluido en cada dirección y por lo tanto contribuye a cada una de las velocidades.

Tomando de nuevo el recinto de la celda de control e integrando:

$$S_O^X = - \int_s^n \int_w^e \frac{\partial p}{\partial x} dx dy = - \int_s^n (p_e - p_w) dy = (p_w - p_e) \Delta y \quad (2.36)$$

$$S_O^Y = - \int_s^n \int_w^e \frac{\partial p}{\partial y} dx dy = - \int_w^e (p_n - p_s) dx = (p_s - p_n) \Delta x \quad (2.37)$$

Dado que la presión en las caras no se conoce, sino que los valores se están calculando en el centro de cada una de las celdas, es necesario aproximar los valores de las caras mediante interpolación, utilizando una ponderación basada en la distancia de las celdas adyacentes. En principio, este método será válido siempre que las celdas estén posicionadas en el interior del dominio. Sin embargo, este será un punto importante a revisar cuando las celdas estén sujetas a condiciones de contorno, ya que esta aproximación podría dejar de ser válida. Por el momento, sin embargo, se considera válida.

$$S_O^X = (p_w - p_e) \Delta y = \left(\frac{p_O + p_W}{2} - \frac{p_E + p_O}{2} \right) \Delta y = \frac{1}{2} (p_W - p_E) \Delta y \quad (2.38)$$

$$S_O^Y = (p_s - p_n) \Delta x = \left(\frac{p_O + p_S}{2} - \frac{p_N + p_O}{2} \right) \Delta x = \frac{1}{2} (p_S - p_N) \Delta x \quad (2.39)$$

En última instancia, estas ecuaciones compactas y las expresiones de los coeficientes de enlace constituyen el núcleo implementado en el algoritmo SIMPLE para resolver las ecuaciones de cantidad de movimiento. A través de este proceso, se obtendrán los valores de velocidad horizontal \hat{u} y vertical \hat{v} .

$$A_O^X \hat{u}_O + A_W^X \hat{u}_W + A_E^X \hat{u}_E + A_N^X \hat{u}_N + A_S^X \hat{u}_S = S_O^X \quad (2.40)$$

$$A_O^X = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (2.41)$$

$$A_E^X = - \left[\frac{|(\rho u)_e| - (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} \right] \Delta y \quad (2.42)$$

$$A_W^X = - \left[\frac{|(\rho u)_w| + (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y \quad (2.43)$$

$$A_N^X = - \left[\frac{|(\rho v)_n| - (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} \right] \Delta x \quad (2.44)$$

$$A_S^X = - \left[\frac{|(\rho v)_s| + (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (2.45)$$

$$A_O^Y \hat{v}_O + A_W^Y \hat{v}_W + A_E^Y \hat{v}_E + A_N^Y \hat{v}_N + A_S^Y \hat{v}_S = S_O^Y \quad (2.46)$$

$$A_O^Y = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (2.47)$$

$$A_E^Y = - \left[\frac{|(\rho u)_e| - (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} \right] \Delta y \quad (2.48)$$

$$A_W^Y = - \left[\frac{|(\rho u)_w| + (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y \quad (2.49)$$

$$A_N^Y = - \left[\frac{|(\rho v)_n| - (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} \right] \Delta x \quad (2.50)$$

$$A_S^Y = - \left[\frac{|(\rho v)_s| + (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (2.51)$$

2.2.2 Discretización y linealización de la ecuación de continuidad para un volumen finito

Queda por abordar la discretización de la ecuación de continuidad. Al seguir el proceso discutido hasta ahora, se debe tomar la ecuación diferencial (2.28) y llevar a cabo su integración a lo largo y ancho de la celda CFD. No obstante, también se puede abordar la discretización desde el punto de vista físico de esta ecuación. La ecuación de continuidad indica que debe existir un equilibrio físico entre los flujos que entran y salen del dominio fluido, en este caso de la celda. Por lo tanto, al discretizar la ecuación de continuidad, se debe evaluar el flujo que entra o sale por cada una de las caras este E , oeste W , norte N y sur S de la celda.

$$\int_{V_o} \nabla(\rho \mathbf{u}) dV = \int_S (\rho \mathbf{u}) \cdot \vec{n} dA = 0 \quad (2.52)$$

$$\int_S (\rho \mathbf{u}) \cdot \vec{n} dA = \rho_e(\mathbf{u} \cdot \vec{n})_e A_e + \rho_w(\mathbf{u} \cdot \vec{n})_w A_w + \rho_n(\mathbf{u} \cdot \vec{n})_n A_n + \rho_s(\mathbf{u} \cdot \vec{n})_s A_s = 0 \quad (2.53)$$

$$[(\rho u)_e - (\rho u)_w] \Delta y + [(\rho v)_n - (\rho v)_s] \Delta x = 0 \quad (2.54)$$

Al revisar la expresión, Se observa que la velocidad sigue apareciendo de forma explícita, sin embargo, no se encuentra rastro alguno de la presión en la ecuación de continuidad. Este asunto será abordado más adelante en este estudio, y se procederá a modificar la expresión para incorporar explícitamente la presión.

Llegado este punto, ya se cuenta con las ecuaciones fundamentales de Navier Stokes expresadas en la forma adecuada para poder ser implementadas en cualquier código y poder ser resueltas de forma numérica.

2.3 Desarrollo del método SIMPLE

La simulación de la dinámica fluidos computacional (CFD) es una herramienta crucial en ingeniería y ciencia para comprender y predecir el comportamiento de los fluidos en una amplia gama de aplicaciones, desde el diseño de aviones y automóviles hasta la optimización de procesos industriales. En este sentido, uno de los métodos más utilizados en CFD es el método SIMPLE (Semi-Implicit Method for Pressure-Linked Equations), que proporciona una solución aproximada eficiente a las ecuaciones de Navier-Stokes para flujos incompresibles.

2.3.1 Fundamentos del método SIMPLE

Resolver directamente las ecuaciones de Navier-Stokes es computacionalmente costoso debido a la complejidad de los términos no lineales y la necesidad de discretizar el dominio de simulación. El método SIMPLE aborda estos desafíos mediante un enfoque iterativo que simplifica la resolución del problema.

Tal como indica Charlesworth en su tesis, la dificultad en resolver las ecuaciones de Navier-Stokes incompresibles radica en la falta de una ecuación explícita para el campo de presión. Este desafío se supera con el algoritmo SIMPLE, que emplea un procedimiento de aproximación y corrección para el campo de presión. Dicho de otra forma, se propone inicialmente un campo de presión y velocidades, para luego resolver las ecuaciones de momento y, posteriormente, se actualiza el campo de presión mediante un mecanismo adecuado. Este mecanismo consiste en aprovechar la ecuación de continuidad, ya que dada una presión correcta, una solución de las ecuaciones de momento también cumplirá con la ecuación de continuidad.

En el contexto del algoritmo SIMPLE, esto se traduce en formular la ecuación de continuidad como una ecuación de corrección de presión, de ahí el nombre del método, que de forma semi-implícita, consigue una ecuación de continuidad expresada en presiones. Al resolver esta ecuación, se actualizan tanto el campo de presión como las velocidades utilizadas en la solución de las ecuaciones de momento. La derivación detallada de esta ecuación de corrección de presión se mostrará más adelante en el apartado 3.3.2.5.

Durante el proceso de iteración, se garantiza la conservación de la masa y la presión en el dominio de simulación mediante la aplicación adecuada de los esquemas de corrección de velocidad y presión.

Sin embargo, puede requerir un número significativo de iteraciones para alcanzar la convergencia y la solución puede ser sensible a la elección de parámetros y condiciones iniciales.

2.3.2 Algoritmo SIMPLE

Como se puede observar en el diagrama de la figura 2.3, el método SIMPLE sigue un esquema de iteración entre la velocidad y la presión del fluido. Comienza con una suposición inicial para la velocidad y la presión del fluido en el dominio de simulación. Luego, se llevan a cabo tres pasos principales de iteración: predicción de velocidad, corrección de presión y corrección de velocidad. Estos pasos se repiten iterativamente hasta que se cumpla el criterio de convergencia seleccionado.

A continuación, se detallarán minuciosamente cada uno de los pasos del algoritmo, explorando en profundidad el desarrollo de las ecuaciones requeridas para su implementación en un código.

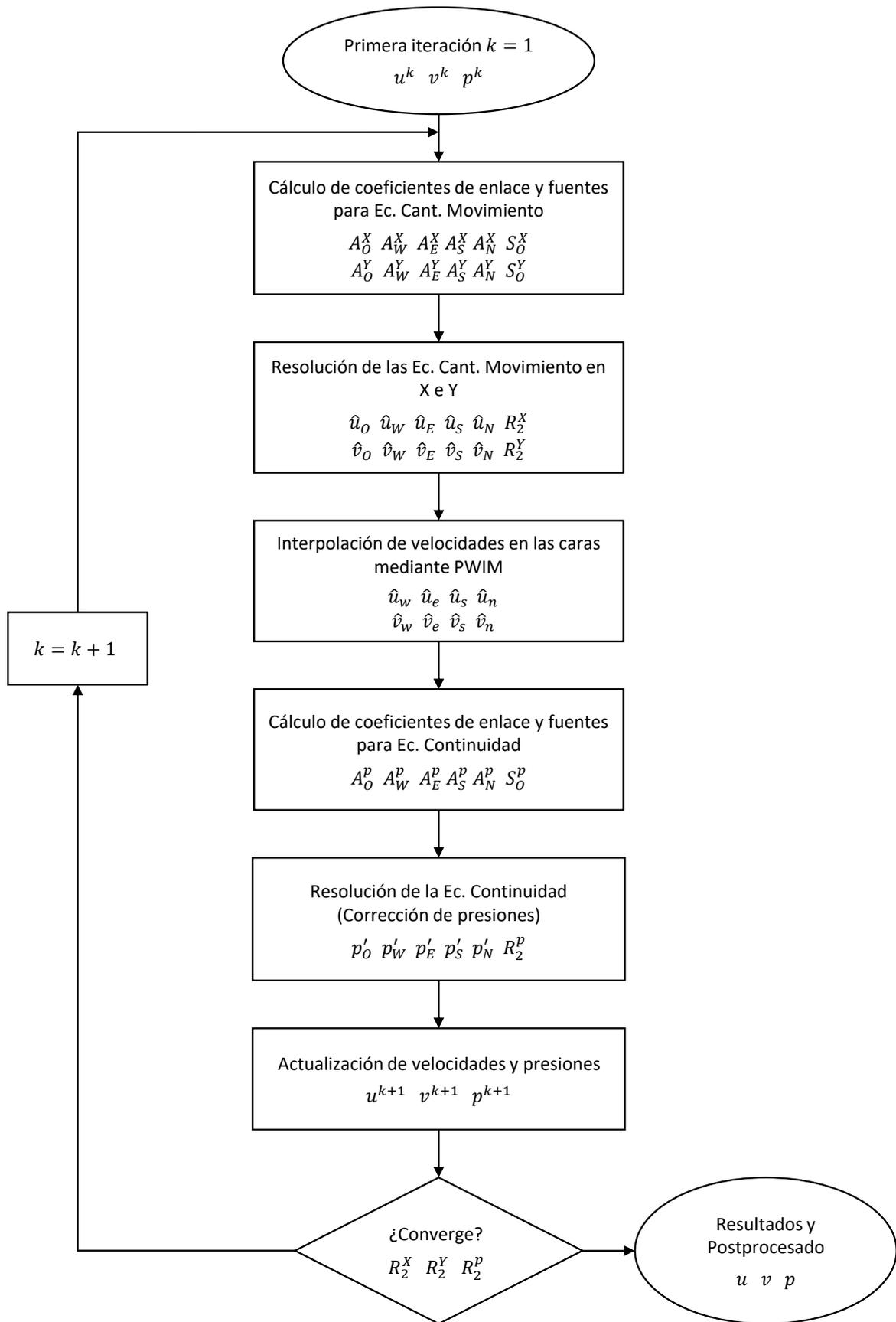


Figura 2.3 Algoritmo SIMPLE.

2.3.2.1 Primera iteración

Como se han indicado anteriormente, el algoritmo SIMPLE comienza con la suposición de las velocidades u^k y v^k y presiones p^k en el campo fluido, lo que representa una primera aproximación a la solución buscada para $k = 1$. Es importante que esta aproximación inicial tenga sentido físico y no esté demasiado alejada de la solución final, ya que una desventaja del método es su sensibilidad a las condiciones iniciales, lo que puede dificultar la convergencia hacia una solución.

En función del problema a resolver, es crucial elegir las condiciones iniciales adecuadas. Sin embargo, esta variabilidad también puede ser útil para garantizar la consistencia de la solución. Si diferentes condiciones iniciales conducen a la misma solución del campo fluido, es probable que esta sea una solución sólida y confiable.

Por ejemplo, en los casos tratados más adelante, se han considerado diferentes condiciones iniciales. Para un canal longitudinal y para el problema de cambio de sección, se propuso una velocidad uniforme como condición inicial, simulando un flujo uniforme en todo el canal, permitiendo que los efectos viscosos adapten gradualmente el campo fluido. Por otro lado, para el problema de cavitación, se optó por una velocidad inicial nula, permitiendo que la condición de contorno del movimiento de la cubierta superior y los efectos de advección y difusión moldeen el campo fluido con cada iteración.

2.3.2.2 Cálculo de los coeficientes y fuentes para las ecuaciones de cantidad de movimiento

Siguiendo el diagrama, el siguiente paso implica el cálculo de los coeficientes de enlace A_i^X y A_i^Y , y las fuentes S_O^X y S_O^Y , necesarios para resolver las ecuaciones de cantidad de movimiento en los ejes X e Y . Es importante recordar que los coeficientes de enlace, derivados en secciones anteriores y expresados desde 2.41 hasta 2.45 para el eje X y 2.47 hasta 2.51 para el eje Y , representan las relaciones entre celdas adyacentes.

Dado que estas expresiones se dedujeron a través de la discretización de la ecuación de cantidad de movimiento, las expresiones son válidas para celdas interiores. En principio, para celdas interiores, los coeficientes de enlace en los ejes X e Y tienen expresiones equivalentes. Respecto a las fuentes, solo tienen en cuenta los gradientes de presión entre las celdas, lo que impulsa al fluido a través de ellas y cuyas expresiones se encuentran en 2.38 y 2.39.

En este punto, es crucial considerar las condiciones de contorno para una representación precisa del dominio fluido, pues no todas las celdas serán interiores. En el proceso de implementación del código, es fundamental tener en cuenta estas condiciones, dado que diferentes áreas del dominio estarán sujetas a distintas condiciones. Por consiguiente, es imperativo reflejar esta variabilidad en el código correspondiente para asegurar una representación precisa del sistema físico en estudio. En este caso, mediante diferentes bucles que recorran las distintas zonas según sus condiciones de contorno, por ejemplo recorriendo las celdas por filas o por columnas.

Esta consideración implica la necesidad de tener en cuenta las condiciones de contorno que afectan tanto a las expresiones de los coeficientes como a las fuentes. Las adaptaciones específicas de las expresiones conforme a las condiciones de contorno serán abordadas en secciones posteriores.

Un detalle a tener en cuenta en este apartado, específicamente durante la primera iteración, es la necesidad de interpolar la velocidad en las caras. Para este propósito, una aproximación inicial válida podría ser una interpolación ponderada por distancia, similar a la expresión (2.55). Es importante señalar que esto podría plantear interrogantes, ya que como se explicó previamente, esto puede dar origen al problema conocido como oscilaciones de presiones en tablero de ajedrez (checkboard

pressure oscillation). Sin embargo, como se discutirá más adelante, este escenario está contemplado dentro del método de interpolación PWIM para la aproximación de las velocidades en las caras de las celdas.

$$u_e = \frac{u_E + u_O}{2} \quad (2.55)$$

2.3.2.3 Resolución de las ecuaciones de cantidad de movimiento

Llega el momento de la resolución por separado de las ecuaciones de cantidad de movimiento en los ejes X e Y . Aunque la metodología empleada es la misma, una nos proporcionará el campo de velocidades horizontales \hat{u} y la otra el campo de velocidades verticales \hat{v} . Sin embargo, es crucial tener en cuenta que el campo de velocidades obtenido de esta resolución no será un campo de velocidades convergente completo. Es decir, cumplirá únicamente con las ecuaciones de cantidad de movimiento, pero aún no satisfará la ecuación de continuidad. Para distinguirlo, estos campos se representan con los caracteres (\hat{u}) y (\hat{v}) .

A partir de la ecuación discretizada (2.34), podría pensarse que bastaría con resolverla para las incógnitas u_O, u_E, u_W, u_N y u_S . Pero desde el punto de vista computacional, es más eficiente abordar la resolución por partes, aplicando un método ADI. En este estudio, se emplea el *método Line-by-line*, que resuelve primero las ecuaciones en dirección de las filas y luego en dirección de las columnas. Los detalles del *método Line-by-line* se pueden encontrar en el anexo A, explicado por Majumdar. Tal como se indica en la siguientes ecuaciones, en el primer barrido, las incógnitas son u_O, u_E y u_W , mientras que en el segundo barrido, las incógnitas son u_O, u_N , y u_S .

$$A_O^X \hat{\mathbf{u}}_O + A_W^X \hat{\mathbf{u}}_W + A_E^X \hat{\mathbf{u}}_E = S_O^X - A_N^X \hat{u}_N - A_S^X \hat{u}_S \quad (2.56)$$

$$A_O^X \hat{\mathbf{u}}_O + A_N^X \hat{\mathbf{u}}_N + A_S^X \hat{\mathbf{u}}_S = S_O^X - A_W^X \hat{u}_W - A_E^X \hat{u}_E \quad (2.57)$$

Para hacer aún más eficiente la resolución del sistema, se realiza en términos correctivos. Esto implica calcular la corrección necesaria para cada velocidad para cumplir con la ecuación, lo cual está relacionado con el concepto de los residuos de la ecuación, que se utilizan como criterio de convergencia. De esta manera, si se introducen los términos correctivos u' , se tiene a la izquierda las incógnitas a resolver y a la derecha el valor del residuo de dicha ecuación.

$$A_O^X \hat{u}_O^{k+1} + A_W^X \hat{u}_W^{k+1} + A_E^X \hat{u}_E^{k+1} = S_O^X - A_N^X \hat{u}_N^k - A_S^X \hat{u}_S^k \quad (2.58)$$

$$\hat{u}_i^{k+1} = \hat{u}_i^k + u'_i \quad i = O, E, W, N, S \quad (2.59)$$

$$A_O^X (\hat{u}_O^k + u'_O) + A_W^X (\hat{u}_W^k + u'_W) + A_E^X (\hat{u}_E^k + u'_E) = S_O^X - A_N^X \hat{u}_N^k - A_S^X \hat{u}_S^k \quad (2.60)$$

$$A_O^X \mathbf{u}'_O + A_W^X \mathbf{u}'_W + A_E^X \mathbf{u}'_E = S_O^X - A_N^X \hat{u}_N^k - A_S^X \hat{u}_S^k - A_O^X \hat{u}_O^k - A_W^X \hat{u}_W^k - A_E^X \hat{u}_E^k \quad (2.61)$$

$$R_O = S_O^X - A_N^X \hat{u}_N^k - A_S^X \hat{u}_S^k - A_O^X \hat{u}_O^k - A_W^X \hat{u}_W^k - A_E^X \hat{u}_E^k \quad (2.62)$$

Debido tanto a la discretización como a la linealización de la expresión, la mayoría de los métodos añaden un factor de amortiguamiento inercial $\omega_{inercial}$ que modera estas pequeñas correcciones en la velocidad, quedando la expresión donde el término de la derecha constituye los residuos de las ecuaciones. El valor de este amortiguamiento suele oscilar entre 0.2 y 0.8.

$$(1 + \omega_{inercial}) A_O^X \mathbf{u}'_O + A_W^X \mathbf{u}'_W + A_E^X \mathbf{u}'_E = R_O \quad (2.63)$$

El proceso para obtener el campo de velocidades comienza calculando los residuos de todas las celdas antes de iniciar las iteraciones. Esto se hace para evaluar la convergencia del campo de velocidades mediante la raíz del error cuadrático medio (RMSE), una medida comúnmente utilizada para evaluar la calidad del ajuste del modelo.

$$R_2^X = \sqrt{\sum_{i=2}^{n-1} \sum_{j=2}^{m-1} R_{O_{i,j}}^2} \quad (2.64)$$

Una vez completado este paso, comienza una iteración interna, donde se resuelven los sistemas primero por filas y luego por columnas. La ecuación 2.63, es propia de cada celda, pero forma parte del sistema para una fila j , de manera que como resultado, cada fila k en realidad constituye un sistema tridiagonal, fácil de resolver para cualquier código que tenga implementada una función TDMA. Después de resolver el sistema de la fila k , se actualizan las velocidades con las correcciones obtenidas y se pasa a la siguiente fila. Una vez completadas todas las filas, se repite el proceso, pero esta vez resolviendo los sistemas en columnas. Esta iteración se conoce como iteración interna, ya que se utiliza para resolver una ecuación específica [13].

$$A x = d \quad (2.65)$$

$$A = \begin{bmatrix} A_{O_{1,j}} & A_{E_{1,j}} & & & & 0 \\ A_{W_{2,j}} & A_{O_{2,j}} & A_{E_{2,j}} & & & \\ & A_{W_{3,j}} & A_{O_{3,j}} & \ddots & & \\ & & \ddots & \ddots & & A_{E_{n-1,j}} \\ 0 & & & & A_{W_{n,j}} & A_{O_{n,j}} \end{bmatrix}, \quad x = \begin{pmatrix} u'_{O_{1,j}} \\ u'_{O_{2,j}} \\ u'_{O_{3,j}} \\ \vdots \\ u'_{O_{n,j}} \end{pmatrix}, \quad d = \begin{pmatrix} R_{O_{1,j}} \\ R_{O_{2,j}} \\ R_{O_{3,j}} \\ \vdots \\ R_{O_{n,j}} \end{pmatrix}$$

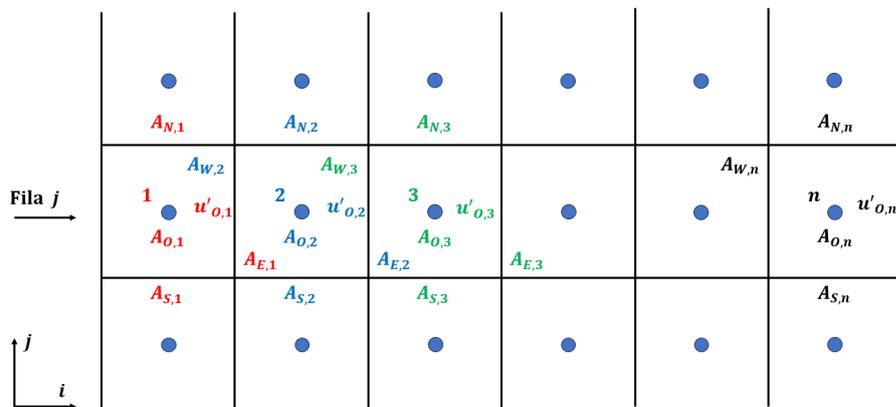


Figura 2.4 Ilustración del sistema de ecuaciones de la fila j .

2.3.2.4 Método PWIM para la interpolación de las velocidades en las caras

Hasta este punto, se han obtenido valores de velocidad en el centro de cada celda que cumplen con la ecuación de cantidad de movimiento, pero no con la ecuación de continuidad. Por lo tanto, la solución solo ha convergido parcialmente. Para lograr una convergencia completa, es necesario satisfacer también la ecuación de continuidad. Sin embargo, al analizar esta ecuación (2.54), se observa que los flujos están expresados en función de las velocidades en las caras de las celdas, no en las velocidades centrales de cada celda. Por lo tanto, se requiere algún tipo de interpolación para aproximar las velocidades en las caras. Aquí radica una de las claves del método. Como se explicó previamente, no cualquier tipo de interpolación es adecuada, ya que el uso de una interpolación basada en la ponderación de distancia puede dar lugar al problema del tablero de ajedrez, como se explicó en secciones anteriores.

La clave para interpolar los valores de velocidad en las caras radica en el método desarrollado por Rhie & Chow en 1983 y posteriormente refinado por Miller & Schmidt en 1988 [13], el cual todavía es ampliamente utilizado en muchos códigos CFD en la actualidad. Este método, conocido como PWIM (*Pressure-Weight Interpolation Method*) en inglés, utiliza una ponderación por presión en lugar de por distancia. A continuación, se muestra su desarrollo detallado.

La ecuación de cantidad de movimiento puede expresarse de manera más concisa si se agrupan todos los términos de las celdas adyacentes en un único sumatorio, como se muestra en 2.66. A continuación, se deriva una expresión para el valor de las velocidades u_O y v_O en el centro de la celda O .

$$\sum_{nb} A_{nb}^X \hat{u}_{nb} = A_W^X \hat{u}_W + A_E^X \hat{u}_E + A_N^X \hat{u}_N + A_S^X \hat{u}_S \quad (2.66)$$

$$A_O^X \hat{u}_O + \sum_{nb} A_{nb}^X \hat{u}_{nb} = S_O^X = \frac{p_W - p_E}{2} \Delta y \quad (2.67)$$

$$A_O^Y \hat{v}_O + \sum_{nb} A_{nb}^Y \hat{v}_{nb} = S_O^Y = \frac{p_S - p_N}{2} \Delta x \quad (2.68)$$

$$\hat{u}_O = -\frac{1}{A_{O|O}^X} \sum_{nb} A_{nb}^X \hat{u}_{nb} + \frac{p_W - p_E}{2 A_{O|O}^X} \Delta y \quad (2.69)$$

$$\hat{v}_O = -\frac{1}{A_{O|O}^Y} \sum_{nb} A_{nb}^Y \hat{v}_{nb} + \frac{p_S - p_N}{2 A_{O|O}^Y} \Delta x \quad (2.70)$$

De manera similar, la expresión (2.72) para la celda adyacente *Este* seguiría una formulación parecida, adaptando los términos de enlace y las presiones del término fuente. Siguiendo este desarrollo, es posible suponer una expresión similar para la velocidad en la cara este u_e como se muestra en (2.73). Hay que destacar que en esta expresión no es necesario incorporar la división por 2 en el término fuente, ya que el gradiente de presión en la cara de la celda es exacto y no aproximado como en las otras dos expresiones del centro de la celda.

$$\hat{u}_O = -\frac{1}{A_{O|O}^X} \sum_{nb} A_{nb|O}^X \hat{u}_{nb|O} + \frac{p_W - p_E}{2 A_{O|O}^X} \Delta y \quad (2.71)$$

$$\hat{u}_E = -\frac{1}{A_{O|E}^X} \sum_{nb} A_{nb|E}^X \hat{u}_{nb|E} + \frac{p_O - p_{EE}}{2 A_{O|E}^X} \Delta y \quad (2.72)$$

$$\hat{u}_e = -\frac{1}{A_{O|e}^X} \sum_{nb} A_{nb|e}^X \hat{u}_{nb|e} + \frac{p_O - p_E}{A_{O|e}^X} \Delta y \quad (2.73)$$

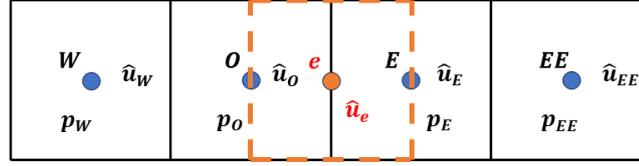


Figura 2.5 Aproximación de la interpolación de \hat{u}_e .

Analizando de nuevo en la expresión de u_e , no se dispone de una expresión para calcular los coeficientes de enlace $A_{O|e}^X$ en las caras, ni tampoco el sumatorio. Por lo tanto, aquí es necesario hacer una hipótesis para aproximar estos valores en función de lo que sí se conoce. La hipótesis consiste en suponer que estos valores pueden interpolarse según las expresiones (2.74) y (2.75).

$$\frac{1}{A_{O|e}^X} = \frac{1}{2} \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \quad (2.74)$$

$$\frac{1}{A_{O|e}^X} \sum_{nb} A_{nb|e}^X \hat{u}_{nb|e} = \frac{1}{2} \left[\frac{1}{A_{O|E}^X} \sum_{nb} A_{nb|E}^X \hat{u}_{nb|E} + \frac{1}{A_{O|O}^X} \sum_{nb} A_{nb|O}^X \hat{u}_{nb|O} \right] \quad (2.75)$$

Al revisar estas expresiones, es posible hacer una analogía con los sistemas eléctricos, ya que la hipótesis realizada puede recordar a la forma en que se suman las resistencias eléctricas en paralelo y en serie para obtener una resistencia equivalente. En este sentido, el concepto puede ser similar, ya que los coeficientes de enlace A_i incluyen los términos difusivos y convectivos que representan la resistencia que el fluido tiene para moverse por el dominio. En este caso, se usa ese mismo concepto para interpolar las resistencias en ambas celdas, para obtener una resistencia equivalente en la cara de interposición. Por lo tanto, estas expresiones pueden ser una buena hipótesis para aproximar el valor de la velocidad en las caras.

Siguiendo con el desarrollo y sustituyendo estas expresiones, se logra la expresión (2.76):

$$\begin{aligned} \hat{u}_e = & -\frac{1}{2} \left[\frac{1}{A_{O|E}^X} \sum_{nb} A_{nb|E}^X \hat{u}_{nb|E} + \frac{1}{A_{O|O}^X} \sum_{nb} A_{nb|O}^X \hat{u}_{nb|O} \right] + \\ & + \frac{1}{2} \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] (p_O - p_E) \Delta y \end{aligned} \quad (2.76)$$

Desarrollando un paso más allá y sustituyendo los sumatorios en función de las velocidades centrales y el término fuente, se consigue (2.77).

$$\hat{u}_e = \frac{1}{2} \left[\hat{u}_E - \frac{p_O - p_{EE}}{2 A_{O|E}^X} \Delta y + \hat{u}_O - \frac{p_W - p_E}{2 A_{O|O}^X} \Delta y \right] + \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_O - p_E}{2} \Delta y \quad (2.77)$$

Al reordenar los términos, se ve como en (2.78) los términos de presión son aproximaciones a los gradientes de presión, como queda patente en (2.79), de ahí el nombre de este método. Pues aquí se ve perfectamente, cómo la velocidad en la cara u_e ya no solo depende de una interpolación ponderada por distancia, si no que también está influenciada por los gradientes de presión de las celdas.

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{1}{2} \frac{p_{EE} - p_O}{2 A_{O|E}^X} \Delta y + \frac{1}{2} \frac{p_E - p_W}{2 A_{O|O}^X} \Delta y - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E - p_O}{2} \Delta y \quad (2.78)$$

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{1}{2} \Delta x \Delta y \left(\frac{1}{A_{O|E}^X} \frac{\partial p}{\partial x} \Big|_E + \frac{1}{A_{O|O}^X} \frac{\partial p}{\partial x} \Big|_O - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{\partial p}{\partial x} \Big|_e \right) \quad (2.79)$$

Además, en una primera iteración donde se comienza con una presión uniforme en todo el dominio, todos los términos de las parciales se anulan, dejando solo la ponderación de la velocidad. Validando la aproximación hecha para $k = 1$.

Siguiendo el mismo proceso, la velocidad vertical en la cara norte v_n será (2.80):

$$\hat{v}_n = \frac{\hat{v}_O + \hat{v}_N}{2} + \frac{1}{2} \frac{p_{NN} - p_O}{2 A_{O|N}^X} \Delta y + \frac{1}{2} \frac{p_N - p_S}{2 A_{O|O}^X} \Delta y - \left[\frac{1}{A_{O|N}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_N - p_O}{2} \Delta y \quad (2.80)$$

El resto de velocidades pueden obtenerse por continuidad de las celdas, ya que la v_s de una será la v_n de la celda *Sur*, y la u_w será la u_e de la celda de la *Oeste*.

$$\hat{u}_w|_O = \hat{u}_e|_W \quad (2.81)$$

$$\hat{v}_s|_O = \hat{v}_n|_S \quad (2.82)$$

2.3.2.5 Cálculo de los coeficientes de enlace y fuentes para la ecuación de continuidad

El próximo paso es resolver la ecuación de continuidad y para ello, es necesario calcular los coeficientes de enlace y las fuentes. A pesar de haber desarrollado expresiones para las velocidades en las caras (2.78, 2.80, y), no basta con simplemente sustituirlas en (2.54). Como se ha mencionado anteriormente en este documento, el objetivo es obtener una expresión que contenga explícitamente las presiones. En esta sección, se presentará el desarrollo necesario para lograr esta formulación.

El proceso comienza de manera similar a la ecuación de cantidad de movimiento, consiste en expresar las velocidades en forma de correcciones. La velocidad resultante u_i , que representa la convergencia completa del problema, puede expresarse como la suma de la velocidad de convergencia parcial \hat{u}_i calculada en los pasos anteriores, más una corrección u'_i . Al hacer esto, la expresión (2.54) puede reescribirse como (2.190).

$$u_i = \hat{u}_i + u'_i \quad (2.83)$$

$$(\rho_e u'_e - \rho_w u'_w) \Delta y + (\rho_n v'_n - \rho_s v'_s) \Delta x = -[(\rho_e \hat{u}_e - \rho_w \hat{u}_w) \Delta y + (\rho_n \hat{v}_n - \rho_s \hat{v}_s) \Delta x] \quad (2.84)$$

$$\dot{m}_{NB} = [(\rho_e \hat{u}_e - \rho_w \hat{u}_w) \Delta y + (\rho_n \hat{v}_n - \rho_s \hat{v}_s) \Delta x] \quad (2.85)$$

$$(\rho_e u'_e - \rho_w u'_w) \Delta y + (\rho_n v'_n - \rho_s v'_s) \Delta x = -\dot{m}_{NB} \quad (2.86)$$

Analizando la parte derecha de la ecuación, se observa que representa el gasto no balanceado \dot{m}_{NB} de la celda. Si este fuera cero, no sería necesaria ninguna corrección de las velocidades. Dicho valor \dot{m}_{NB} es conocido, ya que las expresiones para las velocidades en las caras se han deducido con la interpolación PWIM. Por otro lado, en el término izquierdo de la ecuación, quedan los valores de las correcciones de las velocidades u'_i como incógnitas. Serán estos valores los que se deban desarrollar para expresarlos en términos de presiones.

Comenzando por los términos en el centro de las caras u_O , se sustituye su expresión según lo deducido en (2.87), lo que resulta en la expresión (2.89). A destacar que los coeficientes de enlace, son iguales para u_i y \hat{u}_i , puesto que se trata de la misma iteración externa. Lo único que se pretende con este paso es corregir la velocidad para que también se cumpla la continuidad. El primer término de (2.89) puede despreciarse, ya que cuando la solución converja, las correcciones de las velocidades deben tender a cero. Como conclusión, en una primera aproximación, es razonable quedarse solo con el segundo término, que representa la diferencia entre las presiones corregidas (2.90). De igual forma para las velocidades verticales (2.91).

$$\hat{u}_O = -\frac{1}{A_{O|O}^X} \sum_{nb} A_{nb}^X \hat{u}_{nb} + \frac{p_W^k - p_E^k}{2 A_{O|O}^X} \Delta y \quad (2.87)$$

$$u_O = -\frac{1}{A_{O|O}^X} \sum_{nb} A_{nb}^X u_{nb} + \frac{p_W^{k+1} - p_E^{k+1}}{2 A_{O|O}^X} \Delta y \quad (2.88)$$

$$u'_O = \hat{u}_O - u_O = -\frac{1}{A_{O|O}^X} \sum_{nb} A_{nb}^X \hat{u}'_{nb} + \frac{p'_W - p'_E}{2 A_{O|O}^X} \Delta y \quad (2.89)$$

$$u'_O \approx \frac{p'_W - p'_E}{2 A_{O|O}^X} \Delta y \quad (2.90)$$

$$v'_O \approx \frac{p'_S - p'_N}{2 A_{O|O}^X} \Delta x \quad (2.91)$$

Es el momento de abordar el cálculo de las correcciones de las velocidades en las caras para cumplir con la ecuación de continuidad. Utilizando la expresión derivada con la interpolación PWIM, simplemente se sustituye para ver qué se obtiene (2.78). Al combinar los términos, se logra que todas las velocidades y presiones que quedan sean correcciones.

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{1}{2} \frac{p_{EE}^k - p_O^k}{2 A_{O|E}^X} \Delta y + \frac{1}{2} \frac{p_E^k - p_W^k}{2 A_{O|O}^X} \Delta y - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E^k - p_O^k}{2} \Delta y \quad (2.92)$$

$$u_e = \frac{u_O + u_E}{2} + \frac{1}{2} \frac{p_{EE}^{k+1} - p_O^{k+1}}{2 A_{O|E}^X} \Delta y + \frac{1}{2} \frac{p_E^{k+1} - p_W^{k+1}}{2 A_{O|O}^X} \Delta y - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E^{k+1} - p_O^{k+1}}{2} \Delta y \quad (2.93)$$

$$u'_e = u_e - \hat{u}_e \quad (2.94)$$

$$u'_e = \frac{u'_O + u'_E}{2} + \frac{1}{2} \frac{p'_{EE} - p'_O}{2 A_{O|E}^X} \Delta y + \frac{1}{2} \frac{p'_E - p'_W}{2 A_{O|O}^X} \Delta y - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p'_E - p'_O}{2} \Delta y \quad (2.95)$$

Es importante destacar que estas correcciones de velocidad en el centro de la celda se acaban de deducir (2.90) y pueden aplicarse en este desarrollo. Al hacerlo, todos los términos se anulan excepto el último, lo que permite expresar la corrección de velocidad en la cara *Este* como (2.97). El proceso es similar para las demás velocidades, obteniendo expresiones análogas (2.98, 2.99 y 2.100).

$$\begin{aligned}
 u'_O = & + \frac{1}{2} \frac{p'_O - p'_{EE}}{2 A^X_{O|E}} \Delta y + \frac{1}{2} \frac{p'_W - p'_E}{2 A^X_{O|O}} \Delta y + \\
 & + \frac{1}{2} \frac{p'_{EE} - p'_O}{2 A^X_{O|E}} \Delta y + \frac{1}{2} \frac{p'_E - p'_W}{2 A^X_{O|O}} \Delta y - \\
 & - \left[\frac{1}{A^X_{O|E}} + \frac{1}{A^X_{O|O}} \right] \frac{p'_E - p'_O}{2} \Delta y
 \end{aligned} \tag{2.96}$$

$$u'_e = \left[\frac{1}{A^X_{O|E}} + \frac{1}{A^X_{O|O}} \right] \frac{p'_O - p'_E}{2} \Delta y \tag{2.97}$$

$$u'_w = \left[\frac{1}{A^X_{O|W}} + \frac{1}{A^X_{O|O}} \right] \frac{p'_W - p'_O}{2} \Delta y \tag{2.98}$$

$$v'_n = \left[\frac{1}{A^Y_{O|N}} + \frac{1}{A^Y_{O|O}} \right] \frac{p'_O - p'_N}{2} \Delta x \tag{2.99}$$

$$v'_s = \left[\frac{1}{A^Y_{O|S}} + \frac{1}{A^Y_{O|O}} \right] \frac{p'_S - p'_O}{2} \Delta x \tag{2.100}$$

Finalmente, es posible sustituir estas aproximaciones en la ecuación de continuidad (2.86), donde las incógnitas explícitas pasan a ser las correcciones de presión. Al reorganizar los términos y expresarlos en forma de coeficientes de enlace, la ecuación resultante adopta la forma (2.102), con las correcciones de presión como incógnitas. De esta manera, se consigue formular la ecuación de continuidad en términos de presiones y obtener expresiones para los coeficientes de enlace y fuentes.

$$\begin{aligned}
 & \left(\rho_e \left[\frac{1}{A^X_{O|E}} + \frac{1}{A^X_{O|O}} \right] \frac{p'_O - p'_E}{2} \Delta y - \rho_w \left[\frac{1}{A^X_{O|W}} + \frac{1}{A^X_{O|O}} \right] \frac{p'_W - p'_O}{2} \Delta y \right) \Delta y + \\
 & + \left(\rho_n \left[\frac{1}{A^Y_{O|N}} + \frac{1}{A^Y_{O|O}} \right] \frac{p'_O - p'_N}{2} \Delta x - \rho_s \left[\frac{1}{A^Y_{O|S}} + \frac{1}{A^Y_{O|O}} \right] \frac{p'_S - p'_O}{2} \Delta x \right) \Delta x = -\dot{m}_{NB}
 \end{aligned} \tag{2.101}$$

$$A^p_O p'_O + A^p_W p'_W + A^p_E p'_E + A^p_N p'_N + A^p_S p'_S = S^p_O \tag{2.102}$$

$$\begin{aligned}
 A^p_O = & \frac{\rho_e \Delta y^2}{2} \left[\frac{1}{A^X_{O|E}} + \frac{1}{A^X_{O|O}} \right] + \frac{\rho_w \Delta y^2}{2} \left[\frac{1}{A^X_{O|W}} + \frac{1}{A^X_{O|O}} \right] + \\
 & + \frac{\rho_n \Delta x^2}{2} \left[\frac{1}{A^Y_{O|N}} + \frac{1}{A^Y_{O|O}} \right] + \frac{\rho_s \Delta x^2}{2} \left[\frac{1}{A^Y_{O|S}} + \frac{1}{A^Y_{O|O}} \right]
 \end{aligned} \tag{2.103}$$

$$A_E^p = -\frac{\rho_e \Delta y^2}{2} \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \quad (2.104)$$

$$A_W^p = -\frac{\rho_w \Delta y^2}{2} \left[\frac{1}{A_{O|W}^X} + \frac{1}{A_{O|O}^X} \right] \quad (2.105)$$

$$A_N^p = -\frac{\rho_n \Delta x^2}{2} \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] \quad (2.106)$$

$$A_S^p = -\frac{\rho_s \Delta x^2}{2} \left[\frac{1}{A_{O|S}^Y} + \frac{1}{A_{O|O}^Y} \right] \quad (2.107)$$

$$S_O^p = -\dot{m}_{NB} \quad (2.108)$$

2.3.2.6 Resolución de la ecuación de continuidad

Para abordar la resolución de la ecuación de continuidad, ahora expresada explícitamente en términos de presiones, se deben seguir las pautas establecidas en la sección de resolución de las ecuaciones de cantidad de movimiento, ya que el método para su resolución sigue basándose en el Método Line-by-line.

En primer lugar, se calculan los residuos de las ecuaciones de las celdas interiores, dado que el resto están sometidas a aproximaciones y los errores son pequeños. Para evaluar estos residuos, hay que centrarse en los términos fuente de las ecuaciones, que representan el gasto no balanceado. Este hecho tiene sentido físico, ya que mientras exista un desequilibrio en el gasto de una celda, la presión debe ajustarse para compensar dicho desequilibrio y, por ende, las velocidades de los flujos hasta alcanzar una cierta tolerancia. Como conclusión, para evaluar la convergencia de la continuidad del dominio, se monitorizan los gastos no balanceados, que se pueden calcular conociendo las velocidades en las caras de las celdas, como se dedujo en la expresión (2.85).

$$R_2^p = \sqrt{\sum_{i=2}^{n-1} \sum_{j=2}^{m-1} \dot{m}_{NB_{i,j}}^2} \quad (2.109)$$

En cuanto a las correcciones de las presiones, se resuelven de forma iterativa siguiendo el Método Line-by-line, con una primera pasada por filas y luego por columnas, repitiendo estas iteraciones internas un cierto número de veces o hasta alcanzar la tolerancia fijada.

$$A x = d \quad (2.110)$$

$$A = \begin{bmatrix} A_{O_{1,j}}^p & A_{E_{1,j}}^p & & & 0 \\ A_{W_{2,j}}^p & A_{O_{2,j}}^p & A_{E_{2,j}}^p & & \\ & A_{W_{3,j}}^p & A_{O_{3,j}}^p & \ddots & \\ & & \ddots & \ddots & A_{E_{n-1,j}}^p \\ 0 & & & A_{W_{n,j}}^p & A_{O_{n,j}}^p \end{bmatrix}, \quad x = \begin{pmatrix} p'_{O_{1,j}} \\ p'_{O_{2,j}} \\ p'_{O_{3,j}} \\ \vdots \\ p'_{O_{n,j}} \end{pmatrix}, \quad d = \begin{pmatrix} \dot{m}_{O_{1,j}} \\ \dot{m}_{O_{2,j}} \\ \dot{m}_{O_{3,j}} \\ \vdots \\ \dot{m}_{O_{n,j}} \end{pmatrix}$$

Una reflexión necesaria en este punto es: ¿qué presión se está corrigiendo? ¿La presión total o la presión relativa? Atendiendo a la definición de la presión en un sistema, se puede representar como la presión total, que es la suma de una presión de referencia y una presión relativa o manométrica. Dado que lo que realmente afecta a la ecuación de continuidad son los cambios en la presión, es decir, las correcciones, es independiente de si se está considerando la presión total o la presión manométrica. Por conveniencia en la resolución, y tal como lo hacen la mayoría de los códigos comerciales, es más ventajoso utilizar la presión manométrica y realizar las correcciones sobre esta última, ya que en términos de precisión numérica, las correcciones serán cantidades muy pequeñas en comparación con la presión total. La única excepción a esta consideración sería si alguna otra magnitud dependiera de la presión, como en el caso de un problema compresible, donde la densidad dependería de la presión total.

$$p_t = p_{ref} + p_m \quad (2.111)$$

$$p_{t\text{corregida}} = p_{ref} + p_{m\text{corregida}} \quad (2.112)$$

$$p_{m\text{corregida}} = p_m + p' \quad (2.113)$$

$$p_{t\text{corregida}} - p_t = p_{m\text{corregida}} - p_m = p' \quad (2.114)$$

2.3.2.7 Actualización de presiones y velocidades

Una vez resueltas las correcciones de presión, se procede a actualizar tanto las velocidades como las presiones del campo fluido. Es fundamental destacar que, en el caso de las velocidades, al tratarse de un mallado co-ubicado, no solo se debe actualizar las velocidades centrales de las celdas, sino también los valores de las velocidades en las caras. Recordar que estos últimos serán necesarios en la siguiente iteración para recalculer los coeficientes de enlace de las ecuaciones de cantidad de movimiento. Esta actualización constituye la segunda ocasión durante el algoritmo en la que se modifica el campo de velocidades. En la primera actualización, se alcanza solo una convergencia parcial al resolver la ecuación de cantidad de movimiento. En cambio, en la segunda actualización, se cumple también con la ecuación de continuidad, lo que implica una convergencia completa.

En lo que respecta a las expresiones para actualizar los valores de velocidades y presiones, es necesario incorporar factores de amortiguamiento ω_p y ω_{uv} . Estos factores son indispensables debido a la aproximación realizada durante el desarrollo de la ecuación de continuidad, donde se despreció el primer término (2.89). Además, como es común en muchos métodos numéricos, la inclusión de un factor de amortiguamiento o relajación lineal al resolver ecuaciones linealizadas de forma numérica contribuye a mejorar la estabilidad, la convergencia y el control del proceso de solución, especialmente en sistemas dinámicos o sensibles a perturbaciones, como es el caso de este estudio.

$$p_O^{k+1} = p_O^k + \omega_p p'_O \quad (2.115)$$

$$u_O^{k+1} = \hat{u}_O + \omega_{uv} u'_O = \hat{u}_O + \omega_{uv} \frac{p'_W - p'_E}{2 A_{O|O}^X} \Delta y \quad (2.116)$$

$$v_O^{k+1} = \hat{v}_O + \omega_{uv} v'_O = \hat{v}_O + \omega_{uv} \frac{p'_S - p'_N}{2 A_{O|O}^Y} \Delta x \quad (2.117)$$

$$u_e^{k+1} = \hat{u}_e + \omega_{uv} u'_e = \hat{u}_e + \omega_{uv} \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p'_O - p'_E}{2} \Delta y \quad (2.118)$$

$$v_n^{k+1} = \hat{v}_n + \omega_{uv} v'_n = \hat{v}_n + \omega_{uv} \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] \frac{p'_O - p'_N}{2} \Delta x \quad (2.119)$$

$$u_w|_O = u_e|_W \quad (2.120)$$

$$v_s|_O = v_n|_S \quad (2.121)$$

2.3.2.8 Evaluación de la convergencia de los residuos

En la etapa final del proceso iterativo, es crucial evaluar la convergencia de los resultados, determinando si es necesario continuar iterando o si se puede considerar el proceso como finalizado. En el marco del algoritmo SIMPLE, este proceso se lleva a cabo mediante la evaluación de los residuos de las ecuaciones de cantidad de movimiento y continuidad, que se calculan al inicio de cada iteración en su correspondiente bloque. Es importante recordar que estos residuos representan la raíz del error cuadrático medio, una medida estándar de la discrepancia entre los resultados obtenidos y los esperados.

Para determinar la convergencia, se comparan los valores de los residuos R_2^X , R_2^Y y R_2^p con una tolerancia predefinida, que en este estudio se establece en 10^{-10} . Si todos los residuos son inferiores a esta tolerancia, se detiene la iteración, indicando que se ha alcanzado una convergencia satisfactoria en el campo de velocidades y presiones. En caso contrario, se continúa el proceso iterativo comenzando por la actualización de los coeficientes de enlace y fuentes de las ecuaciones de cantidad de movimiento.

Es recomendable adoptar dos precauciones adicionales durante este proceso. En primer lugar, se sugiere visualizar los residuos a medida que avanza el algoritmo, lo que permite detectar si están convergiendo o divergiendo. En segundo lugar, se debe establecer un límite máximo de iteraciones del algoritmo para evitar posibles divergencias en los residuos. Este límite de iteraciones debe ajustarse según la complejidad del problema que se esté resolviendo, ya que puede variar considerablemente según la configuración específica del dominio fluido y las condiciones de contorno.

En resumen, el algoritmo SIMPLE representa un proceso iterativo que gradualmente converge hacia una solución del campo de velocidades y presiones en un dominio fluido. Inicia con condiciones iniciales y se enfoca inicialmente en ajustar las velocidades, las cuales determinan los flujos que ingresan y salen de cada celda del dominio.

A partir de los resultados de velocidad, se derivan los gastos no balanceados, que necesitan ajustarse mediante correcciones en las presiones. Por ejemplo, si en una celda el flujo de entrada es elevado respecto al flujo de salida, esto incrementará la presión, lo que a su vez influirá en las velocidades en la siguiente iteración. Este proceso de ajuste continúa iteración tras iteración, con la presión influyendo en las velocidades y viceversa, hasta que se alcanza un equilibrio. Este equilibrio se traduce en la convergencia del algoritmo, donde las velocidades y las presiones alcanzan valores estables iteración tras iteración. Alternativamente, si se establece un límite en el número de iteraciones y este límite se alcanza antes de lograr la convergencia, el proceso se detiene.

2.3.3 Implementación del algoritmo SIMPLE en Matlab

Para la implementación de este algoritmo, se ha diseñado un programa principal que actúa como punto central, donde se definen y gestionan los parámetros clave del problema a resolver. Estos parámetros incluyen la geometría del dominio fluido, las propiedades del fluido, el mallado co-ubicado estructurado, las tolerancias e iteraciones y los factores de amortiguamiento, entre otros aspectos fundamentales. La estructura principal del algoritmo SIMPLE se encuentra en este programa principal, el cual se encarga de llamar a cada bloque funcional representado en la figura 2.3 como una función independiente.

Además de la lógica del algoritmo, el programa principal también incluye una funcionalidad para generar una representación gráfica dinámica de los residuos, a medida que se realizan las iteraciones. Esta visualización en tiempo real proporciona una manera efectiva de monitorear la convergencia del algoritmo y comprender el comportamiento del sistema durante su ejecución.

Una vez que se completa la ejecución del algoritmo, el programa principal se encarga de almacenar los resultados obtenidos, incluyendo los campos de velocidades y presiones, así como los datos relacionados con los residuos y el número de Reynolds. Estos resultados son utilizados más adelante para análisis adicionales en procesos de postprocesado o para su visualización en herramientas externas. La integración de todas estas funcionalidades en un único programa ofrece una solución completa y eficiente para la resolución de problemas de dinámica de fluidos mediante el algoritmo SIMPLE.

El código implementado en Matlab, que incluye tanto el programa principal como las funciones auxiliares, está disponible para su consulta en el Anexo C adjunto a este documento. Este recurso proporciona una referencia detallada y accesible para aquellos que deseen examinar el código fuente utilizado en la implementación del algoritmo descrito.

2.4 Particularización de las ecuaciones para las condiciones de contorno

Una vez deducidas las expresiones generales que deben emplearse para las celdas interiores en el algoritmo SIMPLE, es momento de revisar estas ecuaciones y considerar cómo se deben adaptar según las diversas condiciones de contorno presentes en los dominios fluidos que se estén analizando. Estas adaptaciones deben implementarse en el código dentro del contexto de las celdas afectadas por cada condición de contorno específica. Para lograr esta particularización de las ecuaciones, es crucial comenzar con las expresiones generales no discretizadas y luego aplicar nuevas aproximaciones. No es viable derivarlas directamente de las ecuaciones discretizadas, ya que estas últimas incorporan otras aproximaciones que podrían no ser válidas para estos casos particulares.

Se expondrán cuatro condiciones de contorno típicas:

- Pared sin deslizamiento
- Pared con deslizamiento
- Entrada de fluido a velocidad constante
- Salida de fluido a presión constante

Cada una de estas condiciones requerirá ajustes específicos en las ecuaciones del algoritmo, los cuales se explicarán detalladamente para su implementación correcta en el código.

2.4.1 Pared sin deslizamiento

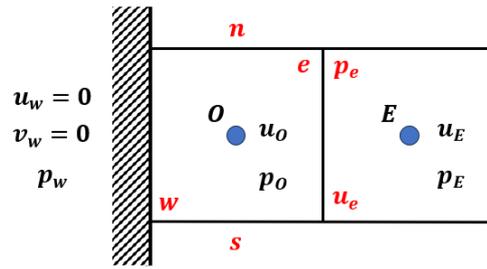


Figura 2.6 Pared izquierda sin deslizamiento.

La condición de pared sin deslizamiento, comúnmente denotada como "Wall" en muchos programas comerciales, es esencial en los dominios fluidos, ya que el fluido debe permanecer confinado dentro del recinto. Esta condición se caracteriza porque el fluido en contacto con la pared no presenta movimiento, es decir, tanto su velocidad horizontal como vertical son nulas. Respecto a la presión sobre la pared, aunque es desconocida, nunca será nula, lo que la convierte en una incógnita que debe aproximarse de alguna manera.

En este caso particular, se considera una pared situada a la izquierda (figura 2.6), lo que implica que las velocidades en la cara oeste de la celda sean nulas. El proceso de desarrollo que se presentará a continuación puede aplicarse a cualquier otra pared, adaptando los términos según la ubicación específica de la misma. Para adaptar las ecuaciones, se seguirá el diagrama del algoritmo SIMPLE, revisando cada uno de los bloques y ajustando los términos según sea necesario.

Ecuación de cantidad de movimiento

Se comienza por examinar la adaptación de las ecuaciones de cantidad de movimiento (2.122 y 2.123). Dado que la pared se encuentra a la izquierda w , se deben revisar los términos relacionados con el flujo J_w y la presión p_w .

$$(J_e^X - J_w^X) \Delta y + (J_n^X - J_s^X) \Delta x = (p_w - p_e) \Delta y \quad (2.122)$$

$$(J_e^Y - J_w^Y) \Delta y + (J_n^Y - J_s^Y) \Delta x = (p_s - p_n) \Delta x \quad (2.123)$$

Al considerar el flujo J_w , se recuerda que este término abarca tanto el flujo convectivo como el flujo difusivo, expresado tanto para el eje X (2.124) como para el eje Y (2.125). Al aplicar la condición de contorno de pared, las velocidades son nulas, lo que implica que no hay flujo convectivo y solo queda el término difusivo, el cual debe aproximarse de alguna manera.

$$J_w^X = \rho_w u_w u_w - \mu_w \left. \frac{\partial u}{\partial x} \right|_w \quad (2.124)$$

$$J_w^Y = \rho_w u_w v_w - \mu_w \left. \frac{\partial v}{\partial x} \right|_w \quad (2.125)$$

Una aproximación sencilla es utilizar una aproximación de Taylor de segundo orden basada en dos puntos. Por ejemplo, al desarrollar los términos para el eje X, donde u_O y u_E se expresan en función de u_w , se puede combinar ambos para despejar el valor de la derivada parcial de u con respecto a x en w .

$$u_O = u_w + \frac{\Delta x}{2} \frac{\partial u}{\partial x} \Big|_w + \frac{(\Delta x/2)^2}{2!} \frac{\partial^2 u}{\partial x^2} \Big|_w \quad (2.126)$$

$$u_E = u_w + \frac{3\Delta x}{2} \frac{\partial u}{\partial x} \Big|_w + \frac{(3\Delta x/2)^2}{2!} \frac{\partial^2 u}{\partial x^2} \Big|_w \quad (2.127)$$

Despejando $\frac{\partial^2 u}{\partial x^2} \Big|_w$,

$$\frac{\partial^2 u}{\partial x^2} \Big|_w = \frac{8}{\Delta x^2} \left[u_O - u_w - \frac{\Delta x}{2} \frac{\partial u}{\partial x} \Big|_w \right] \quad (2.128)$$

Sustituyendo en u_E ,

$$u_E = u_w + \frac{3\Delta x}{2} \frac{\partial u}{\partial x} \Big|_w + \frac{9\Delta x^2}{8} \frac{8}{\Delta x^2} \left[u_O - u_w - \frac{\Delta x}{2} \frac{\partial u}{\partial x} \Big|_w \right] \quad (2.129)$$

Despejando $\frac{\partial u}{\partial x} \Big|_w$,

$$\frac{\partial u}{\partial x} \Big|_w = \frac{9u_O - u_E - 8u_w}{3\Delta x} \quad (2.130)$$

Esta estimación del flujo difusivo (2.130) también será de gran utilidad en otras condiciones de contorno que se analicen más adelante en este estudio. Proporciona una forma general de abordar el flujo difusivo en las fronteras donde las velocidades son nulas, permitiendo adaptar eficazmente el tratamiento de los términos difusivos en diversas situaciones de contorno.

$$J_w^X \approx -\mu_w \frac{9u_O - u_E - 8u_w}{3\Delta x} \quad (2.131)$$

Esto permite obtener una aproximación del valor del flujo J_w^X en función de u_O , u_E y u_w , pero, dado que u_w es nulo por la condición de la pared, J_w^X , queda como la expresión 2.132. El mismo razonamiento, sirve para J_w^Y , quedando en función de v_O y v_E .

$$J_w^X \approx -\mu_w \frac{9u_O - u_E}{3\Delta x} \quad (2.132)$$

$$J_w^Y \approx -\mu_w \frac{9v_O - v_E}{3\Delta x} \quad (2.133)$$

Respecto al valor de la presión p_w , como se ha indicado anteriormente, es desconocido pero se sabe que no es nulo, por lo que es necesario realizar una estimación. Mazumder [13] sugiere en este punto, emplear la teoría de la capa límite, la cual es propia de la mecánica de fluidos y se aplica en zonas cercanas a superficies donde los efectos viscosos son significativos, como es el caso que se

está analizando. Por lo tanto, es apropiado considerar que dicha celda se puede encontrar dentro de la región conocida como capa límite.

En esta teoría, se demuestra que la variación de la velocidad es mucho mayor en el eje perpendicular a la superficie que en el paralelo a ella. Es decir, según se ilustra en la figura (2.7), la componente vertical de la velocidad v experimentará un mayor incremento que la componente horizontal u . Al relacionar este fenómeno con la ecuación de continuidad, se puede deducir que los gradientes de presión en dirección y son mucho mayores que los gradientes de presión en x . Esto significa que la presión en dirección x desde la superficie varía muy poco dentro de la propia capa límite.

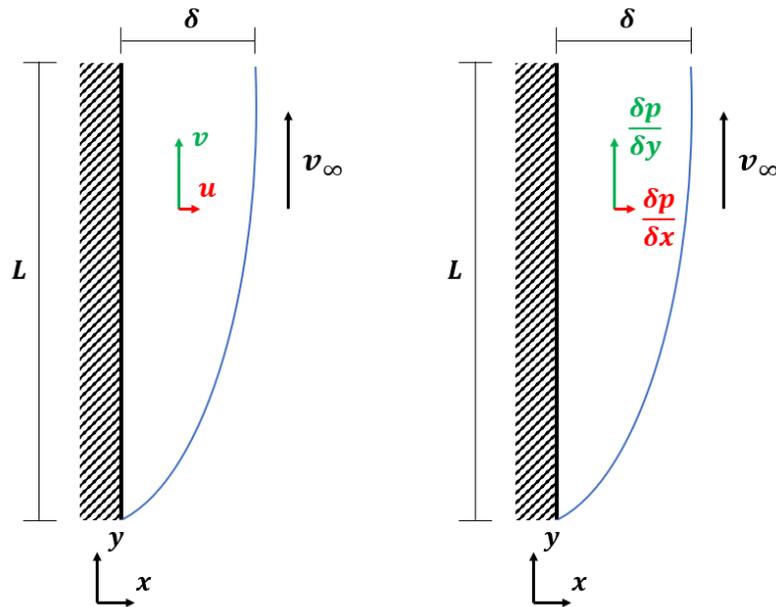


Figura 2.7 Capa límite.

$$\frac{\partial p}{\partial y} \gg \frac{\partial p}{\partial x} \tag{2.134}$$

En conclusión, una buena estimación podría ser considerar que la presión p_w es igual a la presión en el centro de la celda p_O . El error asociado a esta aproximación sería del orden de Δx , cuyo valor suele ser pequeño debido al tamaño de la malla empleada.

$$p_w \approx p_O \tag{2.135}$$

Otro enfoque posible para estimar p_w podría ser de naturaleza matemática, recurriendo nuevamente a una aproximación de Taylor de segundo orden basada en p_O . Al desarrollar los términos y retener hasta el segundo término, es necesario aproximar el gradiente de presiones en O , el cual puede aproximarse mediante el gradiente en la cara e , que se conoce en función de las presiones p_O y p_E . Esto conduce a otra posible estimación, como se muestra en la ecuación 2.138 para el valor de p_w . Al observar esta estimación, se ve claramente que la primera aproximación realizada, basada en la teoría de la capa límite, sería equivalente a considerar una aproximación de primer orden de Taylor.

$$p_w = p_O - \frac{\Delta x}{2} \left. \frac{\partial p}{\partial x} \right|_O \quad (2.136)$$

$$\left. \frac{\partial p}{\partial x} \right|_O \approx \left. \frac{\partial p}{\partial x} \right|_e = \frac{p_E - p_O}{\Delta x} \quad (2.137)$$

$$p_w = p_O - \frac{\Delta x}{2} \frac{p_E - p_O}{\Delta x} = \frac{3}{2}p_O - \frac{1}{2}p_E \quad (2.138)$$

Una vez realizadas las estimaciones para J_w y p_w , simplemente se procede a sustituir estos valores en las ecuaciones de cantidad de movimiento y luego a reagrupar los términos para obtener los nuevos coeficientes de enlace A_i^X , A_i^Y y las fuentes S^X .

$$A_O^X = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{3\mu_w}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (2.139)$$

$$A_E^X = - \left[\frac{|(\rho u)_e| - (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{\mu_w}{3\Delta x} \right] \Delta y \quad (2.140)$$

$$A_W^X = 0 \quad (2.141)$$

$$\text{Aprox. 1º Orden} \quad S_O^X = \frac{p_O - p_E}{2} \Delta y \quad (2.142)$$

$$\text{Aprox. 2º Orden} \quad S_O^X = (p_O - p_E) \Delta y \quad (2.143)$$

$$A_O^Y = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{3\mu_w}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (2.144)$$

$$A_E^Y = - \left[\frac{|(\rho u)_e| - (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{\mu_w}{3\Delta x} \right] \Delta y \quad (2.145)$$

$$A_W^Y = 0 \quad (2.146)$$

Es evidente que los coeficientes de enlace afectados son únicamente los términos O , E y W . En el caso del coeficiente W , se anula debido a que no se incorpora flujo desde la pared. Sin embargo, en los coeficientes O y E , surge un nuevo término como resultado de la estimación del flujo difusivo en la pared. Además, la expresión de la fuente S^X se altera, dependiendo de si se elige la aproximación de primer orden o de segundo.

Interpolación de velocidades en las caras por el método PWIM

El siguiente ajuste que se debe considerar, y que se ve influenciado por las condiciones de contorno, concierne a la interpolación PWIM para determinar el valor de las velocidades en las caras de la celda O . En este escenario específico donde el muro se ubica al *Oeste*, las velocidades en las caras *Oeste* y *Este* se ven afectadas. En el caso de \hat{u}_w , por la condición de contorno, es nula. En cuanto a \hat{u}_e , es la más afectada, ya que, al revisar la expresión 2.78 deducida anteriormente con la interpolación de PWIM, esta depende del gradiente de presiones en la celda O , que se aproximaba como la diferencia ponderada de las presiones p_w y p_E . Sin embargo, ahora, la estimación realizada para p_w es diferente. Por coherencia, se debe aplicar la misma aproximación anterior, ya sea de primer orden o de segundo orden. Por ende, la nueva expresión queda como:

$$\hat{u}_w|_O = 0 \tag{2.147}$$

Aprox. 1º Orden

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{1}{2} \frac{p_{EE} - p_O}{2 A_{O|E}^X} \Delta y + \frac{\Delta y}{2 A_{O|O}^X} \left[\frac{p_E - p_O}{2} \right] - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E - p_O}{2} \Delta y \tag{2.148}$$

Aprox. 2º Orden

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{1}{2} \frac{p_{EE} - p_O}{2 A_{O|E}^X} \Delta y + \frac{\Delta y}{2 A_{O|O}^X} [p_E - p_O] - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E - p_O}{2} \Delta y \tag{2.149}$$

En este contexto, es crucial tener precaución en otros dos casos particulares que involucran celdas adyacentes a la pared, ya sea en el lado derecho o en la parte superior, dado que también se deben ajustar de manera especial las velocidades en las caras. Esto se debe a que, en estas celdas, como se muestra en la figura 2.8, el término afectado es el gradiente de presiones de la celda E o N , donde las presiones p_{ee} o p_{nn} son desconocidas y se aproximan mediante una aproximación de Taylor de primer o segundo orden. Por lo tanto, siguiendo el mismo razonamiento anterior, se adaptan de la siguiente manera:

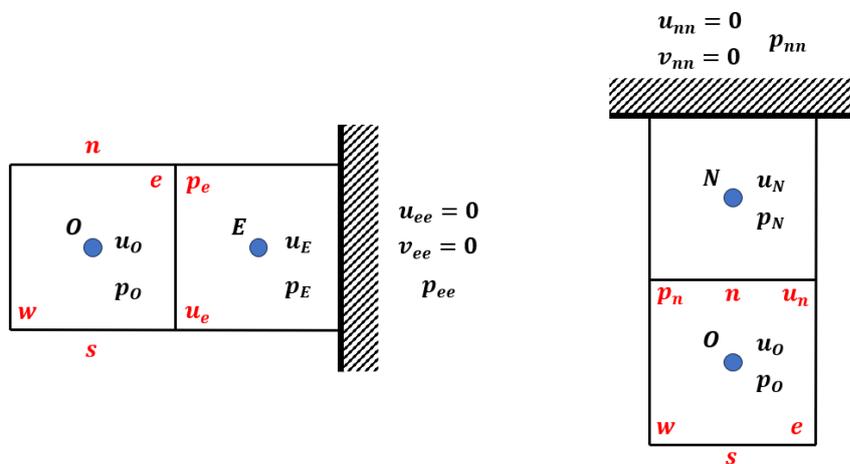


Figura 2.8 Adaptación de velocidades u_e y v_n en condiciones cercanas a la pared.

Aprox. 1º Orden

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{\Delta y}{2 A_{O|E}^X} \left[\frac{p_E - p_O}{2} \right] + \frac{1}{2} \frac{p_E - p_W}{2 A_{O|O}^X} \Delta y - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E - p_O}{2} \Delta y \quad (2.150)$$

$$\hat{v}_n = \frac{\hat{v}_O + \hat{v}_N}{2} + \frac{\Delta x}{2 A_{O|N}^Y} \left[\frac{p_N - p_O}{2} \right] + \frac{1}{2} \frac{p_N - p_S}{2 A_{O|O}^Y} \Delta y - \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] \frac{p_N - p_O}{2} \Delta y \quad (2.151)$$

Aprox. 2º Orden

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{\Delta y}{2 A_{O|E}^X} [p_E - p_O] + \frac{1}{2} \frac{p_E - p_W}{2 A_{O|O}^X} \Delta y - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E - p_O}{2} \Delta y \quad (2.152)$$

$$\hat{v}_n = \frac{\hat{v}_O + \hat{v}_N}{2} + \frac{\Delta x}{2 A_{O|N}^Y} [p_N - p_O] + \frac{1}{2} \frac{p_N - p_S}{2 A_{O|O}^Y} \Delta y - \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] \frac{p_N - p_O}{2} \Delta y \quad (2.153)$$

Corrección de presiones (Ec. de Continuidad)

La corrección de presiones derivada de la ecuación de continuidad también experimenta ajustes debido a la presencia de la pared. Dado que el inicio del desarrollo de esta ecuación se centra en las velocidades de las caras, las cuales se expresan posteriormente como correcciones que involucran las presiones, es necesario analizar la velocidad en la cara oeste. Dado que esta velocidad es conocida y nula, su corrección u'_w debe ser igualmente nula. Por consiguiente, la expresión 2.154 se anula, eliminando así estos términos de los coeficientes de enlace A_i^p . En cuanto a la fuente, el término \hat{u}_w también debe ser anulado, ya que no hay flujo a través de la pared. Los ajustes en las expresiones quedan de la siguiente manera:

$$u'_w = \left[\frac{1}{A_{O|W}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p'_W - p'_O}{2} \Delta y = 0 \quad (2.154)$$

$$A_O^p = \frac{\rho_e \Delta y^2}{2} \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] + \frac{\rho_n \Delta x^2}{2} \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] + \frac{\rho_s \Delta x^2}{2} \left[\frac{1}{A_{O|S}^Y} + \frac{1}{A_{O|O}^Y} \right] \quad (2.155)$$

$$A_W^p = 0 \quad (2.156)$$

$$S_O^p = -[(\rho_e \hat{u}_e) \Delta y + (\rho_n \hat{v}_n - \rho_s \hat{v}_s) \Delta x] \quad (2.157)$$

Actualización de velocidades

En el último paso, que implica actualizar las velocidades y presiones para la próxima iteración, el ajuste es sencillo, simplemente se debe imponer la condición de contorno de que la velocidad en la pared sea nula. Para el caso aquí presentado de la pared oeste, entonces:

$$u_w^{k+1}|_O = 0 \quad (2.158)$$

2.4.2 Pared con deslizamiento

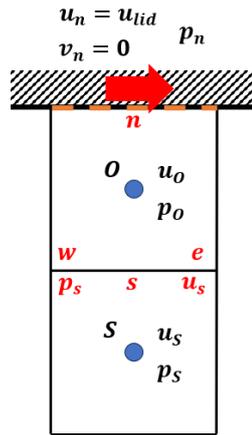


Figura 2.9 Pared superior con deslizamiento.

La siguiente condición de contorno que se desea analizar es un caso particular en el que la pared tiene deslizamiento. Esto significa que la pared se mueve de manera paralela al fluido, como se ilustra en la figura 2.9. Aunque no es una condición común, es importante estudiarla, ya que en los dominios posteriores planteados en este estudio, esta condición se presenta en la resolución de problemas típicos de cavitación.

Lógicamente, si la pared se desliza paralelamente al fluido con una velocidad u_{tid} , este movimiento se transmitirá al fluido en contacto con la superficie debido a los efectos viscosos. Es importante tener en cuenta que la velocidad vertical v_n sigue siendo cero, ya que de lo contrario implicaría un movimiento de la pared y, por consiguiente, un cambio en el volumen, lo cual no es el caso.

Ecuación de cantidad de movimiento

Al analizar los flujos J_n^X y J_n^Y , se observa cómo efectivamente quedan los flujos difusivos debido a los efectos viscosos. Aprovechando la estimación del gradiente de velocidades realizada anteriormente, se observa que para el eje X aparece un nuevo término que incluye la velocidad de la pared, mientras que para el eje Y , los términos son los mismos que para una pared sin deslizamiento.

$$J_n^X = \rho_n v_n u_n - \mu_n \frac{\partial u}{\partial y} \Big|_n = -\mu_n \frac{\partial u}{\partial y} \Big|_n \approx -\mu_n \frac{9u_o - u_s - 8u_n}{3\Delta y} \quad (2.159)$$

$$J_n^Y = \rho_w v_n v_n - \mu_n \frac{\partial v}{\partial y} \Big|_n = -\mu_n \frac{\partial v}{\partial y} \Big|_n \approx -\mu_n \frac{9v_o - v_s - 8v_n}{3\Delta y} \quad (2.160)$$

$$J_n^X \approx -\mu_n \frac{9u_O - u_S - 8u_{lid}}{3\Delta y} \quad (2.161)$$

$$J_n^Y \approx -\mu_n \frac{9v_O - v_S}{3\Delta y} \quad (2.162)$$

Por lo tanto, los cambios importantes para esta condición de contorno están relacionados principalmente con el término u_{lid} , que se suma a la fuente de las presiones como una fuente más del sistema.

$$A_O^X \hat{u}_O + A_W^X \hat{u}_W + A_E^X \hat{u}_E + A_N^X \hat{u}_N + A_S^X \hat{u}_S = S_O^X + \mu_n \frac{8\Delta x}{3\Delta y} u_{lid} \quad (2.163)$$

$$A_O^X = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \left[\frac{3\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (2.164)$$

$$A_N^X = 0 \quad (2.165)$$

$$A_S^X = - \left[\frac{|(\rho v)_s| + (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} + \frac{\mu_n}{3\Delta y} \right] \Delta x \quad (2.166)$$

Interpolación de velocidades en las caras por el método PWIM

En lo que respecta a la interpolación de PWIM, no se observan cambios significativos con respecto a las expresiones utilizadas para una pared sin deslizamiento. Simplemente es necesario aplicar la condición de contorno a \hat{v}_n y tener la precaución, ya mencionada, para las celdas cercanas a la pared, donde es necesario aproximar la presión p_n utilizando las mismas expresiones que se utilizaron previamente (2.151 y 2.153).

$$\hat{v}_n = 0 \quad (2.167)$$

Corrección de presiones (Ec. de Continuidad)

Para la corrección de las presiones, se mantiene la misma dinámica que en el caso de una pared sin deslizamiento. No se ven afectadas por el término u_{lid} , por lo que su tratamiento es idéntico. Es decir, aplicando la misma lógica, el término v_n' es nulo, ya que no requiere corrección, lo que conlleva el ajuste de los términos A_O^X , A_N^X , A_O^Y y A_N^Y . Del mismo modo, debido a la ausencia de flujo desde la pared, se ajusta el término S_O^P .

Actualización de velocidades

En la actualización de las velocidades, se sigue un proceso similar. Simplemente se aplica la condición de contorno para la siguiente iteración, sin cambios significativos en comparación con el caso de una pared sin deslizamiento.

2.4.3 Entrada a velocidad uniforme

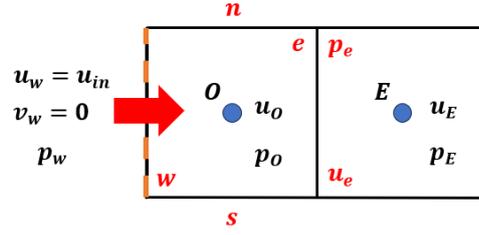


Figura 2.10 Entrada del fluido a velocidad uniforme.

La condición de contorno que sigue en este análisis es la entrada de un flujo de fluido estable con una velocidad constante u_{in} . Este escenario es común en cualquier dominio fluido con una entrada y salida definidas, como un depósito o un estudio de aerodinámica de un vehículo. Cuando se menciona "flujo estable", se hace referencia a que se está considerando que el fluido entra de manera paralela, es decir, sin componentes de velocidad perpendicular a la entrada. En la representación de la figura 2.10, la entrada se posiciona a la izquierda, donde el fluido ingresa con una velocidad u_w igual a u_{in} , mientras que la componente vertical v_w es cero.

Ecuación de cantidad de movimiento

Al igual que en el caso de una pared, los términos afectados en la ecuación de cantidad de movimiento son J_w y p_w . Dado que la velocidad v_w es nula en el eje Y , esa componente se asemeja a una pared, lo que resulta en ajustes similares en la ecuación de cantidad de movimiento para el eje Y .

El eje de mayor interés es el eje X , donde la velocidad u_{in} tiene influencia, generando una parte advectiva y otra difusiva en el flujo que se debe incorporar a las expresiones. Al reemplazar el valor de u_w , nuevamente queda también el gradiente de velocidad, que puede aproximarse utilizando la expresión previamente deducida en función de u_O , u_E y u_w .

$$J_w^X = \rho_w u_w u_w - \mu_w \left. \frac{\partial u}{\partial x} \right|_w \approx \rho_w u_w u_w - \mu_w \frac{9u_O - u_E - 8u_w}{3\Delta x} \approx \rho_{in} u_{in} u_{in} - \mu_{in} \frac{9u_O - u_E - 8u_{in}}{3\Delta x} \quad (2.168)$$

Los términos u_O y u_E se incorporan de manera similar a como se haría en una pared, modificando los coeficientes de enlace A_O^X y A_E^X . Los términos relacionados con el flujo de entrada u_{in} se trasladan al otro lado de la ecuación, junto con la fuente de presiones S_O^X . Respecto a dicha fuente, al igual que en la condición de pared, su expresión dependerá de si se considera una estimación de primer o segundo orden para la presión p_w .

$$A_O^X \hat{u}_O + A_W^X \hat{u}_W + A_E^X \hat{u}_E + A_N^X \hat{u}_N + A_S^X \hat{u}_S = S_O^X + \left[\rho_{in} u_{in} u_{in} + \frac{8\mu_{in}}{3\Delta x} u_{in} \right] \Delta y \quad (2.169)$$

$$A_O^X = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{3\mu_{in}}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (2.170)$$

$$A_E^X = - \left[\frac{|(\rho u)_e| - (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{\mu_{in}}{3\Delta x} \right] \Delta y \quad (2.171)$$

$$A_W^X = 0 \quad (2.172)$$

$$\text{Aprox. 1º Orden} \quad S_O^X = \frac{p_O - p_E}{2} \Delta y \quad (2.173)$$

$$\text{Aprox. 2º Orden} \quad S_O^X = (p_O - p_E) \Delta y \quad (2.174)$$

Interpolación de velocidades en las caras por el método PWIM

La interpolación de las velocidades en las caras resulta sencilla. Dado que la velocidad \hat{u}_w es conocida e igual a la velocidad de entrada u_{in} por la condición de contorno, no requiere ajustes adicionales. Respecto a la velocidad u_e , se pueden emplear las expresiones 2.148 y 2.149 derivadas para la pared, ya que se aplica el mismo ajuste debido a la estimación de la presión p_w .

Corrección de presiones (Ec. de Continuidad)

En los ajustes de la corrección de presión, se observa una similitud con los casos analizados anteriormente. Dado que la velocidad de entrada u_{in} es conocida, no se requieren correcciones en la velocidad u_w , lo que implica que u'_w sea nulo. Como resultado, el término A_W^p se anula y se reduce el término A_O^p , obteniendo expresiones equivalentes a las ecuaciones 2.155 y 2.156. Respecto a la fuente S_O^p , se registra un aporte constante de caudal debido a la entrada de flujo con velocidad u_{in} .

$$S_O^p = - [(\rho_e \hat{u}_e - \rho_{in} u_{in}) \Delta y + (\rho_n \hat{v}_n - \rho_s \hat{v}_s) \Delta x] \quad (2.175)$$

Actualización de velocidades

En la actualización de las velocidades, se aplica el mismo concepto. Simplemente se aplica la condición de contorno para la siguiente iteración, sin cambios significativos en comparación con el resto de casos vistos anteriormente.

$$u_w^{k+1} \Big|_O = u_{in} \quad (2.176)$$

2.4.4 Salida a presión uniforme

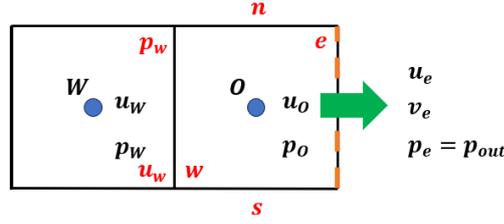


Figura 2.11 Salida del fluido a presión uniforme.

En términos generales, donde hay una entrada, es común encontrar también una salida en el dominio. Sin embargo, en los casos de salidas, el parámetro a configurar suele ser la presión, ya que típicamente las descargas se establecen a una presión conocida, ya sea requerida por un sistema o simplemente a la presión ambiente. Si durante el estudio se ha supuesto que la entrada se encontraba a la izquierda, para la salida se va a considerar que está a la derecha, tal como se muestra en la figura 2.11. Es importante destacar que la ubicación solo influye en el desarrollo de los términos, pero el proceso para ajustar las ecuaciones es el mismo en cualquiera de las cuatro caras de la celda.

Es esencial tener en cuenta que se está estableciendo un valor promedio de la presión en la salida, ya que no se conoce la distribución real de presiones. Por ello, es crucial ubicar correctamente esta condición de contorno en el dominio, garantizando que el flujo esté completamente desarrollado para evitar zonas de inversión de flujo. En muchas ocasiones, bastará con ampliar el dominio fluido de estudio para lograr este efecto.

Ecuación de cantidad de movimiento

En cuanto a las ecuaciones de cantidad de movimiento, el proceso comienza de la misma manera, revisando los términos afectados por la condición de salida, que en este caso son J_E y p_e . Ahora, dado que se conoce la presión, p_e es igual a la presión de salida p_{out} . Sin embargo, se requiere realizar una estimación para la velocidad de salida u_e .

$$(J_e^X - J_w^X) \Delta y + (J_n^X - J_s^X) \Delta x = (p_w - p_e) \Delta y \quad (2.177)$$

$$(J_e^Y - J_w^Y) \Delta y + (J_n^Y - J_s^Y) \Delta x = (p_s - p_n) \Delta x \quad (2.178)$$

Al desarrollar el término de flujo J_e^X , se identifica tanto la componente advectiva como la difusiva. Considerando que se trata de una salida, es coherente pensar que el efecto predominante será la advección sobre la difusión. Por ende, en este contexto particular, resulta lógico despreciar los esfuerzos viscosos, siempre y cuando el flujo esté completamente desarrollado.

$$J_e^X = \rho_e u_e u_e - \mu_e \left. \frac{\partial u}{\partial x} \right|_e \approx \rho_e u_e u_e \quad (2.179)$$

Respecto al valor de u_e , lo más sencillo es realizar una aproximación de primer orden y equipararlo al valor central de la celda u_o , ya que en esa mitad de celda restante, el gradiente de presión no debería ser muy pronunciado. Por consiguiente, el valor del flujo puede ser aproximado utilizando la expresión 2.180 para el eje X . En lo que respecta al eje Y , dado que se trata de un flujo desarrollado,

resulta razonable suponer que la velocidad vertical de salida v_e sea nula, lo que deja únicamente los términos difusivos, que se aproximan de manera similar a los de una pared.

$$J_e^X \approx \rho_e u_e u_e \approx \rho_O u_O u_O \tag{2.180}$$

$$J_e^Y = \rho_e v_e u_e - \mu_e \left. \frac{\partial v}{\partial x} \right|_e = -\mu_e \left. \frac{\partial v}{\partial x} \right|_e \approx -\mu_e \frac{-9v_O + v_W}{3\Delta x} \tag{2.181}$$

Al revisar estas modificaciones en el desarrollo de las ecuaciones de cantidad de movimiento y ajustar los coeficientes de enlace y fuentes, quedan expresados de la siguiente manera:

$$A_O^X = \left[\rho_O u_O + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \tag{2.182}$$

$$A_E^X = 0 \tag{2.183}$$

$$S_O^X = \left[\frac{p_O + p_W}{2} + p_{out} \right] \Delta y \tag{2.184}$$

$$A_O^Y = \left[\frac{3\mu_O}{\Delta x} + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \tag{2.185}$$

$$A_E^Y = 0 \tag{2.186}$$

$$A_W^Y = - \left[\frac{|(\rho u)_w| + (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} + \frac{\mu_O}{3\Delta x} \right] \Delta y \tag{2.187}$$

Interpolación de velocidades en las caras por el método PWIM

Sobre la interpolación de las velocidades en las caras, es crucial mantener la coherencia con la estimación realizada, manteniendo \hat{u}_e igual a \hat{u}_O para las celdas que limitan con la salida.

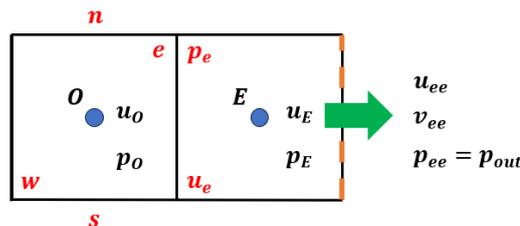


Figura 2.12 Celda cerca de la salida del fluido a presión constante.

Sin embargo, en el caso de las celdas cercanas a la salida, como se muestra en la figura 2.12, para interpolar \hat{u}_e , es necesario ajustar el gradiente de presiones de la celda E , dado que p_{ee} es igual a p_{out} , lo que conduce a la expresión 2.188.

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{\Delta y}{2 A_O^X|_E} \left[p_{out} - \frac{p_O + p_E}{2} \right] + \frac{1}{2} \frac{p_E - p_W}{2 A_O^X|_O} \Delta y - \left[\frac{1}{A_O^X|_E} + \frac{1}{A_O^X|_O} \right] \frac{p_E - p_O}{2} \Delta y \quad (2.188)$$

Corrección de presiones (Ec. de Continuidad)

Para la corrección de presiones, el proceso comienza nuevamente con la ecuación de continuidad. Aquí, se reemplaza la aproximación $\rho_e u_e$ por los valores centrales de la celda $\rho_O u_O$. Se vuelve a proceder a desarrollar la ecuación en términos de las correcciones de velocidad.

$$(\rho_O u_O - \rho_w u_w) \Delta y + (\rho_n v_n - \rho_s v_s) \Delta x = 0 \quad (2.189)$$

$$(\rho_O u'_O - \rho_w u'_w) \Delta y + (\rho_n v'_n - \rho_s v'_s) \Delta x = -[(\rho_O \hat{u}_O - \rho_w \hat{u}_w) \Delta y + (\rho_n \hat{v}_n - \rho_s \hat{v}_s) \Delta x] \quad (2.190)$$

Recordando que el término correctivo u'_O se expresa como se indica en la ecuación 2.191, y dado que la corrección p'_e es nula debido a que la presión de salida es conocida, la expresión de u'_O se puede formular en función de las correcciones p'_O y p'_W .

$$u'_O \approx \frac{\Delta y}{2 A_O^X|_O} (p'_W - p'_e) \quad (2.191)$$

$$u'_O \approx \frac{\Delta y}{2 A_O^X|_O} \left(\frac{p'_W - p'_O}{2} \right) \quad (2.192)$$

Esto lleva a la inclusión de estos aportes en los coeficientes de enlace A_O^X y A_W^X . En relación con el balance de masa, simplemente se sustituye el valor aproximado de u_e .

$$A_O^p = \frac{\rho_O \Delta y^2}{2 A_O^X|_O} + \frac{\rho_w \Delta y^2}{2} \left[\frac{1}{A_O^X|_W} + \frac{1}{A_O^X|_O} \right] + \frac{\rho_n \Delta x^2}{2} \left[\frac{1}{A_O^Y|_N} + \frac{1}{A_O^Y|_O} \right] + \frac{\rho_s \Delta x^2}{2} \left[\frac{1}{A_O^Y|_S} + \frac{1}{A_O^Y|_O} \right] \quad (2.193)$$

$$A_E^p = 0 \quad (2.194)$$

$$A_W^p = -\frac{\rho_w \Delta y^2}{2} \left[\frac{1}{A_O^X|_W} + \frac{1}{A_O^X|_O} \right] + \frac{\rho_O \Delta y^2}{2 A_O^X|_O} \quad (2.195)$$

$$S_O^p = -[(\rho_O \hat{u}_O - \rho_w \hat{u}_w) \Delta y + (\rho_n \hat{v}_n - \rho_s \hat{v}_s) \Delta x] \quad (2.196)$$

Actualización de velocidades

Finalmente, para actualizar las velocidades, se sigue manteniendo la coherencia y se aplica nuevamente la igualdad entre la velocidad de salida y la velocidad central de la celda.

$$u_e^{k+1} = u_o^{k+1} \quad (2.197)$$

3 Resolución computacional de dominios fluidos

Una vez cubiertos los fundamentos y el desarrollo de las ecuaciones que sustentan el algoritmo SIMPLE, se procederá a someterlo a prueba mediante una serie de casos en los que el dominio fluido presente una geometría simple. Siguiendo el enfoque propuesto en el desarrollo, el mallado debe ser estructurado, co-ubicado y ortogonal. Se ha optado por analizar tres casos clásicos: el flujo en un canal de sección constante, el problema de cavidad impulsada y el flujo en un canal con cambio de sección. Estos casos abarcan las diversas condiciones de contorno que se han desarrollado durante la particularización de las ecuaciones. Para todos los casos se estudiará el campo de presiones, el campo de velocidades, las líneas de corriente y los perfiles de velocidad en diferentes secciones según el número de Reynolds del dominio. Esta evaluación permitirá verificar la validez del método y del código implementado.

3.1 Flujo en canal de sección constante

3.1.1 Descripción del problema

El problema de flujo en un canal de sección constante es un aspecto fundamental en la mecánica de fluidos, que se centra en el estudio del movimiento de líquidos o gases confinados dentro de un conducto o canal cuya geometría no varía a lo largo de su longitud. Su estudio es de gran importancia debido a su amplia gama de aplicaciones en diversas áreas de la ingeniería y la física. Comprende aspectos esenciales para el diseño y la operación eficiente de sistemas de transporte de fluidos, como tuberías en sistemas de suministro de agua, redes de distribución de gas o conductos de refrigeración. Comprender cómo fluyen los fluidos en estos canales es crucial para optimizar el diseño y el rendimiento de estos sistemas además de garantizar un transporte eficaz y seguro de los recursos.

Para comenzar a aplicar el algoritmo SIMPLE y poder validar el modelo implementado, se ha optado por analizar el comportamiento de un fluido en un canal de sección constante como el que se muestra en la figura 3.1. Esto indica que tanto la forma como el área de la sección transversal, no cambian. El canal definido tiene una altura de H y una longitud de $5H$, siendo la longitud característica $H = 0.01\text{m}$. Se decide implementar una configuración clásica del problema de mecánica de fluidos en la que se define una velocidad uniforme de entrada u_{in} y una presión de salida p_{out} que bien podría ser la presión atmosférica. Esto provocará el movimiento del fluido de izquierda a derecha, en cuyo avance, debido a los efectos viscosos es de esperar que los perfiles de velocidad vayan cambiando.

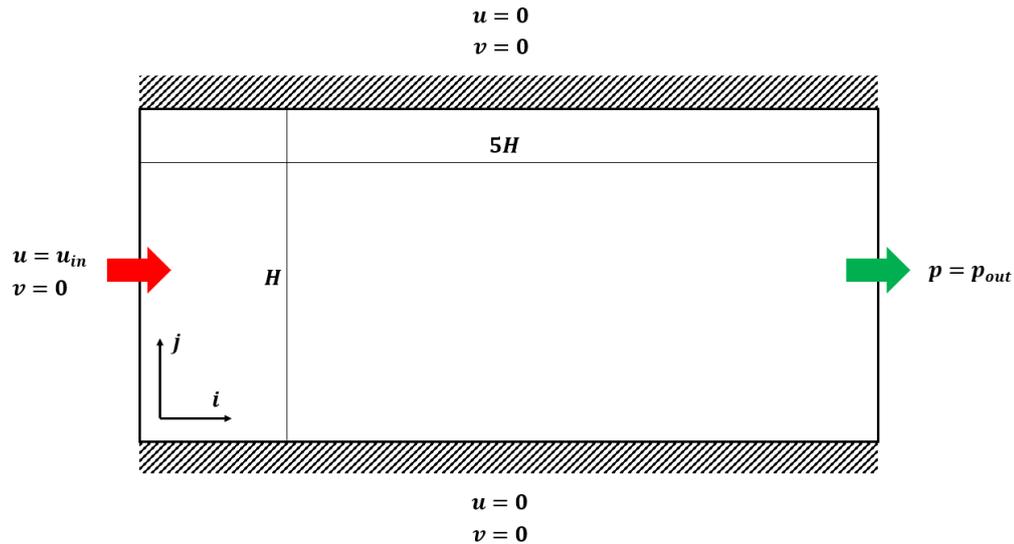


Figura 3.1 Canal de sección constante.

3.1.2 Condiciones de contorno

Revisando las condiciones de contorno del caso planteado, consta de dos paredes sin deslizamiento que son las que limitan la sección del canal. Al actuar como paredes, la velocidad del fluido que esté en contacto con la superficie deberá ser nula. El fluido entra por la cara izquierda del conducto con una velocidad uniforme u_{in} , mientras que sale por la derecha con una presión de salida uniforme p_{out} .

En lo referente al mallado del dominio fluido, se ha implementado un mallado uniforme y cúbico. La longitud característica H se ha subdividido en 20 celdas cuadradas, cada una con una longitud de 0.5 mm. Esta división hace que el dominio quede caracterizado con una matriz de 20x100 celdas.

Como ya se indicó anteriormente en la particularización de las ecuaciones, hay que tener especial cuidado con las celdas que se encuentran cercanas a la salida y a la pared superior, puesto que se deben adaptar las aproximaciones realizadas para mantener la coherencia del planteamiento. Si se revisa el código en el anexo C, lo más sencillo es recorrer las filas y columnas en distintos bucles "For" diferenciando las zonas con condiciones de contorno y las celdas interiores. De forma detallada, se representa en la figura 3.2 las diferentes zonas descritas.

Para realizar un análisis fluidodinámico más efectivo, se empleará el número de Reynolds como parámetro esencial. Esto permitirá una mejor comprensión y comparación de cómo el flujo se comporta en diferentes condiciones. Con este fin, se elegirán las propiedades del fluido de tal manera que el número de Reynolds pueda ser modificado fácilmente mediante la introducción del parámetro de velocidad de entrada u_{in} . En este contexto, se definirán la densidad del fluido como $\rho = 1\text{ kg/m}^3$ y su viscosidad como $\mu = 10^{-5}\text{ kg/(m}\cdot\text{s)}$. Además, se fijará una longitud característica del problema, $H = 0.01\text{ m}$. De esta manera, el único parámetro que variará será la velocidad de la entrada. Este último será ajustado en un rango que va desde 0.1 m/s hasta 0.4 m/s, lo que resultará en un intervalo de valores de Re comprendido entre 100 y 400.

$$Re = \frac{\rho u_{in} H}{\mu} \quad (3.1)$$

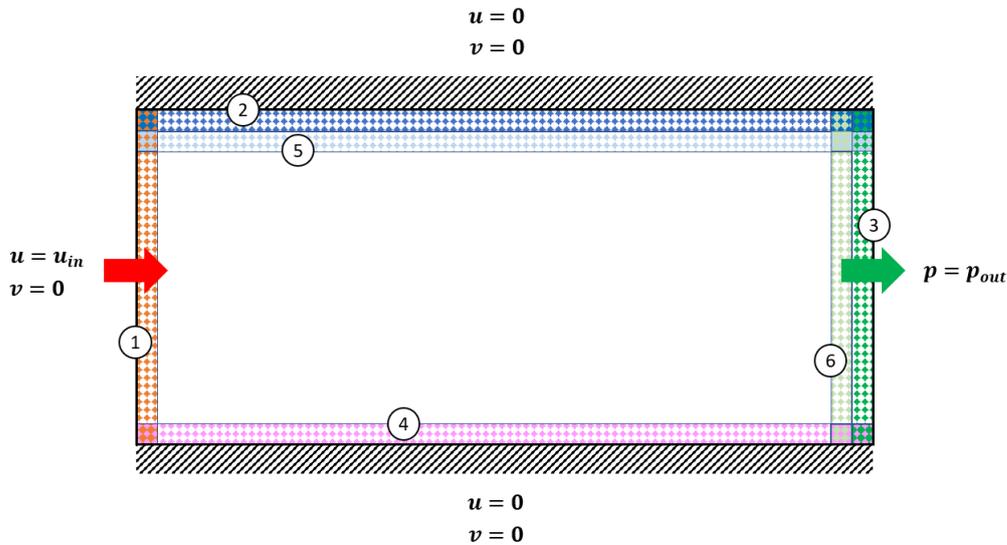


Figura 3.2 Condiciones de contorno del canal de sección constante.

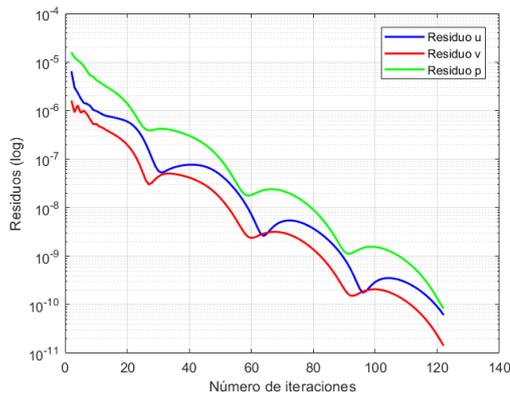
- Zona 1: Entrada de velocidad uniforme
- Zona 2: Pared superior sin deslizamiento
- Zona 3: Salida de presión uniforme
- Zona 4: Pared inferior sin deslizamiento
- Zona 5: Celdas cercanas a pared superior
- Zona 6: Celdas cercanas a la salida a presión uniforme.

3.1.3 Resultados y análisis

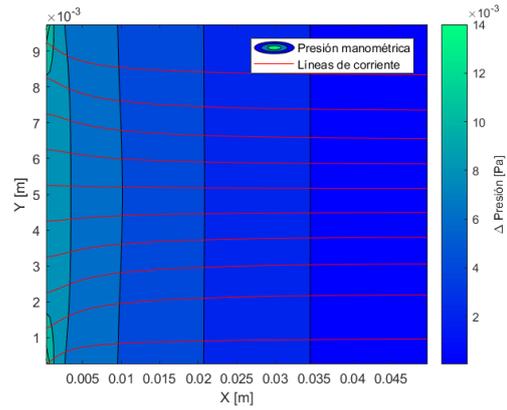
En esta sección, se examinará la evolución del dominio fluido en relación con tres aspectos fundamentales: el campo de presiones, el campo de velocidades y los perfiles de velocidad, en función de la variación del número de Reynolds. Para poder validar el modelo implementado, es de esperar que a medida que aumente el Reynolds, los efectos viscosos sean menores y por tanto sus efectos sobre los perfiles de velocidad serán menos pronunciados.

Para investigar este fenómeno, se realiza un análisis detallado de los perfiles de velocidad horizontal en distintas ubicaciones a lo largo del canal, partiendo desde su entrada. Este enfoque permitirá observar cómo la viscosidad del fluido influye en la distribución y la forma de los perfiles de velocidad a lo largo del recorrido del flujo, esperando obtener la clásica forma de campana cuando el flujo está completamente desarrollado.

Otro aspecto crucial para validar el modelo, será la evolución de los residuos. Es fundamental verificar que el algoritmo utilizado para el análisis sea capaz de converger hacia una solución precisa y coherente. Para ello, se van a monitorear de cerca la evolución de los residuos a lo largo del proceso de simulación. La convergencia adecuada de estos residuos proporcionará una indicación confiable de que el algoritmo está progresando hacia una solución correcta y estable. Esta evaluación de la convergencia no solo asegurará la fiabilidad de los resultados obtenidos, sino que también permitirá comprender mejor la calidad y la precisión del modelo utilizado en este estudio.

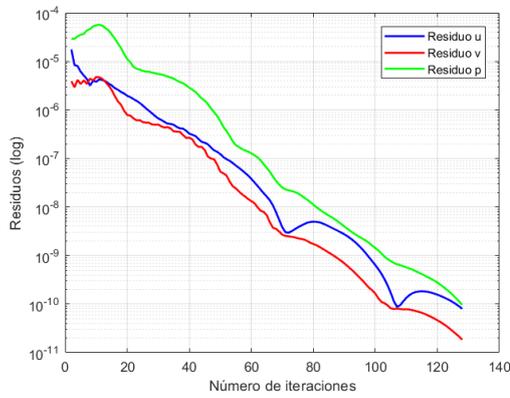


(a) Evolución de los residuos

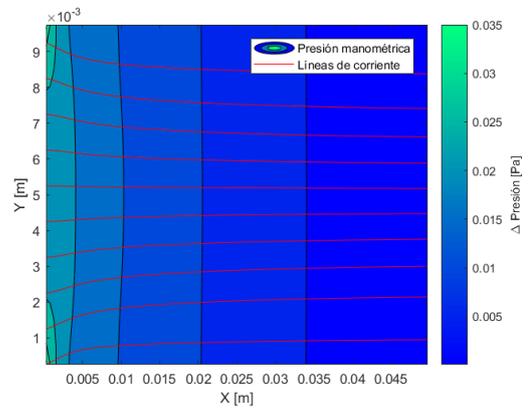


(b) Campo de presiones y líneas de corriente

Figura 3.3 Residuos y campo de presiones $Re = 100$.

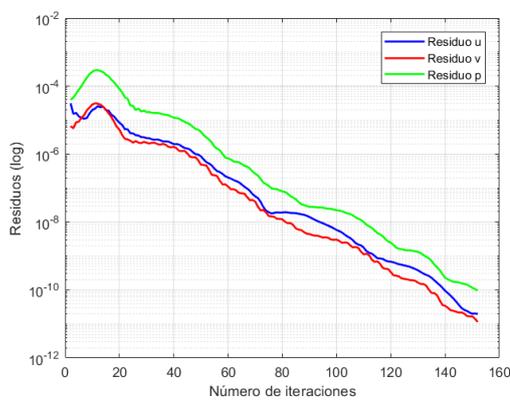


(a) Evolución de los residuos

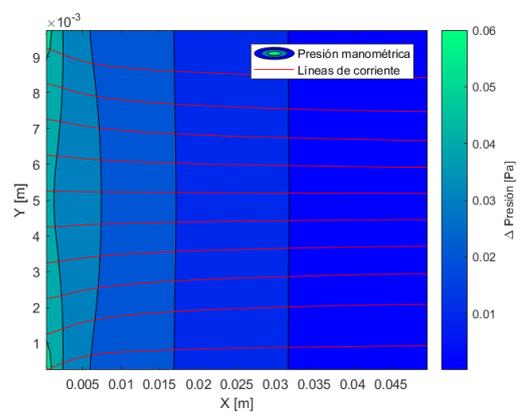


(b) Campo de presiones y líneas de corriente

Figura 3.4 Residuos y campo de presiones $Re = 200$.

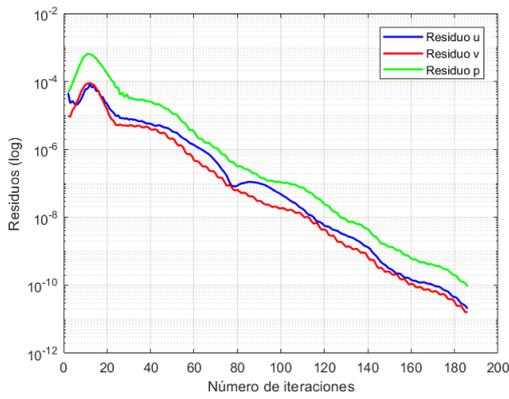


(a) Evolución de los residuos

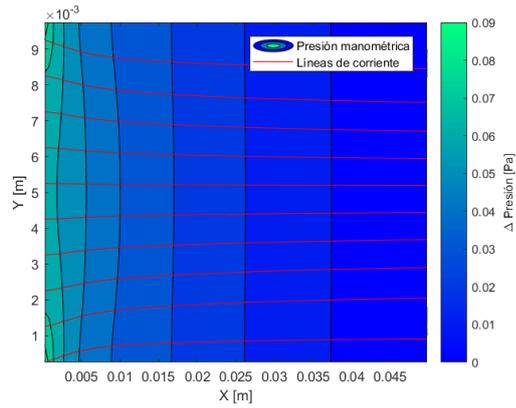


(b) Campo de presiones y líneas de corriente

Figura 3.5 Residuos y campo de presiones $Re = 300$.

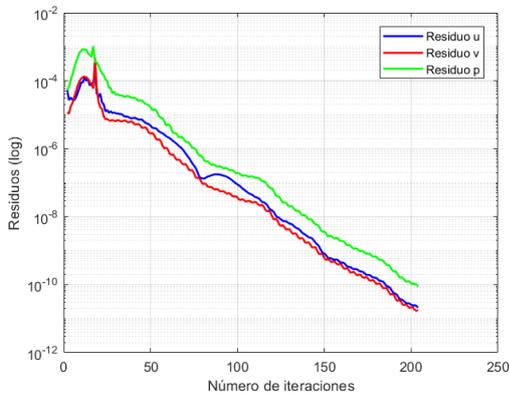


(a) Evolución de los residuos

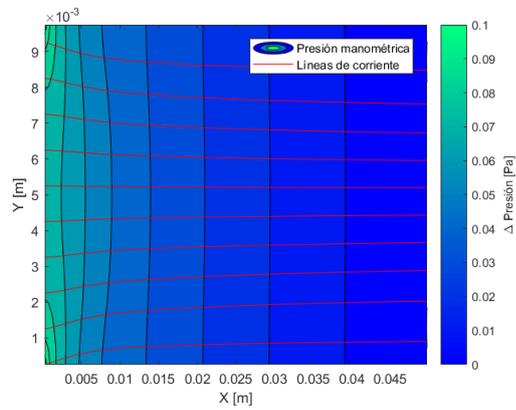


(b) Campo de presiones y líneas de corriente

Figura 3.6 Residuos y campo de presiones $Re = 400$.



(a) Evolución de los residuos



(b) Campo de presiones y líneas de corriente

Figura 3.7 Residuos y campo de presiones $Re = 450$.

Tabla 3.1 Canal longitudinal constante: iteraciones y residuos en función de Re .

Re	Iteraciones	Residuo u	Residuo v	Residuo p
100	122	6,14E-11	1,43E-11	8,39E-11
200	128	8,01E-11	1,84E-11	9,66E-11
300	152	1,98E-11	1,15E-11	9,59E-11
400	186	2,04E-11	1,68E-11	9,27E-11
450	204	2,10E-11	1,79E-11	8,78E-11

Al analizar los residuos de todos los casos, se observa que descienden por debajo de la tolerancia establecida de 10^{-11} , lo que indica una convergencia satisfactoria en todos los resultados y en ningún caso se llega al máximo de iteraciones. Además, todos los residuos alcanzados, son del mismo orden. Inicialmente, los residuos experimentan un aumento hasta transcurrir unas cuantas iteraciones, momento en el cual los resultados comienzan a converger y, por ende, los residuos empiezan a disminuir.

Como se refleja en los gráficos, la ecuación que presenta mayor dificultad para alcanzar la convergencia es la corrección de las presiones, que, como se recuerda, está relacionada con la ecuación de continuidad. Es probable que esto se deba a que es la ecuación que hace de nexo entre las dos ecuaciones de cantidad de movimiento. Durante las simulaciones, se evidenció que esta ecuación es un factor crítico para lograr la convergencia, ya que frecuentemente se detectó como la primera en no converger. Este contratiempo desencadenaba un efecto dominó en el resto de variables, generando resultados insatisfactorios. Se llegó a la conclusión de que es fundamental resolver con precisión la corrección de las presiones, pues su resolución inexacta afecta significativamente la estabilidad del proceso y a la validez de los resultados obtenidos en la simulación.

A medida que el número de Reynolds aumenta, se observa cómo las curvas de los residuos muestran una mayor fluctuación, atribuible al aumento de la turbulencia del flujo y, por tanto, a mayores variaciones. Esto se refleja en los campos de presión, donde debido a esta turbulencia, los gradientes de presión entre diferentes zonas del dominio se incrementan, resultando en una mayor división del dominio. Numéricamente, se aprecia que, por ejemplo, en el caso de $Re = 100$, la diferencia de presión entre la entrada y la salida es de aproximadamente $\Delta p = 0.01$ Pa, mientras que para el último caso de $Re = 450$, asciende a $\Delta p = 0.08$ Pa. Es natural que esto afecte el número de iteraciones necesarias para alcanzar la convergencia, como se refleja en la tabla 3.1, pasando de 122 iteraciones hasta 204 para el último caso, lo que se traduce en un mayor tiempo de computación. A pesar de esto, en todos los casos, las líneas de corriente son prácticamente rectas, lo que indica un flujo laminar sin que se mezclen las capas de fluido, excepto en la entrada que son algo más curvadas hacia el centro.

En las esquinas de la entrada, como era de esperar, se aprecia un fuerte gradiente de presiones debido a que el fluido debe frenarse por el contacto con las paredes debido a los esfuerzos viscosos. Para mantener el caudal, la presión debe aumentar para incrementar la velocidad del fluido de las capas adyacentes, lo que resulta en una aceleración del fluido en su parte central, adquiriendo la forma característica de campana, como se puede observar en las figuras 3.39, 3.41, 3.42, 3.11 y 3.12.

Efectivamente, al revisar los campos de velocidades, se observa que los vectores de velocidad cercanos a la entrada poseen una componente vertical que dirige el fluido desde las paredes hacia el centro de la sección. A medida que el fluido avanza por el canal, la velocidad comienza a aumentar hasta que el flujo se desarrolla completamente. Para comprender mejor este fenómeno, se presentan los perfiles de velocidad horizontal tomados en diferentes secciones y se comparan con el perfil uniforme en la entrada y el perfil presentado en la salida. En todos los casos, se aprecia el desarrollo del flujo y el incremento progresivo de la velocidad desde las paredes hacia el centro. Inicialmente, la velocidad se incrementa cerca de las paredes debido al gradiente de presión en las esquinas de la entrada, que impulsa el fluido hacia el centro, mientras que la parte central ve aumentada su velocidad debido a los efectos difusivos de esas líneas de corriente, hasta que finalmente las líneas se unen en el centro e impulsan la línea central del flujo.

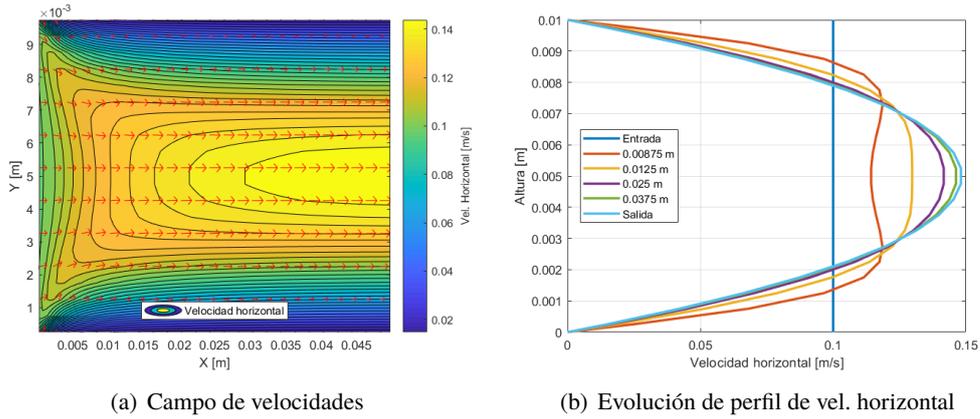


Figura 3.8 Campo de velocidades y perfiles, $Re = 100$.

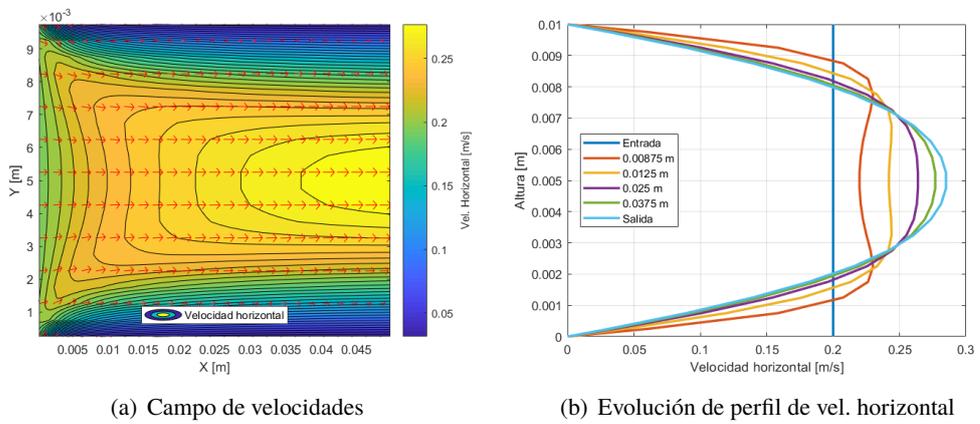


Figura 3.9 Campo de velocidades y perfiles, $Re = 200$.

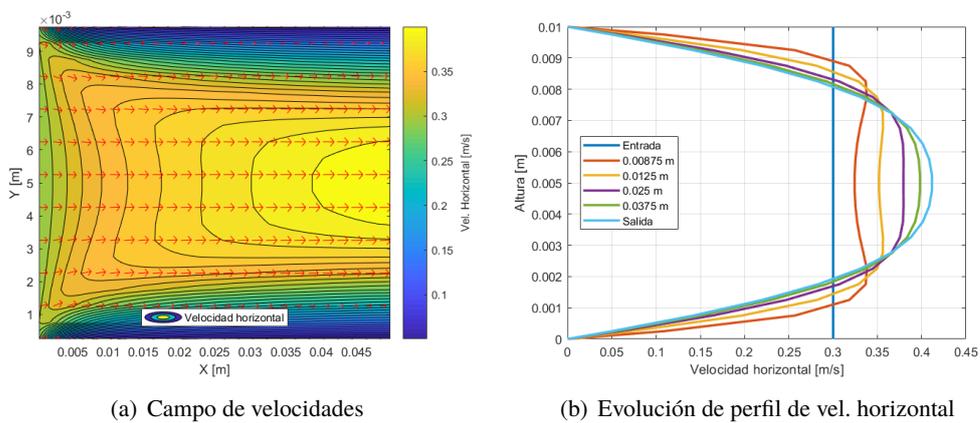
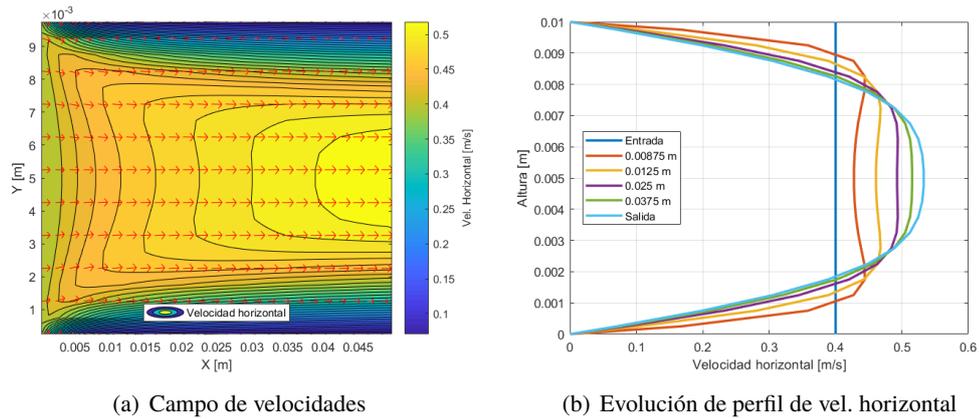
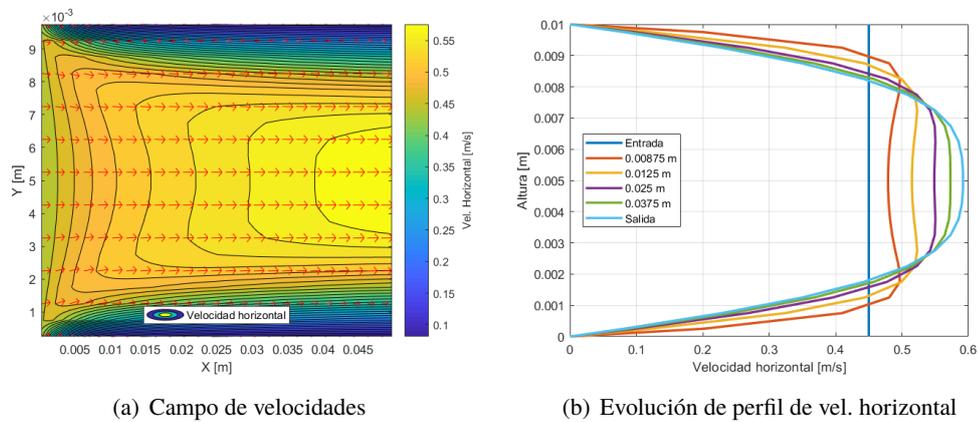


Figura 3.10 Campo de velocidades y perfiles, $Re = 300$.



(a) Campo de velocidades

(b) Evolución de perfil de vel. horizontal

Figura 3.11 Campo de velocidades y perfiles, $Re = 400$.

(a) Campo de velocidades

(b) Evolución de perfil de vel. horizontal

Figura 3.12 Campo de velocidades y perfiles, $Re = 450$.

Si se comparan los perfiles de velocidad, se aprecia que a medida que la turbulencia en el flujo aumenta, las fuerzas inerciales predominan sobre las fuerzas viscosas. Este desequilibrio implica que el flujo requiere más tiempo para desarrollarse a lo largo del canal. Por lo tanto, resulta evidente que el flujo con un número de Reynolds de $Re = 100$ se encuentra prácticamente desarrollado en la mitad del dominio, mientras que para $Re = 450$, el desarrollo del flujo se extiende a lo largo de casi toda la longitud del dominio hasta la salida. Este fenómeno se ilustra claramente en las figuras 3.13, 3.14 y 3.15, donde se comparan los perfiles de velocidad horizontal en diferentes secciones del canal ($x = 0.0175$ m, $x = 0.025$ m y $x = 0.0375$ m). Es notable cómo en los flujos que aún no están completamente desarrollados, la parte central del perfil de velocidad es casi uniforme, hasta que los efectos viscosos alteran la distribución, llevándola a una forma parabólica.

Este fenómeno, también explica las fluctuaciones de los residuos al aumentar el Reynolds y el límite encontrado en $Re = 460$. Este límite en el Re encontrado durante las simulaciones se podría explicar porque con el dominio planteado, el flujo no termina de desarrollarse por completo, y por lo tanto el problema no converge. Para poder solucionarlo, sería necesario ampliar la longitud del canal y mallar el dominio con celdas más pequeñas, aunque esto conllevaría a un aumento significativo del tiempo de simulación.

En términos generales, las soluciones obtenidas en cuanto al campo de presiones, líneas de corriente y campo de velocidades muestran coherencia y concuerdan con los fenómenos esperados en la realidad. Por lo tanto, a primera vista, el algoritmo parece funcionar de acuerdo con las expectativas establecidas.

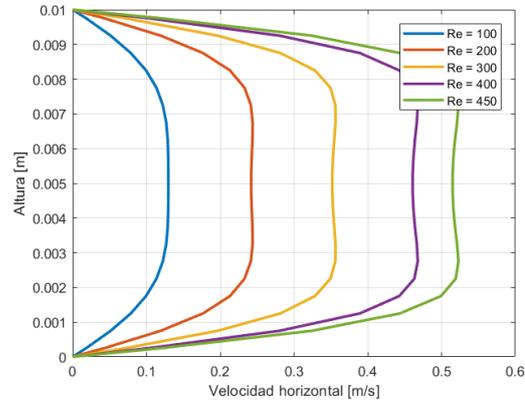


Figura 3.13 Comparación de perfiles de velocidad horizontal, $x = 0.0175\text{ m}$.

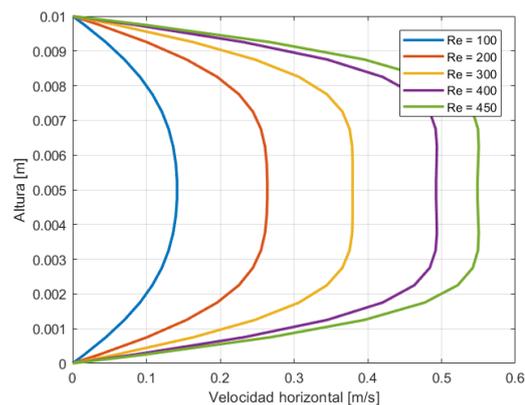


Figura 3.14 Comparación de perfiles de velocidad horizontal, $x = 0.025\text{ m}$.

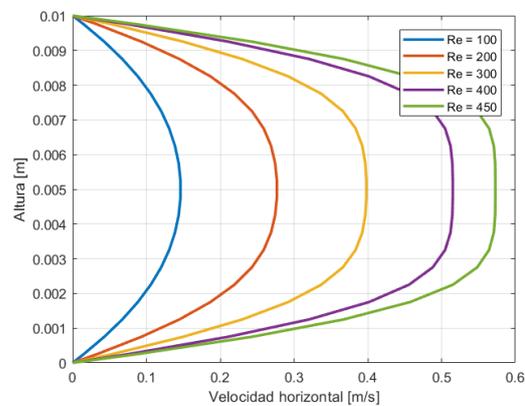


Figura 3.15 Comparación de perfiles de velocidad horizontal, $x = 0.0375\text{ m}$.

3.2 Problema de cavidad impulsada

3.2.1 Descripción del problema

El problema de la cavidad impulsada (driven cavity problem) es un problema clásico en el campo de la mecánica de fluidos que implica el estudio del comportamiento de un fluido confinado en una cavidad rectangular que se encuentra en movimiento debido a una fuerza impulsora externa. La velocidad y la dirección de esta fuerza impulsora pueden variar dependiendo del contexto del problema específico que se esté estudiando.

Este problema es de gran importancia tanto a nivel teórico como a nivel práctico en la mecánica de fluidos, debido a sus diversas aplicaciones y a las complejidades que presenta en la comprensión del flujo en condiciones de confinamiento y movimiento. Algunos de los aspectos más interesantes que permite estudiar este tipo de problemas son:

- Comprender los flujos confinados: proporciona un medio para comprender cómo se comporta un fluido cuando está confinado en un espacio limitado y se encuentra en movimiento, por ejemplo, el flujo de lubricantes en maquinaria industrial.
- Validación de modelos matemáticos y numéricos: Al tratarse de una geometría simple con unas condiciones de contorno sencillas, este problema es un clásico banco de pruebas importante para validar modelos matemáticos y numéricos utilizados en la simulación de flujo. La bibliografía existente sobre este problema es muy extensa, lo que permite comparar los resultados con cualquier modelo desarrollado. La capacidad para predecir con precisión el comportamiento del flujo en una cavidad impulsada es crucial para garantizar la exactitud de los modelos utilizados en aplicaciones más complejas.
- Optimización de diseños: El estudio del flujo en una cavidad impulsada puede ayudar en la optimización de diseños para una variedad de aplicaciones, como, la refrigeración de equipos electrónicos o la mezcla de fluidos en reactores químicos. Comprender cómo cambian las condiciones de flujo dentro de la cavidad bajo diferentes configuraciones y condiciones puede llevar a mejoras significativas en el rendimiento y la eficiencia de los sistemas.

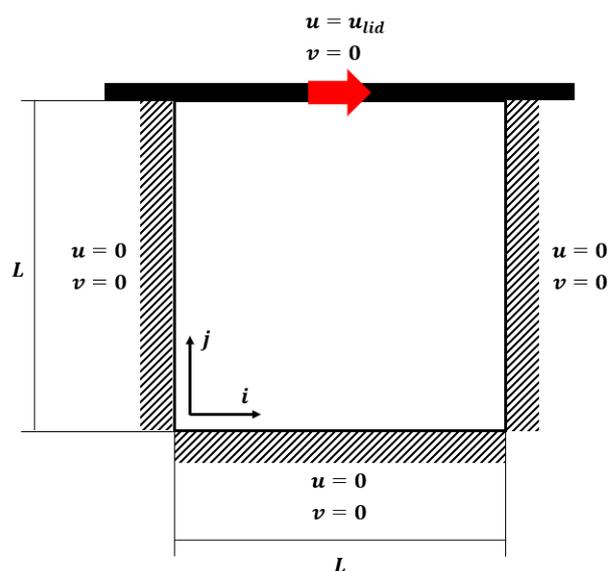


Figura 3.16 Problema de cavidad impulsada.

En el caso de este estudio, tal como se muestra en la figura 3.16, el problema seleccionado se trata de una cavidad cuadrada de lado $L = 0.01m$ cuya tapa superior se mueve con a una velocidad constante u_{lid} , mientras que el resto de superficies actúan como paredes que confinan el fluido. Como es lógico, debido a este movimiento de la tapa, el movimiento será transferido al fluido por los efectos viscosos, provocando su recirculación. Se estudiará como varía esta recirculación en función del número de Reynolds.

3.2.2 Condiciones de contorno

Respecto a las condiciones de contorno del problema, consta de tres paredes sin deslizamiento donde el fluido que esté en contacto con dicha superficie, verá anulada su velocidad. En cuanto a la tapa superior, se mueve con velocidad horizontal u_{lid} que será la que provoque la recirculación del fluido, sin embargo, su velocidad vertical v , será nula, pues de no serlo, implicaría un cambio en el volumen del dominio fluido. Adicionalmente, hay que tener especial cuidado con las celdas que se encuentran cercanas a las paredes Este y Norte, de manera que habrá que tenerlas en cuenta para aplicar las particularidades correspondientes, pues las aproximaciones de las presiones influyen en la interpolación de las velocidades en las caras. De forma detallada, se representa en la figura 3.17 las diferentes zonas descritas:

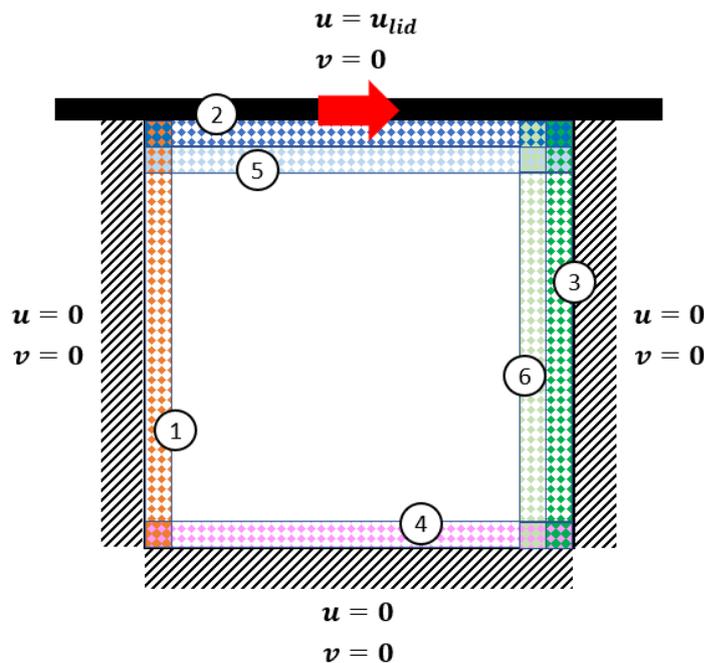


Figura 3.17 Condiciones de contorno del problema de cavidad impulsada.

- Zona 1: Pared izquierda sin deslizamiento
- Zona 2: Pared superior con deslizamiento
- Zona 3: Pared derecha sin deslizamiento
- Zona 4: Pared inferior sin deslizamiento
- Zona 5: Celdas cercanas a pared superior
- Zona 6: Celdas cercanas a pared derecha

En el mallado del dominio fluido, se ha implementado una malla uniforme y co-ubicada, formada por una matriz de 40x40 celdas. Es decir, cada celda tiene una longitud de 0.25 mm.

Para comenzar el proceso iterativo, en esta ocasión se decide empezar desde el arranque del sistema, es decir, como no se esperan que haya grandes gradientes de presión como en el caso de una entrada, parece razonable empezar con velocidad y presiones nulas. De esta manera, en la primera iteración, el desequilibrio del sistema vendrá por los efectos viscosos debido a la velocidad que impulsa a las primeras capas del fluido que están en contacto con la tapa superior. Es de esperar que esta velocidad junto con los efectos difusivos, vayan “arrastrando” a las partículas del resto de capas, iniciando así la rotación.

El estudio del campo fluido se llevará a cabo considerando el número de Reynolds como parámetro fundamental. Para facilitar esta tarea, se seleccionarán las propiedades del fluido de tal manera que el número de Reynolds pueda ser modificado simplemente mediante la introducción del parámetro de velocidad u_{lid} . En este contexto, la densidad del fluido se establecerá como $\rho = 1000 \text{ kg/m}^3$ y su viscosidad como $\mu = 10^{-3} \text{ kg/(m}\cdot\text{s)}$. La longitud característica del problema se fijará en $L = 0.01 \text{ m}$, dejando únicamente el parámetro de velocidad de la pared como variable. Este último será ajustado en un rango que va desde 0.01 m/s hasta 0.2 m/s, lo que resultará en un intervalo de valores de Re comprendido entre 100 y 2000.

$$Re = \frac{\rho u_{lid} L}{\mu} \quad (3.2)$$

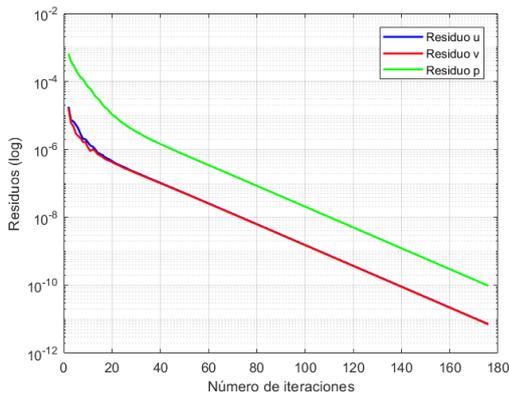
3.2.3 Resultados y análisis

Al igual que en el caso anterior, se examinará el campo fluido desde tres enfoques: el campo de presiones, el campo de velocidades y los perfiles de velocidad en relación con el número de Reynolds. Para validar el código implementado, se anticipa que la recirculación del fluido será más pronunciada a medida que aumente el número de Reynolds, indicando un flujo más turbulento. Esto sugiere que, cuanto menor sea la viscosidad del fluido, las velocidades de rotación de la recirculación serán mayores.

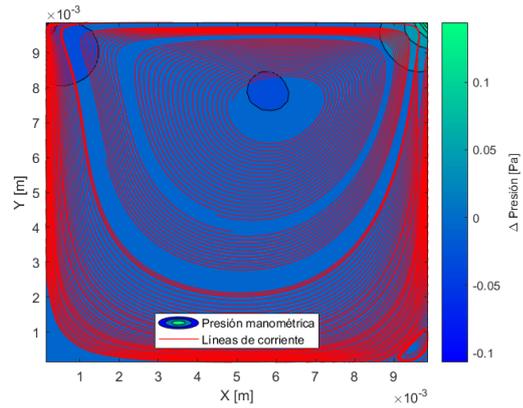
Una manera de confirmar esto será analizar el perfil de velocidades horizontales en la sección central de la cavidad, lo que permitirá evaluar la relación entre los efectos viscosos y las fuerzas inerciales debido a la recirculación del fluido. Se espera observar que, con el incremento del número de Reynolds, las velocidades en la parte inferior de la cavidad aumenten. Además, se llevará a cabo un monitoreo de los residuos para garantizar que el algoritmo converja hacia una solución adecuada.

Tabla 3.2 Cavidad impulsada: iteraciones y residuos en función de Re .

Re	Iteraciones	Residuo u	Residuo v	Residuo p
100	176	7,22E-12	7,10E-12	9,70E-11
200	192	4,68E-12	4,73E-12	9,87E-11
500	169	3,39E-12	3,56E-12	9,51E-11
1000	177	2,93E-12	3,65E-12	9,56E-11
1500	196	2,73E-12	3,39E-12	9,53E-11
2000	211	2,65E-12	3,27E-12	9,29E-11

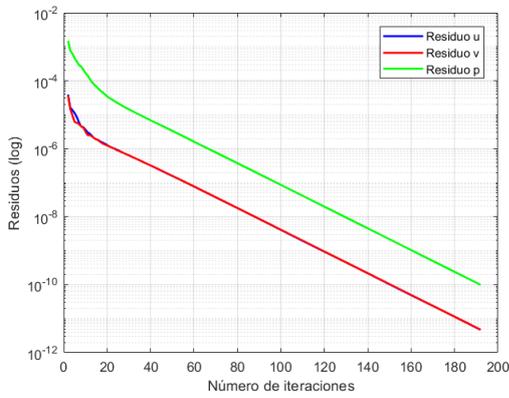


(a) Evolución de los residuos

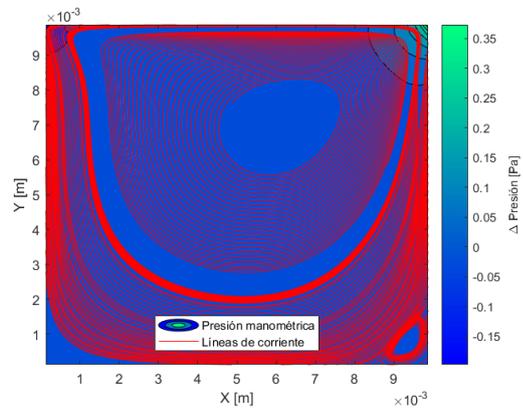


(b) Campo de presiones y líneas de corriente

Figura 3.18 Residuos y líneas de corriente $Re = 100$.

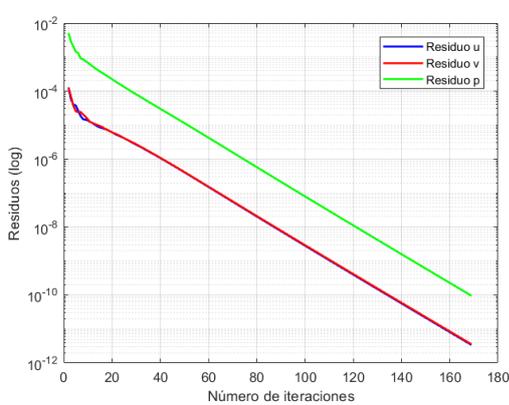


(a) Evolución de los residuos

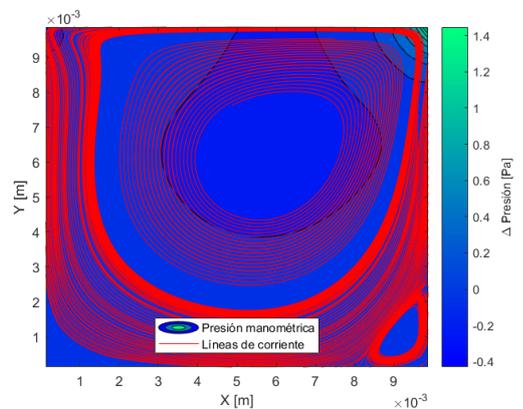


(b) Campo de presiones y líneas de corriente

Figura 3.19 Residuos y líneas de corriente $Re = 200$.

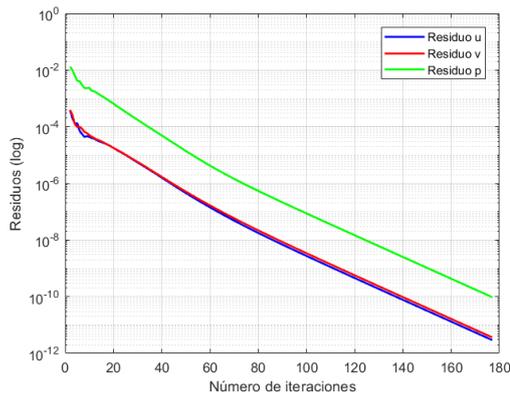


(a) Evolución de los residuos

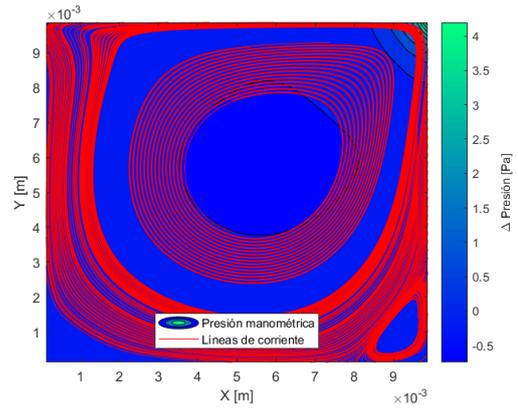


(b) Campo de presiones y líneas de corriente

Figura 3.20 Residuos y líneas de corriente $Re = 500$.

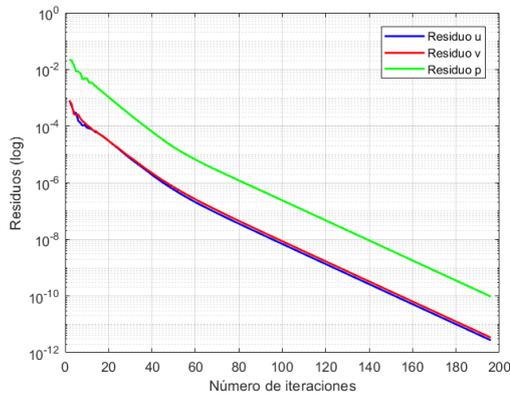


(a) Evolución de los residuos

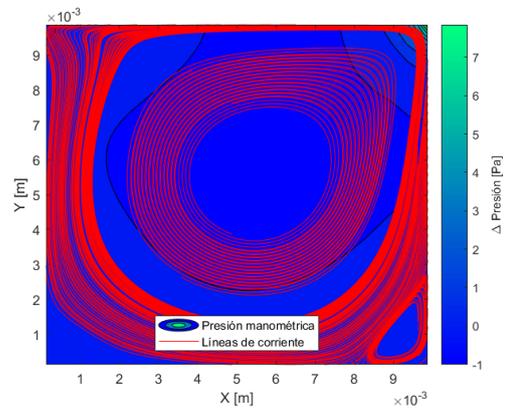


(b) Campo de presiones y líneas de corriente

Figura 3.21 Residuos y líneas de corriente $Re = 1000$.

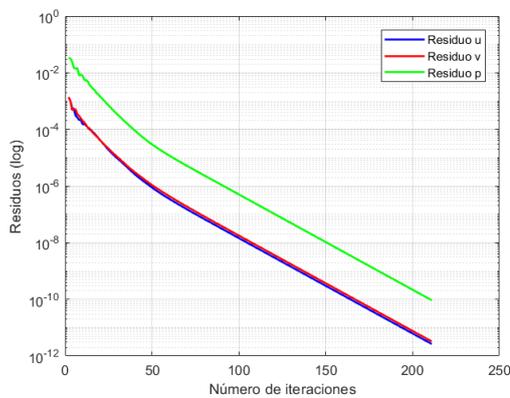


(a) Evolución de los residuos

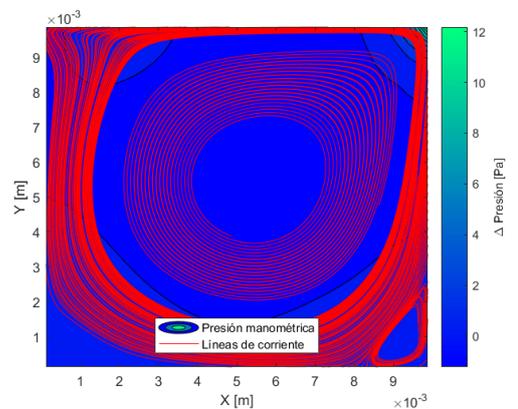


(b) Campo de presiones y líneas de corriente

Figura 3.22 Residuos y líneas de corriente $Re = 1500$.



(a) Evolución de los residuos



(b) Campo de presiones y líneas de corriente

Figura 3.23 Residuos y líneas de corriente $Re = 2000$.

Las gráficas de los residuos obtenidos revelan la notable estabilidad del problema de cavidad impulsada. Las fluctuaciones en las curvas de los residuos son mínimas y la convergencia es rápida, lo que sugiere una mayor estabilidad y consistencia en comparación con otros problemas. Este fenómeno se atribuye probablemente a las condiciones de contorno del propio problema, donde no hay entrada ni salida de flujo, lo que proporciona una estabilidad inherente. Además, se observa que es posible utilizar números de Reynolds muy altos, como $Re = 2000$, sin comprometer significativamente la estabilidad del sistema. Al analizar diferentes casos con variaciones en el número de Reynolds, se encuentra que el número de iteraciones necesarias para la convergencia, aunque es mayor al aumentar Re , sigue siendo del mismo orden de magnitud. Esto se traduce en tiempos de simulación muy similares entre todos los casos.

Una vez más, la ecuación de continuidad es la que presenta mayor dificultad para la convergencia, no obstante, presenta una pendiente igual a las de los residuos de las ecuaciones de cantidad de movimiento, con una diferencia en los resultados de un orden de magnitud respecto a estos. Este hecho, sigue reforzando la evidencia de que la ecuación de continuidad es la clave para la estabilidad de la resolución numérica y la convergencia del problema, ya que es el nexo entre el resto de variables.

En cuanto a los campos de presiones, se pueden identificar cuatro zonas distintas. Una zona de sobrepresión que se localiza en la esquina superior derecha, una zona de depresión en la esquina superior izquierda, una zona central también con menos presión que el resto del entorno y la última zona es el resto del entorno con una presión muy uniforme.

Dado que la velocidad de impulso de la tapa es de izquierda a derecha, eso hace que el flujo gire en el sentido de las agujas del reloj, por tanto, cuando el fluido de las capas superiores llega a la pared derecha, este impacta, generando una sobrepresión sobre el resto del campo haciendo que el resto de líneas de corriente se desvíen hacia abajo, comenzando así la rotación. El caso contrario se observa en la esquina opuesta, debido a esa aceleración que ve el fluido en dicha esquina, se produce un sumidero que el fluido tiende a querer rellenar, de ahí que también se desvíen las líneas de corriente hacia la esquina hasta que los efectos viscosos aceleran dichas partículas hacia la derecha. Ambos efectos se ven amplificadas cuanto mayor es la velocidad en la tapa superior. En el caso de la sobrepresión, se pasa, para el caso de $Re = 100$, de un incremento de presión de $\Delta p = 0.17$ Pa sobre la presión relativa hasta $\Delta p = 13.5$ Pa para $Re = 2000$. Lo mismo ocurre para el sumidero de la esquina superior derecha, donde se obtienen valores de $\Delta p = -0.1$ Pa para $Re = 100$ y llega hasta $\Delta p = -1.22$ Pa para $Re = 2000$.

La rotación generada en el problema de cavidad impulsada, parece que también conlleva la formación de una disminución de la presión, especialmente evidente en las proximidades del centro de esta rotación. Este efecto guarda similitudes con la dinámica del ojo de un huracán, donde la rotación impulsa las partículas fluidas hacia afuera debido a la fuerza centrífuga, creando así una zona de bajas presiones.

Otro fenómeno que se aprecia en las figuras es que, al alcanzar las esquinas inferiores, el flujo se desprende de las paredes, dando origen a zonas secundarias de recirculación en las partes bajas del dominio, próximas a las esquinas. Estas rotaciones secundarias, influenciadas por los efectos viscosos, adoptan un sentido opuesto a la rotación principal de la cavidad, girando en sentido contrario a las agujas del reloj. La magnitud y fuerza de estas rotaciones secundarias aumentan con el número de Reynolds, ya que el flujo se torna más turbulento con este incremento.

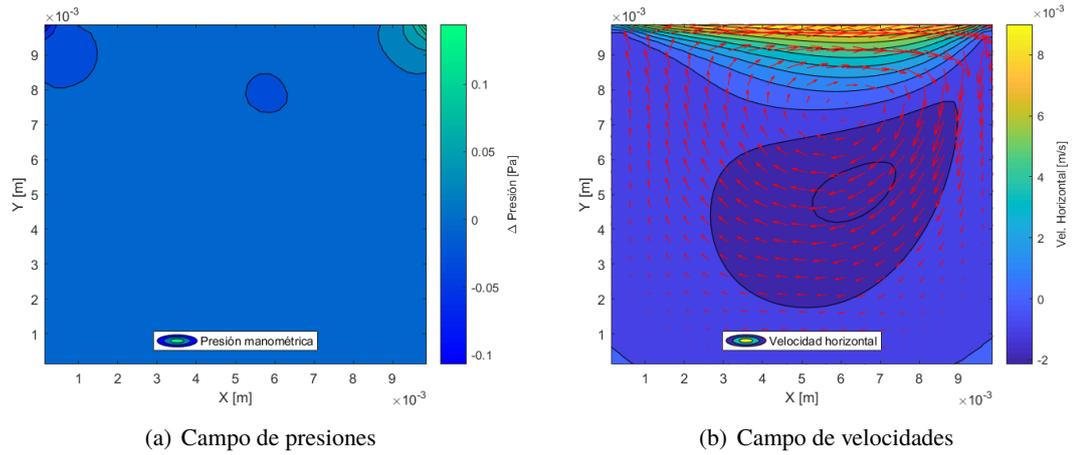


Figura 3.24 Campo de presiones y velocidades $Re = 100$.

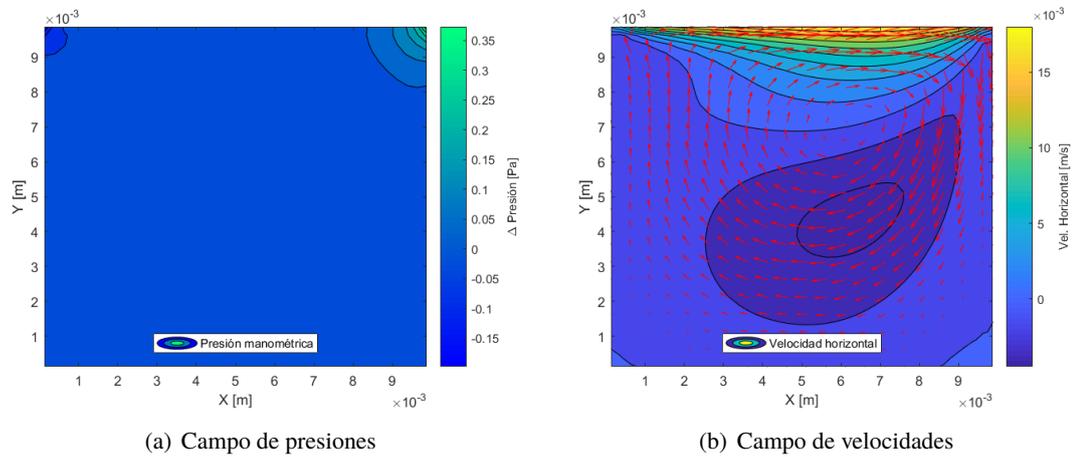


Figura 3.25 Campo de presiones y velocidades $Re = 200$.

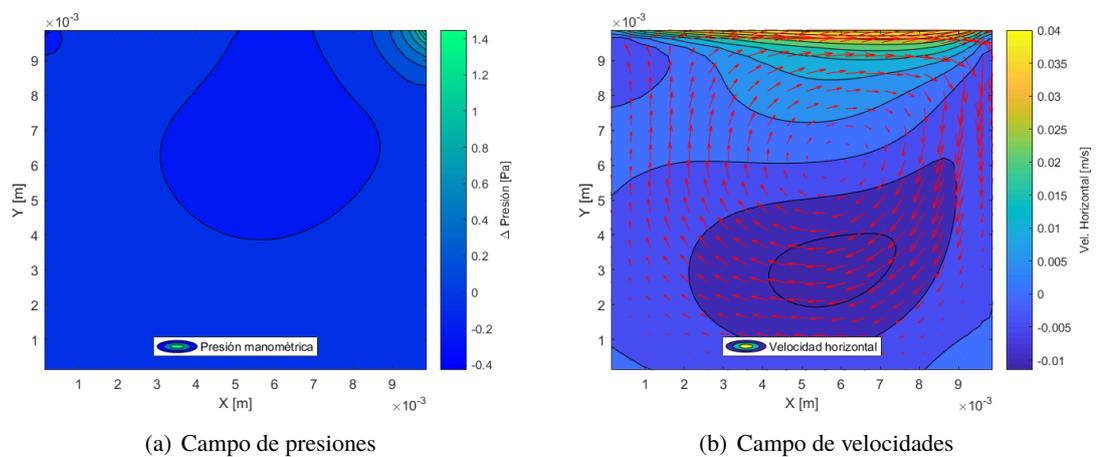


Figura 3.26 Campo de presiones y velocidades $Re = 500$.

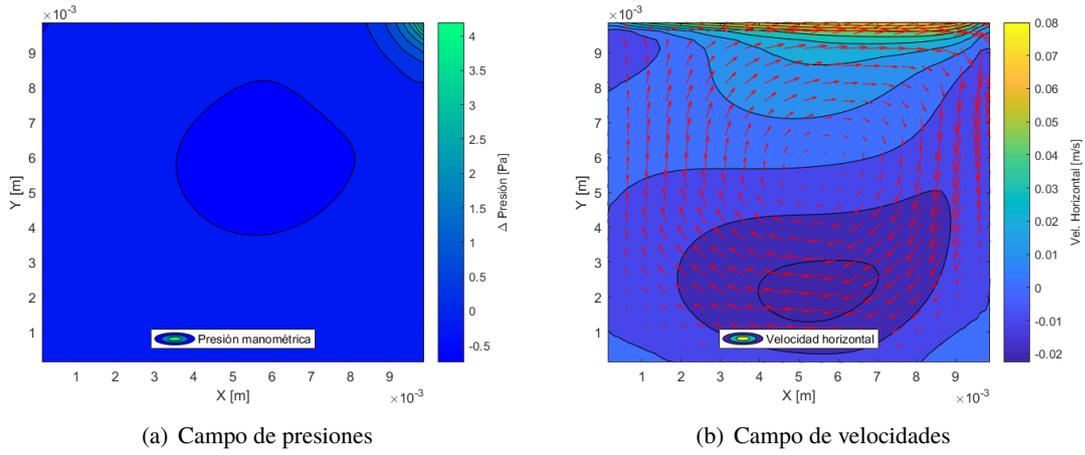


Figura 3.27 Campo de presiones y velocidades $Re = 1000$.

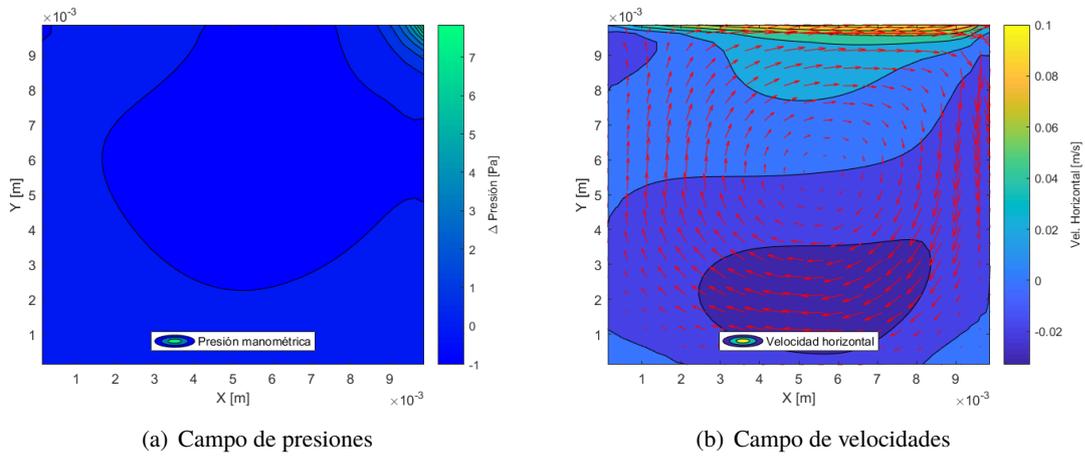


Figura 3.28 Campo de presiones y velocidades $Re = 1500$.

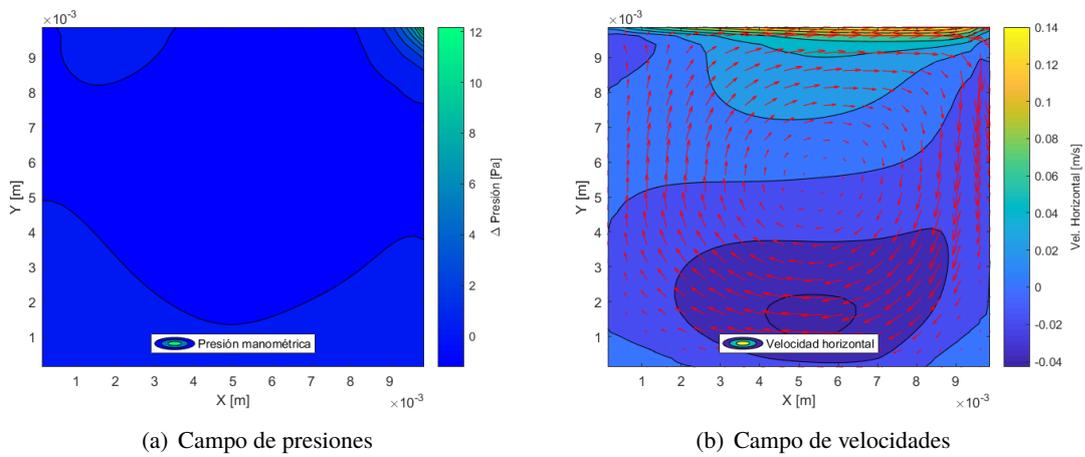


Figura 3.29 Campo de presiones y velocidades $Re = 2000$.

Revisando el campo de velocidades horizontales, muestran cómo las capas superiores se aceleran debido a los efectos difusivos. Para números de Reynolds bajos, estas capas parecen simétricas respecto a la sección central. Sin embargo, a medida que aumenta la velocidad de la tapa superior, el fluido del lado izquierdo no responde al cambio de velocidad hasta que está cerca de la pared superior, debido a las fuerzas inerciales dominantes sobre las viscosas. Del mismo modo, a mayor velocidad de la tapa, mayor es el efecto sobre las capas inferiores, que deben acelerar para mantener el caudal. Esto se refleja claramente en las figuras 3.24 - 3.29, donde la zona de velocidades negativas intensas, representada en color azul oscuro, se desplaza hacia abajo del dominio.

Al representar los vectores de velocidad sobre el campo de velocidades horizontales, es posible identificar el centro de rotación de la recirculación. Debido al aumento de las velocidades en las capas superiores y su influencia en las inferiores, este centro de rotación se desplaza hacia abajo. Este fenómeno es más evidente al observar los perfiles de velocidad horizontal en una sección específica del dominio. En la figura 3.30, se aprecia cómo el punto de cruce de la curva con el valor nulo descende con el aumento del número de Reynolds, comenzando en $Re = 100$ por $y = 7.5$ mm y llegando a descender para $Re = 2000$ hasta $y = 5.5$ mm, lo que indica un desplazamiento hacia abajo del centro de rotación. Además, en la parte superior de este mismo gráfico, se puede confirmar como, efectivamente, las capas de fluido se aceleran de forma exponencial pero solo a partir de $y = 9$ mm. Por debajo de esa altura, la velocidad aumenta, pero de forma lineal.

En conclusión, las soluciones derivadas para el fenómeno de la cavidad impulsada reflejan una coherencia y consistencia notables, alineándose de manera congruente con las expectativas iniciales. No obstante, un paso crucial en la validación y contextualización de estos hallazgos radica en su comparación con los estudios previos existentes, lo cual será abordado en la siguiente sección de discusión de resultados. Este análisis comparativo no solo enriquecerá la comprensión de los resultados obtenidos, sino que también contribuirá a situar el presente estudio dentro del panorama académico actual.

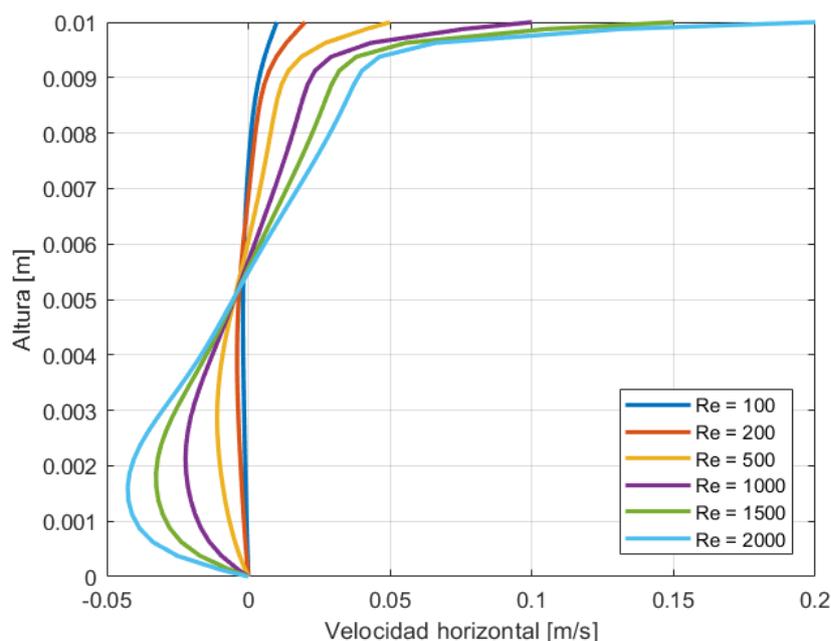


Figura 3.30 Comparación de perfiles de velocidad horizontal, $x = 0.05$ m.

3.3 Flujo en canal con cambio de sección

3.3.1 Descripción del problema

Por último, se ha propuesto una variación del problema de canal de sección constante, en el que esta vez, se propone un flujo por un canal cuya sección se incrementa en forma de escalón, siendo un caso particularmente interesante dentro del estudio de la mecánica de fluidos. En este escenario, la sección transversal del canal experimenta un cambio abrupto en su área, creando un escalón o discontinuidad en la geometría del canal. Esto genera una serie de fenómenos hidrodinámicos que son importantes de comprender y analizar.

En el caso de una sección de canal con área constante, el flujo tiende a ser más uniforme y predecible. Sin embargo, al introducir un escalón, se producen efectos como la separación del flujo, la formación de remolinos, cambios en la velocidad y la presión. Estos fenómenos pueden tener un impacto significativo en diversos aspectos, desde la eficiencia del transporte de fluidos hasta la seguridad de estructuras hidráulicas. En el diseño de perfiles aerodinámicos, como alas de aeronaves o cuerpos de cohetes, la comprensión de cómo el flujo reacciona ante cambios repentinos de geometría es esencial para optimizar la eficiencia y la estabilidad aerodinámica.

Se concluye que, se trata de un problema un poco más complejo con una geometría relativamente sencilla y cuyo estudio es de utilidad para poner a prueba de nuevo el algoritmo SIMPLE y ver cómo se comporta en escenarios no uniformes.

La geometría seleccionada consta de una entrada de altura H y una longitud de $2H$. Tras esta entrada, la sección transversal se incrementa en forma de escalón, pasando a una altura total de $2H$ y una longitud de $8H$ hasta la salida del canal, siendo la longitud $H = 0.01m$. Como se muestra en la figura 3.31, el flujo se considera que fluye desde la izquierda hasta la derecha, entrando con una velocidad uniforme u_{in} y saliendo a una presión uniforme p_{out} . Al igual que con el canal de sección constante, es de esperar que los perfiles de velocidad vayan cambiando debido a los efectos viscosos, pero en este caso, también habrá una zona de fuerte transición debido al cambio abrupto de la sección transversal. Continuando con la misma metodología, el análisis se hará en función del número de Reynolds.

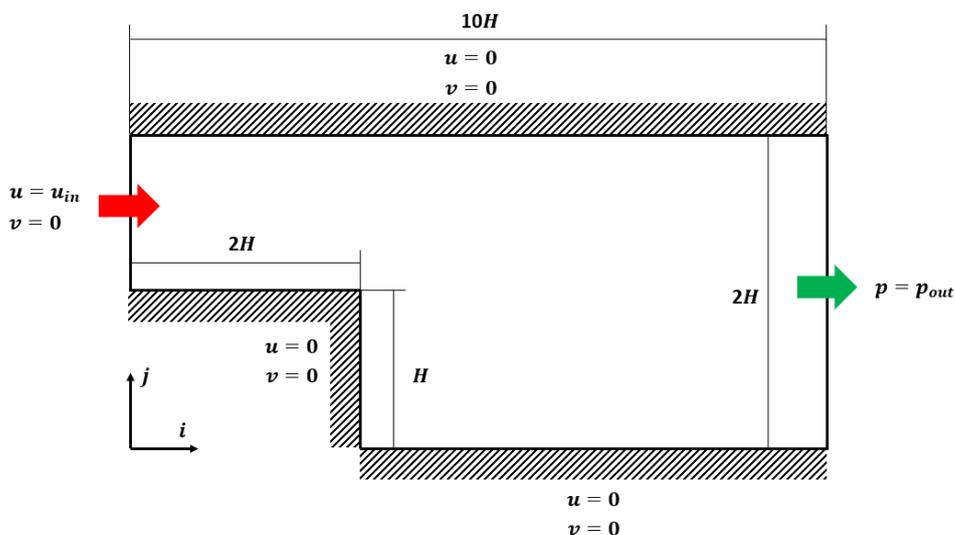


Figura 3.31 Flujo en canal con cambio de sección.

3.3.2 Condiciones de contorno

La geometría planteada, consta de cuatro paredes sin deslizamiento que delimitan el canal por el que se mueve el flujo y donde el fluido en contacto tendrá velocidad nula. Nótese que las dos paredes adicionales, son las correspondientes al cambio de la sección transversal, siendo las zonas 4 y 6 iguales en cuanto a particularización de las ecuaciones. El fluido entra por la cara izquierda del conducto de altura H con una velocidad uniforme u_{in} , mientras que sale por la derecha con una altura de $2H$ y una presión de salida uniforme p_{out} . Al igual que en el canal de sección constante, se deben aplicar las mismas adaptaciones a las celdas que se encuentran cercanas a la salida y a la pared superior.

En el dominio fluido se ha implementado una malla uniforme y co-ubicada, formada por una matriz de 40×200 celdas. Es decir, cada celda tiene una longitud de 0.5 mm. Si se revisa el código en el anexo C, es recomendable separar el dominio en dos zonas. Una primera zona cuya altura sea igual a la del escalón (for $j=1:ny_in$) y otra, desde dicha altura hasta la pared superior (for $j=ny_in+1:ny$). De esta forma, tenemos dos dominios diferenciados con dos longitudes diferentes y condiciones de contorno distintas. Lo recomendable es recorrer el dominio con bucles "For" diferenciando las zonas y sus condiciones de contorno. De forma detallada, se representa en la figura 3.32 las diferentes zonas descritas:

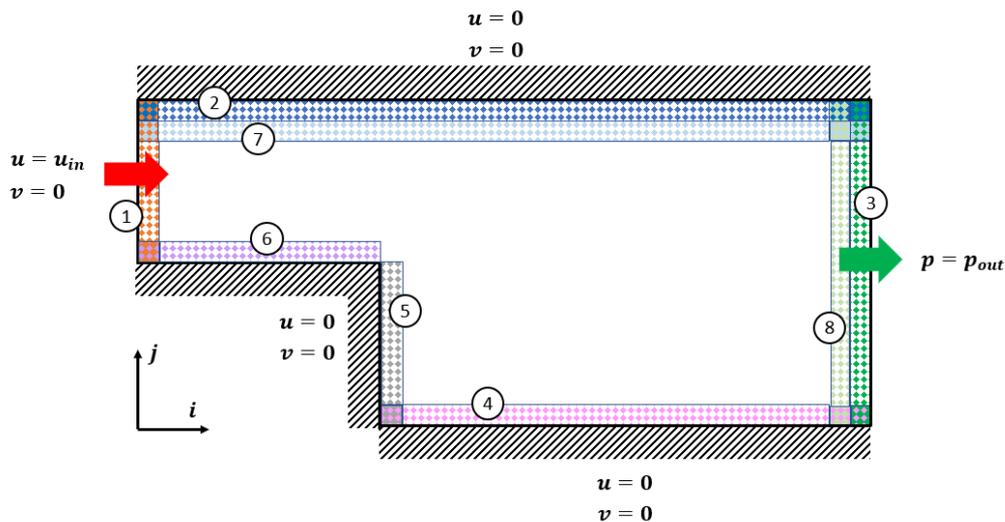


Figura 3.32 Condiciones de contorno del canal con cambio de sección.

- Zona 1: Entrada de velocidad uniforme
- Zona 2: Pared superior sin deslizamiento
- Zona 3: Salida de presión uniforme
- Zona 4: Pared inferior sin deslizamiento
- Zona 5: Pared izquierda sin deslizamiento
- Zona 6: Pared inferior en la entrada sin deslizamiento
- Zona 7: Celdas cercanas a pared superior
- Zona 8: Celdas cercanas a la salida a presión uniforme.

En referencia a las condiciones iniciales necesarias para iniciar el proceso iterativo, es crucial no comenzar desde cero. Iniciar con las variables en un estado nulo no resulta apropiado para este caso. Si se considera esta opción, durante la primera iteración, se experimentará un flujo de entrada con una velocidad u_{in} que inducirá un desequilibrio significativo en la ecuación de continuidad. Este desequilibrio implicará correcciones de presión de gran magnitud, lo cual invalida las suposiciones de pequeñas correcciones efectuadas durante el desarrollo de las ecuaciones. Además, por la experiencia adquirida durante el estudio, se ha visto que esta ecuación es fundamental para la estabilidad del proceso iterativo. Por consiguiente, existe una alta probabilidad de que si las correcciones son demasiado grandes, la solución diverja y no se logre la convergencia esperada. Así pues, para evitar este gran gradiente adverso de presión al inicio de las iteraciones, es esencial imponer u_{in} en todo el dominio. De esta manera, a medida que avancen las iteraciones, se corregirá gradualmente este efecto y se llegará a la convergencia de los resultados.

Siguiendo el método empleado hasta ahora, se usará el número de Reynolds como parámetro. En este contexto, para este caso concreto, la longitud característica del problema es $2H = 0.02m$ que es la altura total del conducto. De la misma forma, se define la densidad del fluido como $\rho = 1000 \text{ kg/m}^3$ y su viscosidad como $\mu = 2 \cdot 10^{-5} \text{ kg/(m} \cdot \text{s)}$, de tal manera que el número de Reynolds pueda ser modificado fácilmente mediante la introducción del parámetro de velocidad de entrada u_{in} . Esta velocidad será ajustada en un rango que va desde 0.1 m/s hasta 0.5 m/s, lo que resultará en un intervalo de valores de Re comprendido entre 100 y 500.

$$Re = \frac{\rho u_{in} 2H}{\mu} \quad (3.3)$$

3.3.3 Resultados y análisis

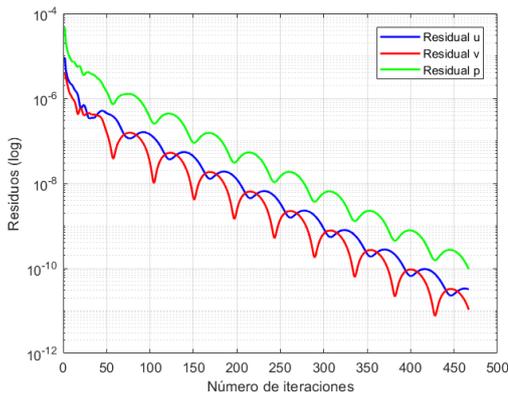
Una vez más, los resultados que se analizarán abarcan el campo de presiones, el campo de velocidades y los perfiles de velocidad. En este último caso, se está interesado en analizar el efecto que tiene el aumento brusco de la sección. Se espera, que este cambio de sección genere una zona de recirculación del fluido que será tanto mayor, cuanto más turbulento sea el flujo, es decir, cuanto mayor sea el Reynolds.

Además, debido a ese aumento en la sección, por conservación del caudal, las velocidades deberán disminuir y volver a desarrollarse hasta tener un perfil en forma de campana. Para verificar este fenómeno, se realiza un análisis detallado de los perfiles de velocidad horizontal en distintas ubicaciones a lo largo del canal, partiendo desde la zona del cambio de sección.

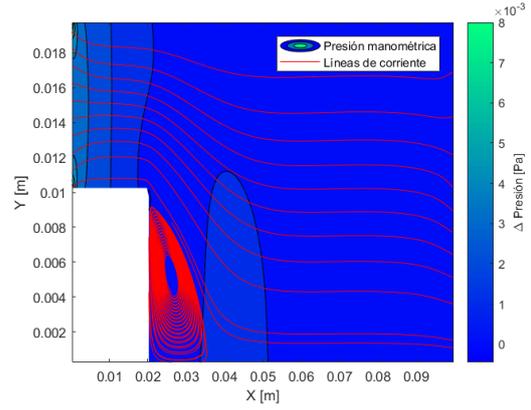
Al igual que en los casos anteriores, también se monitorearán los residuos para confirmar la convergencia de la solución y ver las limitaciones que tiene el algoritmo. Las particularidades de este problema permitirán analizar cómo se comporta el algoritmo ante geometrías no uniformes.

Tabla 3.3 Cambio de sección: iteraciones y residuos en función de Re .

Re	Iteraciones	Residuo u	Residuo v	Residuo p
50	467	3,21E-11	1,06E-11	9,50E-11
100	346	3,69E-11	6,23E-12	9,66E-11
150	304	3,21E-11	1,37E-11	9,96E-11
200	1445	8,39E-11	1,80E-11	9,84E-11
300	1273	4,55E-11	1,91E-11	9,89E-11

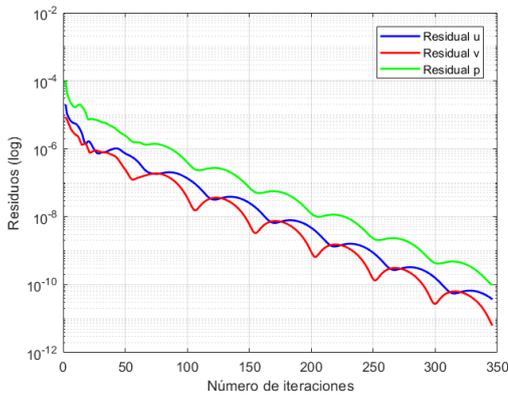


(a) Evolución de los residuos

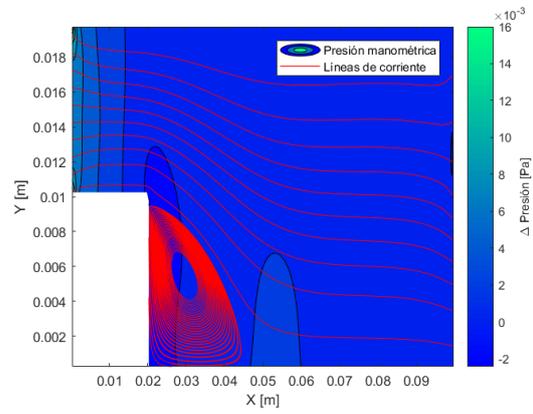


(b) Campo de presiones y líneas de corriente

Figura 3.33 Residuos y líneas de corriente $Re = 50$.

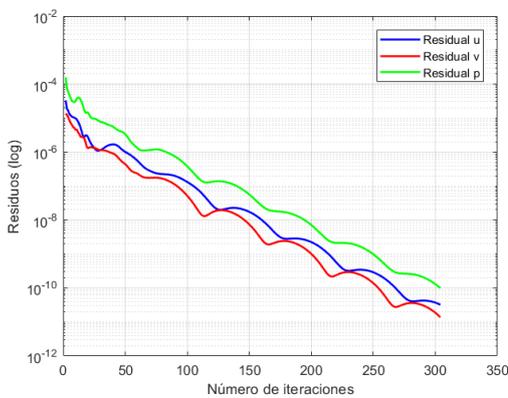


(a) Evolución de los residuos

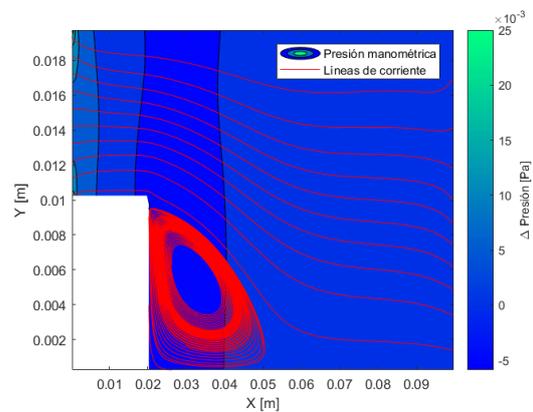


(b) Campo de presiones y líneas de corriente

Figura 3.34 Residuos y líneas de corriente $Re = 100$.

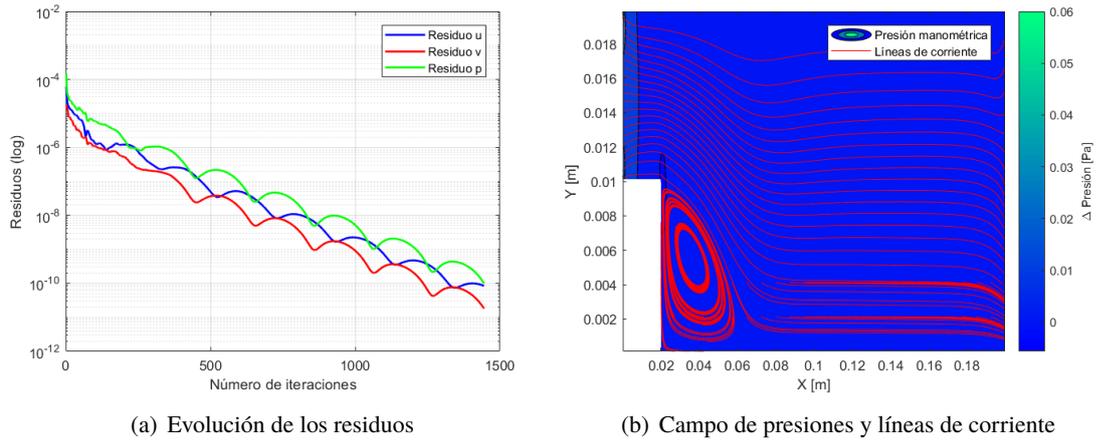


(a) Evolución de los residuos



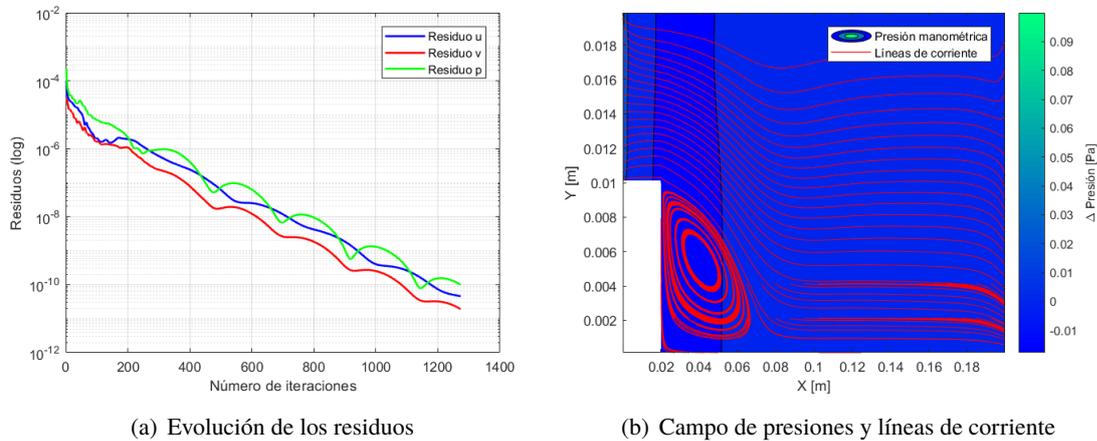
(b) Campo de presiones y líneas de corriente

Figura 3.35 Residuos y líneas de corriente $Re = 150$.



(a) Evolución de los residuos

(b) Campo de presiones y líneas de corriente

Figura 3.36 Residuos y líneas de corriente $Re = 200$.

(a) Evolución de los residuos

(b) Campo de presiones y líneas de corriente

Figura 3.37 Residuos y líneas de corriente $Re = 300$.

Este es un problema de mayor complejidad y, por ende, requiere un número considerablemente mayor de iteraciones para converger. A pesar de esto, se logra alcanzar valores dentro de la tolerancia establecida antes de llegar al límite de iteraciones globales. Es importante destacar que para los casos en los que $Re = 200$ y $Re = 300$, fue necesario realizar ajustes en el código para lograr la convergencia. Estos ajustes implicaron extender la longitud del dominio hasta $20H$ y refinar el malla, resultando en un sistema de ecuaciones más extenso, con una resolución de 80×800 nodos. Naturalmente, estos cambios incrementaron significativamente el tiempo de simulación de estos casos, pasando de unos pocos minutos a varias horas.

En las gráficas de los residuos, se distinguen dos zonas principales: durante las primeras iteraciones, se realizan ajustes iniciales en las variables hasta alcanzar la convergencia, y estas curvas presentan una forma más irregular. Luego, sigue una fase donde los residuos comienzan a disminuir, pero muestran un patrón sinusoidal. Durante estas fluctuaciones, las correcciones de velocidad vertical y de presión parecen estar sincronizadas, alcanzando sus valores extremos en las mismas iteraciones, mientras que la corrección de velocidad horizontal muestra un comportamiento desfasado, con sus extremos ocurriendo en iteraciones opuestas. Estas fluctuaciones, junto con la tardanza de la

ecuación de continuidad en converger, resultan en un número dispar de iteraciones entre los distintos casos, ya que dependen de la evolución de estos ciclos sinusoidales hacia la tolerancia establecida.

Como Mazumder [13] señala, estas fluctuaciones son inherentes al proceso iterativo, ya que cada vez que se resuelve la cantidad de momento, se invalidan los resultados de la ecuación de continuidad y viceversa, cuando se corrige la presión, los resultados de la velocidad ya no son válidos y hay que resolverlos de nuevo. Al final, en última instancia, estos residuos serán tan pequeños, que apenas influirán en la solución final.

Al examinar los campos de presión de los diferentes casos, se observa que en la región de entrada, antes del escalón, el comportamiento de la presión es similar al caso del canal de sección constante, donde la presión disminuye gradualmente hasta el escalón. Sin embargo, al llegar a la zona de cambio de sección, la presión vuelve a disminuir, aunque de manera no uniforme. Posteriormente, hay una zona de transición donde la presión se vuelve más uniforme, con un gradiente de presión bajo, hasta la salida del conducto. En el caso más laminar con $Re = 50$, la diferencia de presión entre la entrada y la salida es de $\Delta p = 0.0092$ Pa, mientras que para el flujo más turbulento con $Re = 300$, la diferencia asciende a $\Delta p = 0.1208$ Pa.

Revisando las líneas de corriente del dominio fluido, se aprecia que son predominantemente paralelas entre sí, indicando poca mezcla entre el fluido. Sin embargo, después del escalón, debido a las fuerzas inerciales, se forma una zona de recirculación de fluido que no se mezcla con el resto del flujo. Esta zona de recirculación es más extensa a medida que aumenta el número de Reynolds. Comparando las figuras 3.33 y 3.37 con $Re = 50$ y $Re = 300$, respectivamente, se observa que en la primera, la zona llega hasta una longitud de $x = 0.035$ m, mientras que para $Re = 300$, la zona se extiende hasta $x = 0.07$ m. Dado que esta zona era considerablemente extensa en los últimos casos en comparación con la longitud original del dominio, fue necesario extender el canal para permitir el desarrollo del flujo y lograr la convergencia para $Re = 200$ y $Re = 300$. Como el flujo se mueve de izquierda a derecha, la recirculación se produce en sentido de las agujas del reloj y aumenta en velocidad con el aumento de Re . En la zona intermedia, las líneas tienden a ser horizontales durante el desarrollo del flujo, y en la salida, parecen abrirse hacia las zonas de baja presión en las esquinas.

Una vez más, al analizar los campos de velocidad obtenidos, se observa claramente cómo el flujo se desarrolla rápidamente en la entrada antes del escalón y cómo, después del escalón, se desvía hacia el nuevo centro de sección para luego comenzar a desarrollarse nuevamente. Naturalmente, debido a la conservación del caudal, la velocidad se reduce sustancialmente al aumentar la sección. Además, se aprecia claramente la zona de recirculación tras el escalón, donde la velocidad horizontal se vuelve negativa, y como su extensión es mayor a medida que aumenta Re .

Para obtener una visión más detallada de la evolución de la velocidad en el canal, se han generado perfiles de velocidad en distintas secciones clave del dominio fluido: en la entrada, antes del escalón, junto al escalón, después del escalón, en una sección intermedia del dominio y en la salida. Estos perfiles revelan patrones significativos en el comportamiento del flujo. Inicialmente, la velocidad entra en el canal de manera uniforme desde la parte superior y se acelera rápidamente, siguiendo la curva de Poiseuille. Sin embargo, para el caso con $Re = 300$, no alcanza a completar su desarrollo antes del escalón, apreciándose una parte central del perfil, aún uniforme. Posterior al cambio de sección, la velocidad máxima disminuye y se desplaza hacia abajo, aunque este efecto es menos notable en casos con Re altos debido al dominio de las fuerzas inerciales. En esta etapa, en la parte inferior de la curva de velocidad, ya aparecen velocidades negativas, indicando la presencia de recirculación en esa región.

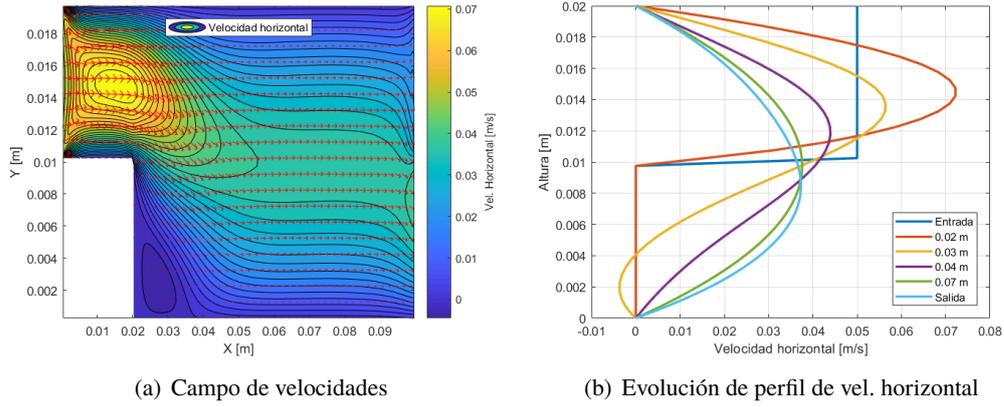


Figura 3.38 Campo de velocidades y perfiles, $Re = 50$.

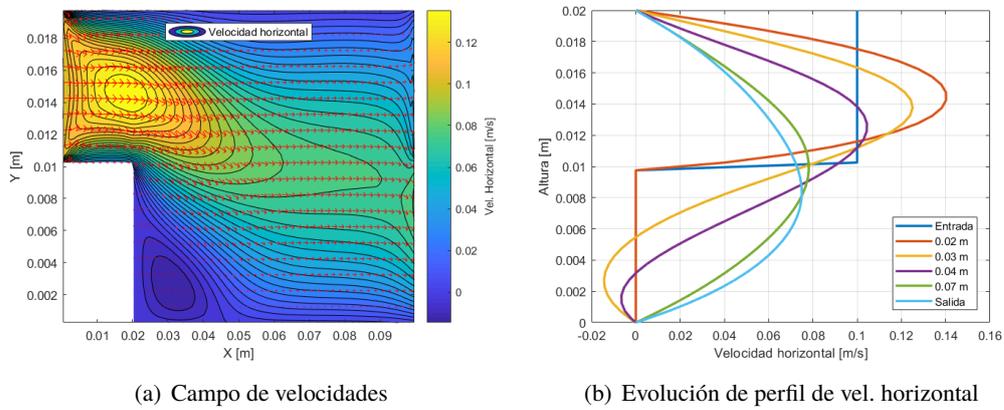


Figura 3.39 Campo de velocidades y perfiles, $Re = 100$.

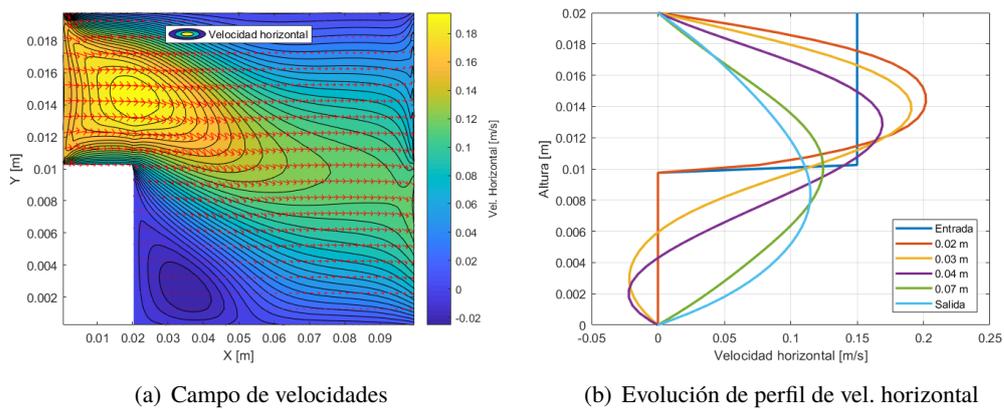


Figura 3.40 Campo de velocidades y perfiles, $Re = 150$.

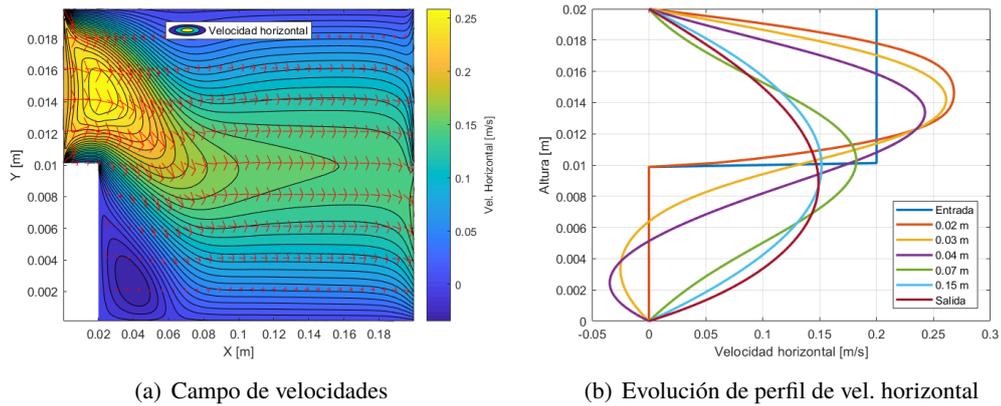


Figura 3.41 Campo de velocidades y perfiles, $Re = 200$.

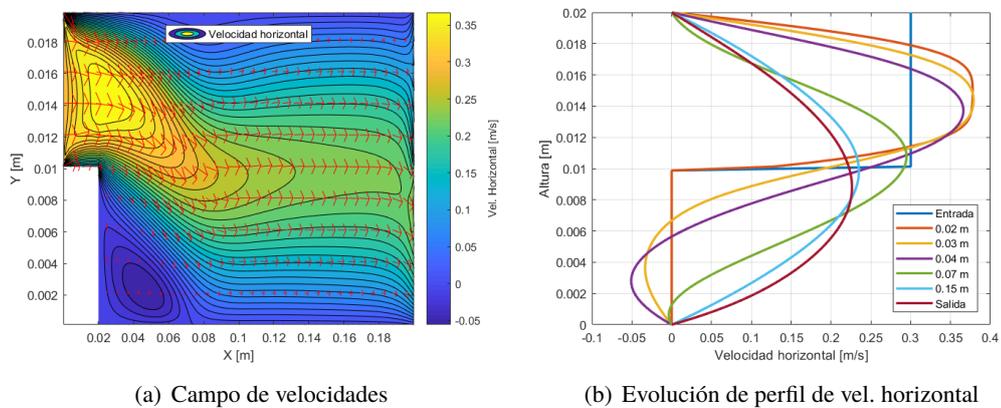


Figura 3.42 Campo de velocidades y perfiles, $Re = 300$.

Avanzando en el dominio, la superación de la zona de recirculación depende en gran medida del caso analizado. Para valores bajos de Re , esta zona está superada y las velocidades recuperan sus valores positivos, o al menos las componentes horizontales negativas disminuyen, lo que sugiere que el flujo tiende a desarrollarse de manera similar a un canal de sección constante. En contraste, para valores altos de Re , la zona de recirculación es más extensa y las velocidades se incrementan, ya que están más cerca del centro del vórtice. En la posición $x = 0.07$ m, prácticamente todas las zonas de recirculación han desaparecido para todos los casos. En los casos con Re más bajos, el flujo está prácticamente desarrollado y es muy similar al perfil de velocidad en la salida. Sin embargo, para $Re = 200$ y $Re = 300$, el flujo aún está en proceso de transición y se requiere avanzar hasta $x = 0.15$ m para que se complete el desarrollo del flujo, hecho que confirma la idea de la necesidad que hubo de extender el dominio para la convergencia del problema.

En conclusión, la resolución de este problema más complejo demuestra la versatilidad del método SIMPLE. Los resultados obtenidos coinciden con el análisis previo realizado para evaluarlos. Además, se destaca que a medida que el dominio se vuelve más complejo, la adaptación del código para reproducir la geometría, ecuaciones y condiciones de contorno se vuelve más laboriosa, requiriendo una mayor fragmentación del código para manejar las diversas casuísticas de manera precisa. Se evidencia también cómo un mismo código, cuando no está configurado adecuadamente, puede divergir inicialmente, pero con ajustes apropiados logra converger, lo que subraya la importancia de los pasos previos de análisis y configuración del problema.

4 Discusión de los resultados

Este capítulo se enfoca en la discusión de los resultados obtenidos en este estudio a través de la aplicación del método SIMPLE en los tres casos expuestos anteriormente: flujo en un canal de sección constante, flujo en una cavidad impulsada y el flujo en un canal con cambio de sección. Como ya se comentó, estos escenarios han sido cuidadosamente seleccionados debido a su relevancia en el ámbito de la ingeniería y las ciencias aplicadas, donde comprender el comportamiento de los fluidos es fundamental para el diseño y análisis de sistemas. Por otro lado, la inclusión de estos casos también ha permitido demostrar la versatilidad del método SIMPLE para adaptarlo a la resolución de diferentes geometrías del dominio fluido.

La metodología utilizada en este estudio ha facilitado la simulación numérica de estos casos, lo que ha generado datos detallados sobre variables clave como la velocidad, la presión en diversos puntos del dominio y las líneas de corriente. A priori, como se expuso en el capítulo anterior, los datos obtenidos en los distintos casos parecen coincidir con las expectativas, al menos cualitativamente. Sin embargo, para revisar la validez y relevancia de estos resultados de manera cuantitativa, es necesario contrastarlos y evaluarlos en relación con investigaciones anteriores y datos disponibles en la literatura científica.

En este sentido, la discusión se estructurará en torno a cada caso individualmente, contrastando los resultados obtenidos en este estudio con hallazgos reportados en investigaciones previas o estudios analíticos. A través de esta comparación y análisis crítico, se espera obtener una visión más completa y contextualizada de los resultados obtenidos.

Por otro lado, se busca no solo evaluar la efectividad del método SIMPLE en situaciones específicas, sino también comprender mejor sus capacidades y limitaciones en distintos contextos. Este análisis crítico es esencial para proporcionar información valiosa sobre la utilidad práctica del método y para identificar posibles áreas de mejora en futuras investigaciones en el ámbito de la dinámica de fluidos computacional.

Con el fin de contrastar y evaluar los resultados obtenidos durante este estudio se emplearán referencias específicas para cada uno de los tres problemas analizados. Para el problema de canal de sección constante, se utilizará el trabajo de Germán Ferreira et al. [7]. En el caso del problema de cavidad impulsada, se usarán los datos proporcionados por Pijush K. Kundu [10], así como al Trabajo Fin de Grado de Manuel Carreño Ruiz [18]. Por último, para el flujo en canal con cambio de sección, se considerarán los estudios de Mohammad A. Hossain et al. [15]. Como se verá a continuación, estas referencias proporcionan una base comparativa sólida para validar y discutir los resultados obtenidos en el presente estudio.

4.1 Flujo en canal de sección constante

En primer lugar, es fundamental realizar una cuidadosa selección de los datos que se utilizarán para contrastar y validar los resultados obtenidos en este estudio. Dado que el fenómeno del flujo en un canal constante es ampliamente reconocido en el campo de la mecánica de fluidos, resulta pertinente compararlo con el perfil de velocidades teórico propuesto por Poiseuille [3]. Poiseuille desarrolló una expresión que describe el perfil de velocidades para cualquier fluido una vez que ha superado su región de entrada, como se muestra en la ecuación 4.1. Este perfil adopta una forma parabólica, con su valor máximo en $y = H/2$, donde H representa la altura del canal, y alcanza un valor máximo de $\frac{u_{teo}}{U_{in}} = 1.5$. Es importante destacar que esta sección central del perfil exhibe una velocidad relativa del fluido respecto a la velocidad de entrada.

$$\frac{u}{U_{in}} = \frac{6y}{H} \left(1 - \frac{y}{H}\right) \quad (4.1)$$

El análisis comparativo de los perfiles de velocidad con el perfil teórico resulta esencial para evaluar la precisión y determinar si el flujo estudiado ha alcanzado su completo desarrollo dentro del dominio definido en el canal de sección constante. En la figura 4.1, se presenta una comparación entre los perfiles obtenidos en la salida del dominio para distintos valores de Reynolds en relación con la curva teórica de Poiseuille. Por razones de simetría, se ha representado únicamente la mitad superior de los perfiles y se han adimensionalizado con respecto a la velocidad de entrada.

Los resultados sugieren que existe una tendencia general a seguir el perfil teórico de Poiseuille; sin embargo, a medida que el número de Reynolds aumenta, se observa una desviación progresiva de esta curva teórica. Para cuantificar esta discrepancia, se ha calculado el error relativo de la velocidad máxima, dado que constituye la mayor diferencia entre los valores analíticos y experimentales. Este cálculo se realiza mediante la expresión 4.2, utilizando $u_{max,teo} = 1.5$. Los resultados de este análisis se detallan en la Tabla 4.1.

$$e_{u_{max}} = \frac{|u_{max} - u_{max,teo}|}{u_{max,teo}} \quad (4.2)$$

Las desviaciones señaladas en la tabla exhiben discrepancias considerables en relación con los valores teóricos esperados. No obstante, es importante destacar que estas discrepancias no implican que los valores obtenidos sean necesariamente deficientes o imprecisos. Más bien, los datos presentados en la tabla simplemente refuerzan la tendencia observada en la gráfica: a medida que el número de Reynolds aumenta en el dominio, el flujo no logra desarrollarse completamente antes de alcanzar la región de salida, lo que resulta en perfiles que se distancian de los valores analíticos.

Este fenómeno se hace evidente al observar los valores de Reynolds superiores a 300, donde las velocidades aún muestran uniformidad en la zona central. Por otro lado, para valores como $Re = 100$, el flujo parece estar casi completamente desarrollado, lo que se refleja en un error de velocidad máxima del 1.31%. Teniendo en cuenta el enfoque metodológico adoptado, que implica simplificaciones y aproximaciones, este nivel de precisión se considera satisfactorio.

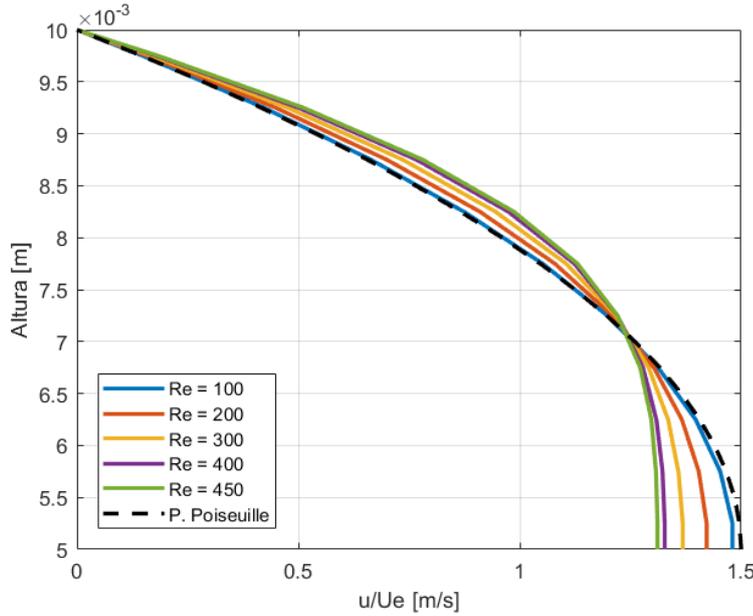


Figura 4.1 Comparación de perfiles de velocidad a la salida con el perfil teórico de Poiseuille.

Tabla 4.1 Error relativo de la velocidad máxima.

Re	Error relativo u_{max}
100	1,31 %
200	5,19 %
300	8,79 %
400	11,51 %
450	12,60 %

Otra característica distintiva de los canales que también es objeto de estudio en mecánica de fluidos es lo que se conoce como la longitud de entrada. Esta región se refiere a la longitud necesaria para que el flujo se desarrolle completamente hasta alcanzar el perfil de Poiseuille, donde la velocidad máxima se estabiliza debido al equilibrio entre los efectos viscosos e inerciales. En este contexto, los resultados obtenidos en las simulaciones de este estudio pueden ser analizados y contrastados con los datos proporcionados por la investigación de Ferreira et al.[7].

El estudio de Ferreira et al. [7] se centra en mejorar la predicción de las longitudes de entrada de fluidos en canales de sección constante en función del número de Reynolds y su relación de aspecto, especialmente en entornos de microfluidos para el rango de $0 < Re \leq 2000$. Para ello, los autores se basan en investigaciones previas de otros expertos como Atkinson, Chen o Durst, quienes ya han abordado estos estudios en el contexto de dos placas paralelas infinitas. Cada uno de estos investigadores ha intentado mejorar las predicciones anteriores mediante la obtención de expresiones diferentes para la longitud de entrada en función de Re . Todas estas expresiones tienen en cuenta el efecto del flujo turbulento; es decir, a medida que aumenta el número de Reynolds, los efectos inerciales adquieren mayor importancia y, por lo tanto, las fuerzas viscosas tardan más en adaptar el fluido hasta que alcanza su desarrollo completo.

$$L_{Atkinson}(Re) = 0.625 + 0.044Re \quad (4.3)$$

$$L_{Chen}(Re) = \frac{0.63}{1 + 0.035Re} + 0.044Re \quad (4.4)$$

$$L_{Durst}(Re) = \left[0.631^{1.6} + (0.0442Re)^{1.6} \right] \frac{1}{1.6} \quad (4.5)$$

Llegados a este punto, se pretende utilizar las expresiones previas para evaluar el grado de desarrollo del flujo en las simulaciones del estudio y determinar la precisión de los datos obtenidos. Siguiendo la metodología propuesta por Ferreira et al. [7], se establece que el flujo se considera completamente desarrollado cuando la velocidad en el centro alcanza el 99% de la velocidad máxima. En el caso del presente estudio, se ha empleado el perfil de velocidad en la salida para realizar esta evaluación y determinar así la longitud de entrada correspondiente. Los resultados de este análisis se presentan en la figura 4.2.

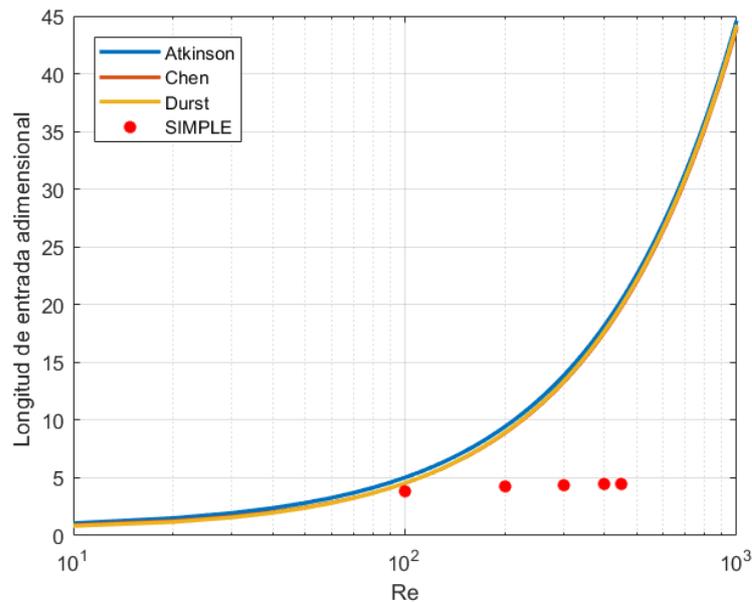


Figura 4.2 Comparación de longitudes de entrada.

Tabla 4.2 Error relativo de la longitud de entrada.

<i>Re</i>	Atkinson	Chen	Durst
100	23,88 %	15,75 %	15,78 %
200	55,17 %	52,41 %	52,64 %
300	68,72 %	67,37 %	67,54 %
400	75,72 %	74,92 %	75,05 %
450	78,09 %	77,44 %	77,56 %

La figura 4.2 reafirma la tendencia observada en los perfiles de velocidad con respecto a Poiseuille:

en las simulaciones con $Re > 100$, el flujo en la región definida no ha alcanzado su completo desarrollo. Únicamente en el caso de $Re = 100$ se puede considerar que está casi desarrollado. Para validar esta afirmación y evaluar la precisión de los datos, se calcula nuevamente el error relativo, pero esta vez entre las curvas de Atkinson, Chen o Durst y las longitudes obtenidas en las simulaciones. Estos errores relativos se detallan en la Tabla 4.2. Es natural que los errores relativos aumenten con el incremento de Re , lo cual indica que estos flujos aún carecen de la longitud necesaria para alcanzar la longitud de entrada completa. Aunque el caso $Re = 100$ parece estar prácticamente desarrollado, la posibilidad de mejorar los resultados mediante la extensión del canal es una consideración a tener en cuenta.

Los resultados obtenidos siguen las tendencias generales esperadas según las simulaciones realizadas: los perfiles de velocidad se modifican a medida que avanzan a lo largo del canal, pero sin alcanzar su desarrollo completo. Surge la pregunta sobre si, el método SIMPLE será capaz de reproducir las longitudes de entrada observadas en las curvas teóricas si se configura la geometría del canal para ello.

Para abordar esta pregunta, se decidió simular nuevamente el caso de $Re = 300$, pero extendiendo el canal hasta una longitud total de $x = 20H$, lo que debería ser suficiente para desarrollar completamente el flujo. Además, se utilizaron celdas más pequeñas para una mejor discretización del dominio, aumentando a 40 celdas por longitud H . Este ajuste conllevó un considerable aumento en el tiempo de simulación. La elección de simular el caso $Re = 300$ se debe a su valor intermedio entre los casos ya simulados anteriormente. Los resultados de estas nuevas simulaciones se presentan en la figura 4.3.

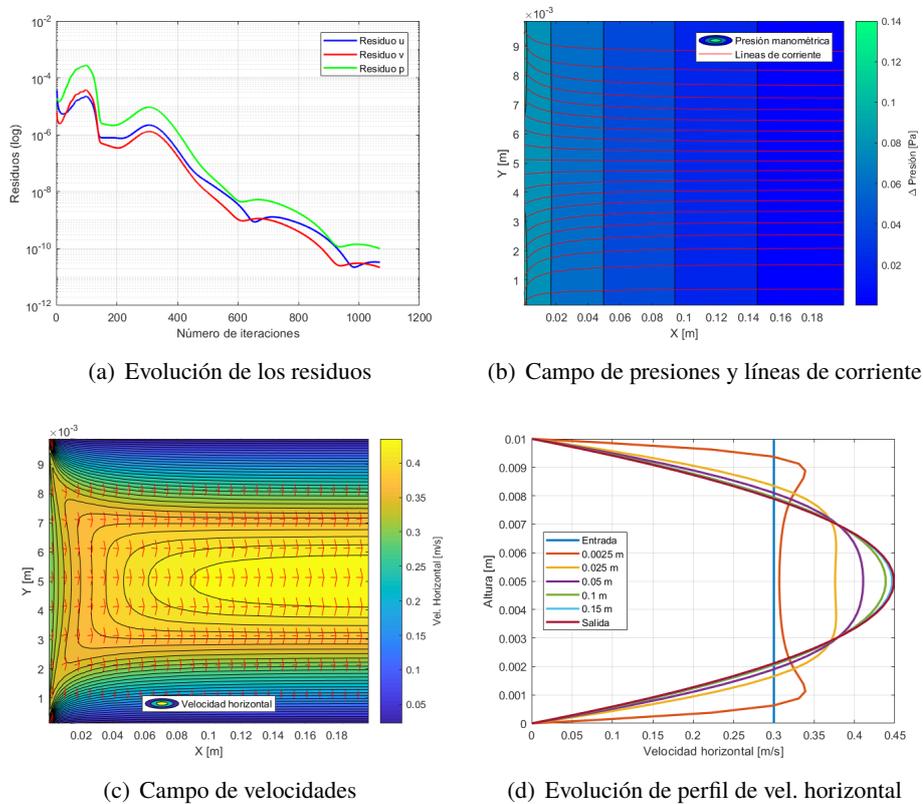


Figura 4.3 Nuevos resultados para $Re = 300$.

En el análisis de estos resultados, se observan las mismas tendencias que se identificaron en el capítulo anterior. Sin embargo, hay una diferencia notable: ahora, con la disponibilidad de un canal de mayor longitud, se evidencia cómo la velocidad alcanza un máximo $u_{max,300} = 0.4486$ m/s y se estabiliza en este valor. Por lo tanto, parece que finalmente se ha logrado desarrollar completamente el flujo. Esta conclusión se ve respaldada al comparar el perfil de velocidades en la salida con el perfil de Poiseuille, donde se observa una coincidencia efectiva, con un error relativo entre ellos de apenas el 0.31 %, y porque al representar el desarrollo de la velocidad central en la figura 4.6, se ve como tiende a una asíntota horizontal que marca la velocidad máxima.

Al calcular nuevamente la longitud de entrada para $Re = 300$, como el 99 % de $u_{max,300} = 0.4486$ m/s, se obtiene un valor de $L = 0.1284$ m, que al adimensionalizarlo y compararlo con las curvas de Atkinson, Chen o Durst, muestra una convergencia mucho más cercana a dichas curvas, con errores significativamente reducidos en comparación con los obtenidos previamente.

En resumen, los resultados alcanzados en las simulaciones son consistentes con los resultados analíticos y experimentales, demostrando una precisión satisfactoria para un estudio académico con objetivos didácticos centrados en el desarrollo de códigos propios para la resolución de dominios fluidos. Resalta la efectividad del enfoque metodológico empleado y su capacidad para generar resultados confiables y coherentes.

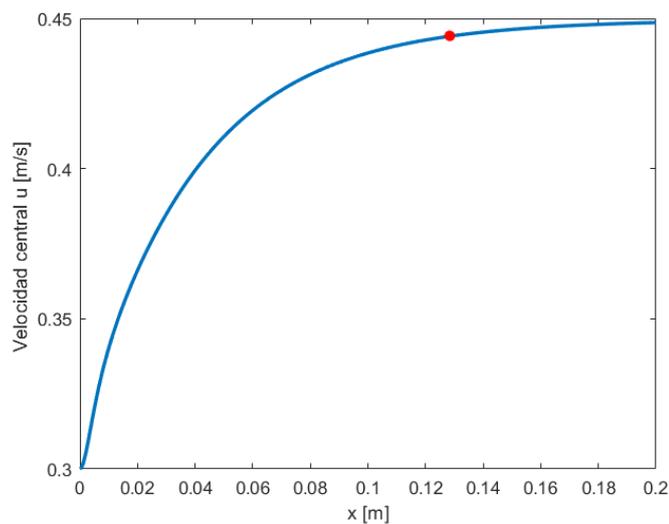


Figura 4.4 Determinación de la longitud de entrada para $Re = 300$.

Tabla 4.3 Error relativo de la velocidad máxima para $Re = 300$.

Re	Error relativo u_{max}
300*	0,31 %

Tabla 4.4 Error relativo de la longitud de entrada para $Re = 300$.

Re	Atkinson	Chen	Durst
300*	7,12 %	3,13 %	3,63 %

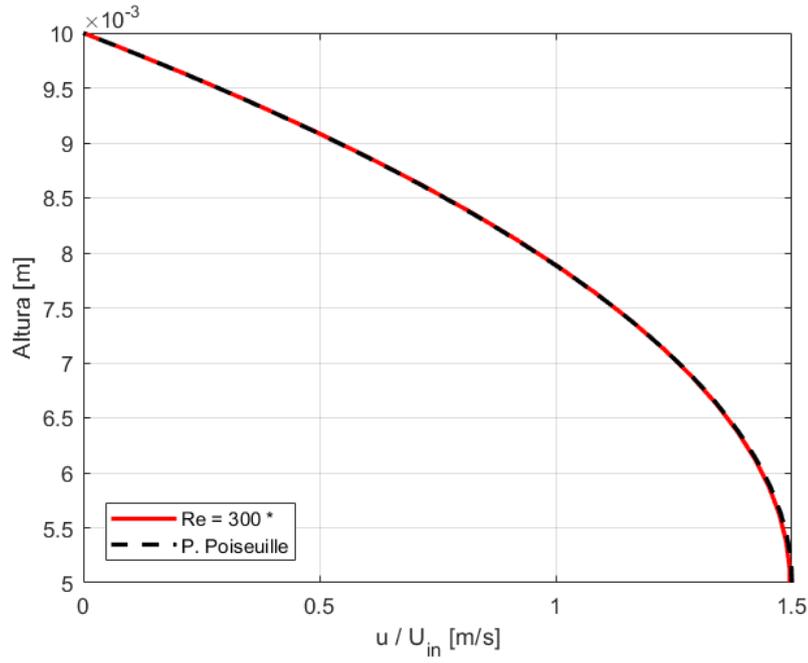


Figura 4.5 Comparación con perfil de Poiseuille para $Re = 300$.

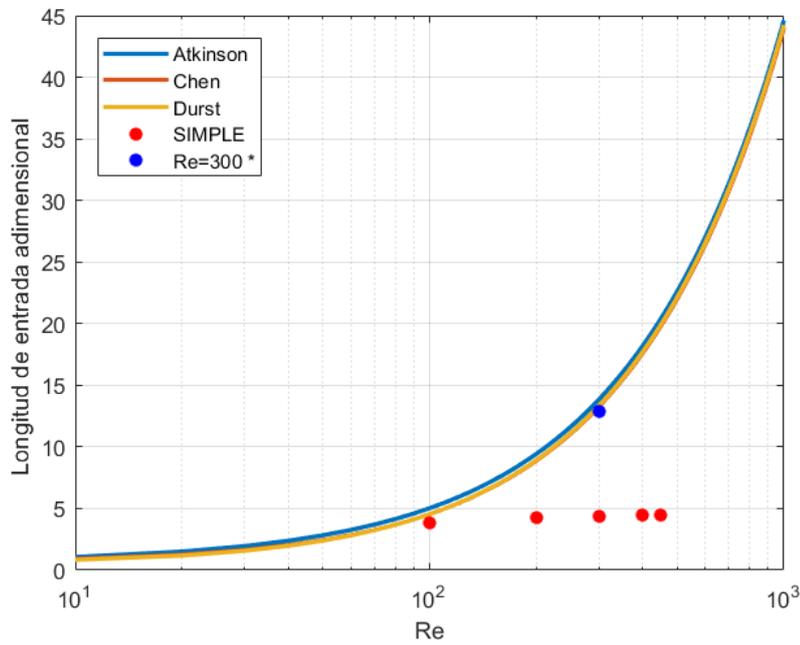


Figura 4.6 Comparación de la longitud de entrada para $Re = 300$.

4.2 Problema de cavidad impulsada

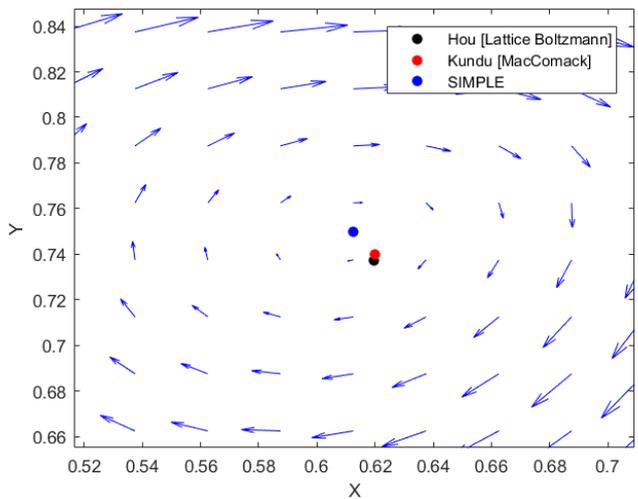
Tal como se ha mencionado en capítulos anteriores, el problema de cavidad impulsada es usado como un problema clásico para la validación de códigos CFD por su geometría y condiciones de contorno simples, y lo que es más importante, ha sido fundamento de muchos y diversos estudios por lo que hay mucha literatura disponible para poder contrastar los resultados de este estudio. Los resultados obtenidos se van a comparar con los mostrados por Kundu et. al. en su libro *Fluid Mechanics* [10], donde resuelven este mismo problema aplicando el método MacCormack y lo comparan con los resultados alcanzados por Hou et al. con el método Lattice Boltzmann.

El método MacCormack es un algoritmo numérico utilizado en la resolución de ecuaciones en derivadas parciales (EDP) especialmente en sistemas dependientes del tiempo y el espacio, lo que hace que sea un método muy empleado en la dinámica de fluidos computacional. Este método se basa en la división del dominio en una malla discreta, donde se calculan las soluciones en pasos de tiempo sucesivos. La idea clave del método MacCormack es dividir la resolución del problema en dos pasos: una predicción y una corrección. En el paso de predicción, consiste en la aproximación de las derivadas espaciales y temporales mediante la diferencia progresiva. Luego, en la etapa de corrección, se aproximan las derivadas espaciales mediante una diferencia regresiva. Combinando estas soluciones, logra obtener soluciones estables y precisas para una amplia gama de problemas. Como señala Carreño en su TFG [18], todo principiante en CFD comienza utilizando este tipo de métodos de diferencias finitas por la familiaridad con las ecuaciones diferenciales y después paulatinamente tienden a métodos basados en volúmenes finitos. Se trata de un método muy sencillo de programar para iniciarse en la mecánica de fluidos computacional.

Por otro lado, se encuentra el método Lattice Boltzmann [1] que se trata de una técnica de simulación numérica. A diferencia de los métodos tradicionales basados en la resolución directa de ecuaciones de Navier-Stokes, Lattice Boltzmann simula el comportamiento del fluido a nivel microscópico, es decir, modela la distribución y comportamiento de partículas ficticias en una malla regular. En este método, las partículas se mueven en direcciones discretas, a través de los nodos de la malla dependiendo del conjunto de velocidades discretas. Cada partícula lleva un peso asociado, que representa su distribución de probabilidad. Durante cada paso de tiempo, las partículas interactúan localmente según reglas predefinidas, que conservan la masa y el momento. El algoritmo consta de dos pasos fundamentales: uno de colisión y otro de propagación. En el paso de colisión se redistribuyen las distribuciones de probabilidad de las partículas en cada nodo en función de sus interacciones. Posteriormente, en el paso de propagación, se mueven las distribuciones de probabilidad de acuerdo con las velocidades discretas, simulando el movimiento de las partículas. Algunas de sus ventajas, son su capacidad para simular flujos complejos en geometrías irregulares, su paralelización eficiente en sistemas computacionales modernos y su flexibilidad para modelar diferentes tipos de fluidos y fenómenos físicos.

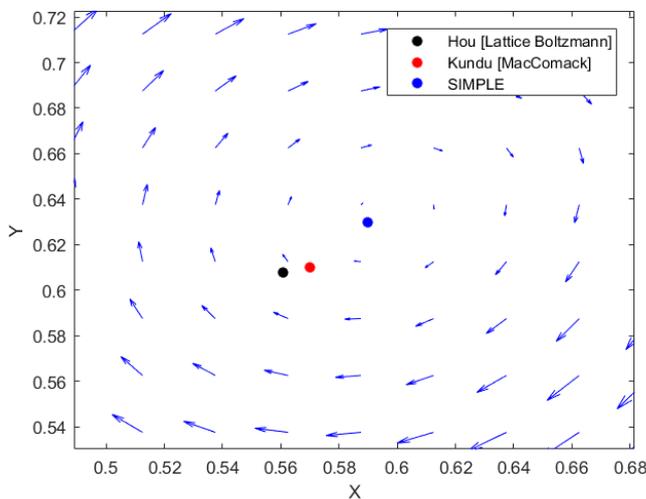
La evaluación de los resultados obtenidos en este estudio, a través de la aplicación del método SIMPLE, será llevada a cabo comparándolos con los enfoques de MacCormack y Lattice Boltzmann. Esta comparación se centrará en contrastar la ubicación del centro de rotación del fluido en la cavidad y en analizar los perfiles de velocidad en la sección intermedia para dos valores de Reynolds, $Re = 100$ y $Re = 400$.

Al analizar minuciosamente la localización del centro de rotación, en general se observa que se encuentran muy próximos unos de otros. Esta proximidad es aún mayor para $Re = 100$ en comparación con $Re = 400$, lo cual se atribuye al incremento de la turbulencia, y cómo es implementada por cada modelo. No obstante, el margen de error máximo se mantiene por debajo del 5 %, mostrando resultados positivos que se alinean con las expectativas y muestran una desviación mínima con respecto a los hallazgos reportados en la literatura. Sin embargo, es probable que estos resultados puedan mejorarse mediante el empleo de una malla más refinada, tal como se logró en el caso del flujo en canal de sección constante.



	Ubicación (X,Y)
Lattice Boltzmann	(0,6196 , 0,7373)
MacComack	(0,62 , 0,74)
SIMPLE	(0,6125 , 0,75)

Figura 4.7 Comparación de la localización del centro de rotación para $Re = 100$.



	Ubicación (X,Y)
Lattice Boltzmann	(0,5608 , 0,6078)
MacComack	(0,57 , 0,61)
SIMPLE	(0,59 , 0,63)

Figura 4.8 Comparación del campo de velocidades para $Re = 400$.

Tabla 4.5 Errores relativos de la posición del centro de rotación.

Re	e_{LB}	e_{MC}
100	1,33 %	1,05 %
400	4,436 %	4,437 %

Inspirado por la figura 4.9 presentada por Kundu et al. [10] en su libro, donde compara los perfiles de velocidad adimensionalizados en la sección intermedia de la cavidad, se ha optado por usar esta figura de referencia para contrastar los resultados alcanzados con el método SIMPLE. Se han extrapolado los resultados y dividido en las figuras para $Re = 100$ y para $Re = 400$, lo que permite evaluar con mayor detalle los valores logrados para los perfiles de velocidad.

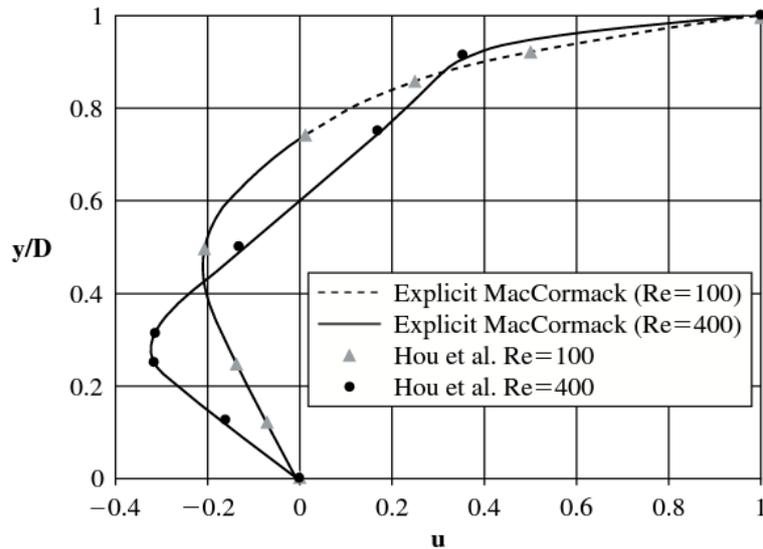


Figura 4.9 Comparación de resultados entre el método MacCormack y Hou et al [10].

En el caso de menor turbulencia, $Re = 100$, se aprecia que las curvas son bastante similares entre sí, reproduciendo casi con exactitud el perfil calculado por otros métodos. Reproduce con exactitud las condiciones de contorno impuestas tanto en la pared inferior donde la velocidad es nula, como la condición de arrastre de la tapa superior con una velocidad de U_{lid} . El perfil de velocidades transcurre casi paralelamente a la dibujada por Kundu. La mayor discrepancia parece encontrarse en el valor máximo de la velocidad negativa, aproximadamente en $y \approx 0.5$.

Como era de esperar, por resultados previos, para el caso de $Re = 400$, las diferencias entre las curvas son mucho más notables, aunque el perfil general sigue una forma similar. Se siguen cumpliendo las condiciones de contorno impuestas por la pared inferior y el movimiento de la tapa superior. Comienzan desde la tapa inferior a tomar valores negativos rápidamente, para luego ir aumentando la velocidad de forma lineal, hasta cerca de la tapa superior. Ambas curvas muestran cómo el flujo cerca de la tapa superior, alrededor de $y \approx 0.9$, cambia su comportamiento lineal para acelerarse hacia la velocidad nominal U_{lid} . El mayor error se localiza nuevamente en el valor máximo negativo de la velocidad que debería presentar el perfil. A pesar de estas diferencias, los resultados siguen siendo satisfactorios y se concluye que, a medida que aumenta la turbulencia, sería recomendable una mejor discretización del mallado para una captura más precisa.

Una vez más, los resultados obtenidos a través del método SIMPLE siguen la tendencia esperada en comparación con los datos experimentales. Esto evidencia su consistencia y confiabilidad, destacando como un enfoque de implementación sencilla para cualquier académico y que logra resultados satisfactorios.

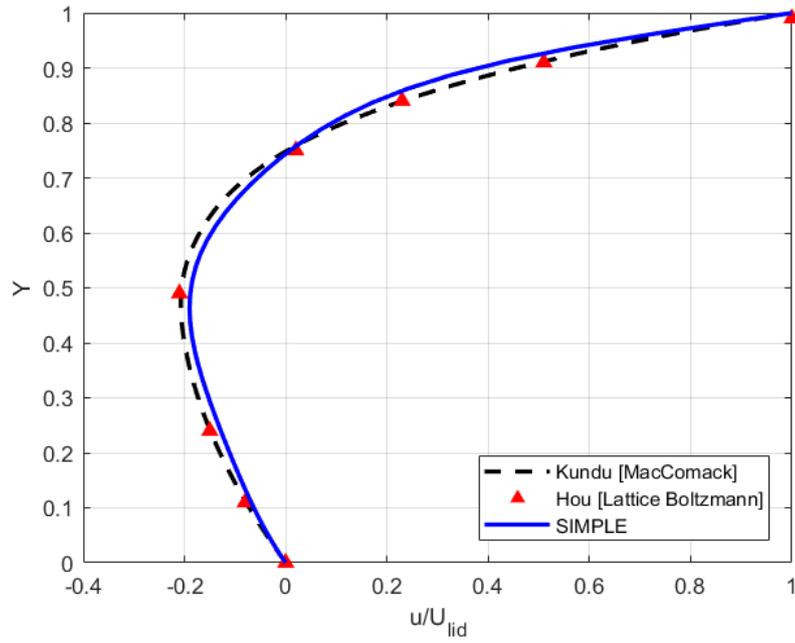


Figura 4.10 Comparación de perfiles de velocidad para $Re = 100$.

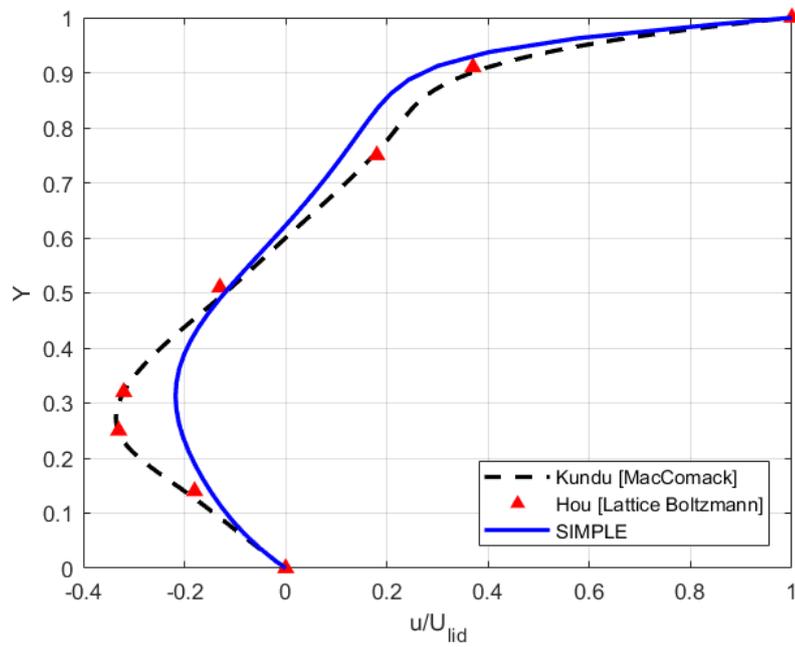


Figura 4.11 Comparación de perfiles de velocidad para $Re = 400$.

4.3 Flujo en canal con cambio de sección

Por último, pero no menos importante, se procede a contrastar los resultados alcanzados para el problema de flujo en canal con cambio de sección. Al igual que el problema de cavidad impulsada, el estudio del flujo tras un cambio de sección, es un problema clásico de la mecánica de fluidos en los que existen numerosos estudios disponibles en la literatura, lo que permite usarse también como un problema para validar códigos CFD. De todos los estudios disponibles en la literatura, se ha seleccionado la publicación *Numerical Investigation of Fluid Flow Through A 2D Backward Facing Step Channel*[15] escrito por Hossain et al, donde compara los resultados experimentales medidos por Armaly et al, con los simulados mediante ANSYS Fluent. De esta forma, se podrá comparar los resultados tanto con datos experimentales como con datos CFD de un software comercial como es ANSYS Fluent.

La geometría planteada por Hossain et al, es casi idéntica a la planteada en el problema de cambio de sección para este estudio. Se dispone de un canal con una sección de entrada de altura h , y tras recorrer una distancia $4h$, el canal se ensancha hasta una altura de $2h$ hasta su salida. Para permitir el desarrollo completo del flujo tras el escalón, la longitud del canal hasta la salida se alarga hasta $20h$. El planteamiento del problema es idéntico al de este trabajo, ir variando el valor de Reynolds y analizar el comportamiento del fluido en sus tres fases: laminar, transitorio y turbulento. El valor de Reynolds lo basa en el diámetro $H = 2h$ y como fluido de trabajo usa el agua, con lo que puede determinar el valor de la velocidad de entrada.

Para resolver los casos, configura el programa con una solución basa en presiones y dentro de las opciones, selecciona el método SIMPLE de segundo orden *upwind*. Como método de convergencia usa la técnica estándar de discretización por mínimos cuadrados y un valor de convergencia de 10^{-5} .

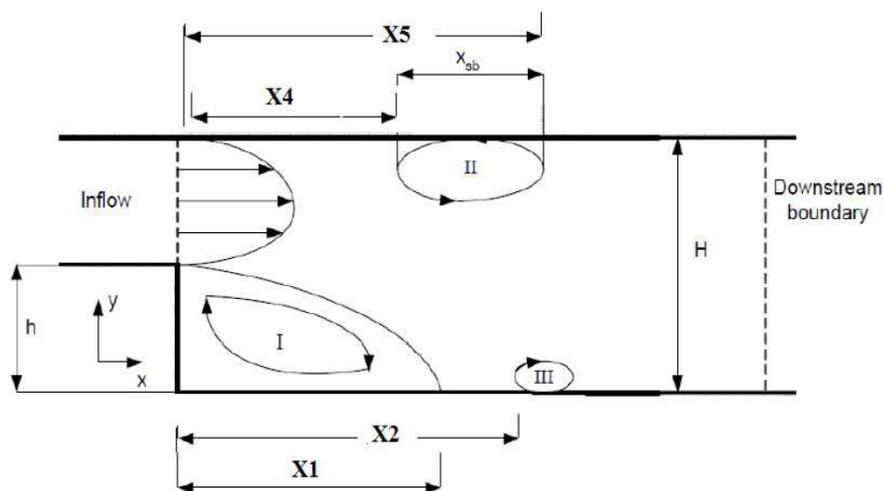


Figura 4.12 Esquema de separación del flujo [15].

Al analizar las soluciones para este caso, tanto Hossain et al. como muchos otros autores, remarcan una particularidad de este caso, y es que a medida que aumenta la turbulencia en el flujo, se observa la generación de varias zonas de recirculación, donde el flujo se desprende de la pared y pasada una cierta distancia, se vuelve a reinyectar. Hossain et al. esquematiza este fenómeno en la figura 4.12. Inicialmente, para flujos laminares entre $Re = 100$ a $Re = 300$, solo aparece una única

zona de separación del flujo (I) que se encuentra junto al escalón. A medida que aumenta Re , en torno a $Re = 350$, se aprecia una segunda zona de separación del flujo (II) pero situada en la parte superior del canal. Si el flujo se vuelve más turbulento, es posible ver una tercera zona de separación (III), fenómeno que parece volverse recurrente cuanto más turbulento es el flujo de entrada, dando lugar a nuevas zonas de separación en la parte superior del canal e inferior de manera secuencial.

A modo de validación de datos, Hossain et al. escoge comparar la longitud de reinyección de las dos primeras zonas de separación del flujo $X1$, $X4$ y $X5$. Contrasta estos datos con otros autores, entre ellos con Armaly et al. que proporciona datos medidos de forma experimental para flujos laminares y turbulentos, dando como resultado el gráfico mostrado en la figura 4.13.

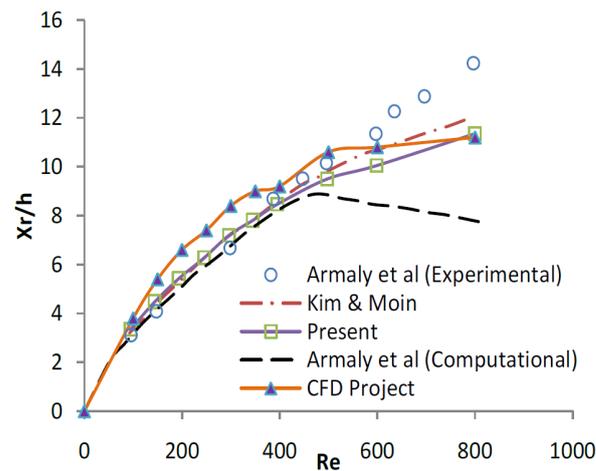


Figura 4.13 Validación de datos, Hossain et al. [15].

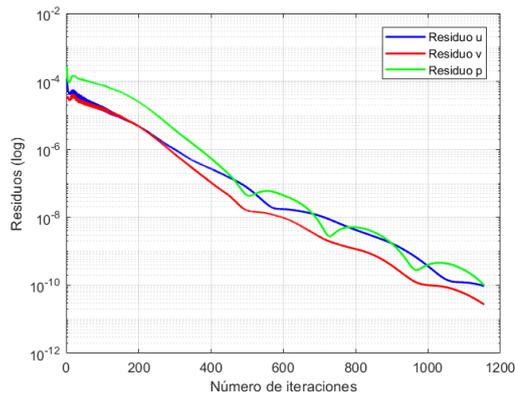
Volviendo a este estudio, es interesante usar esta misma metodología para validar los resultados obtenidos, ya que se compararán al mismo tiempo con datos experimentales y con datos numéricos calculados por el software comercial ANSYS Fluent que usa el mismo método que se ha tratado de implementar en Matlab. Sin embargo, la comparación solo se limitará al rango laminar por dos motivos: es el rango donde las hipótesis impuestas en el desarrollo siguen siendo válidas y, porque, como comenta Hossain en el propio artículo y se corrobora en el propio gráfico, los datos a partir de $Re = 600$, empiezan a mostrar una gran desviación respecto a los datos experimentales.

Llegados a este punto, para poder comparar resultados, se tuvieron que realizar más simulaciones con el código desarrollado, de esta manera se lograron obtener más casos con los que poder comparar. Se simularon tres casos más, hasta $Re = 500$ puesto que, como se ha comentado anteriormente, Hossain explica que los datos a partir de $Re = 600$, mostraban desviaciones importantes.

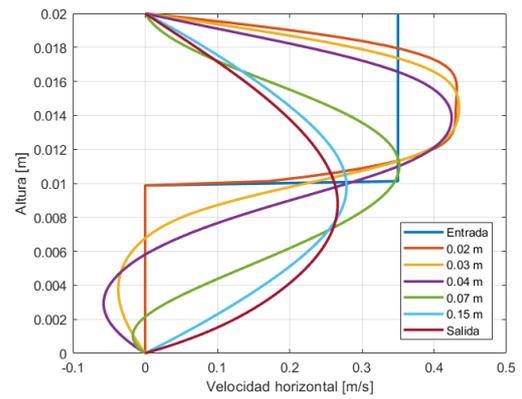
Los parámetros de simulación se adaptaron a cada caso según el Re para lograr un compromiso entre convergencia y tiempo de computación, seleccionando los siguientes parámetros:

- $Re = 300$: Tol. convergencia 10^{-11} , $n = 40$, sistema de ecuaciones de 80×800 nodos.
- $Re = 400$: Tol. convergencia 10^{-5} , $n = 80$, sistema de ecuaciones de 160×1600 nodos.
- $Re = 500$: Tol. convergencia 10^{-5} , $n = 80$, sistema de ecuaciones de 160×1600 nodos.

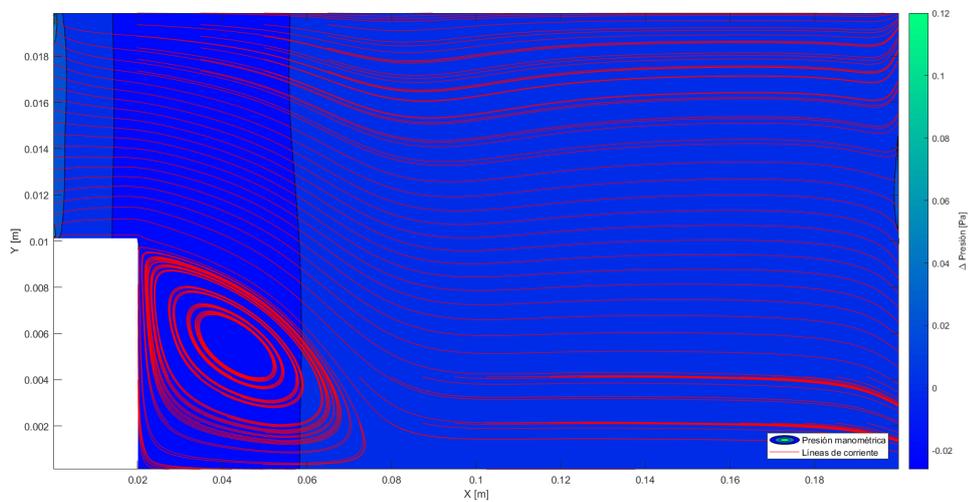
A continuación se muestran los resultados obtenidos para estas simulaciones en las figuras 4.14, 4.15 y 4.16.



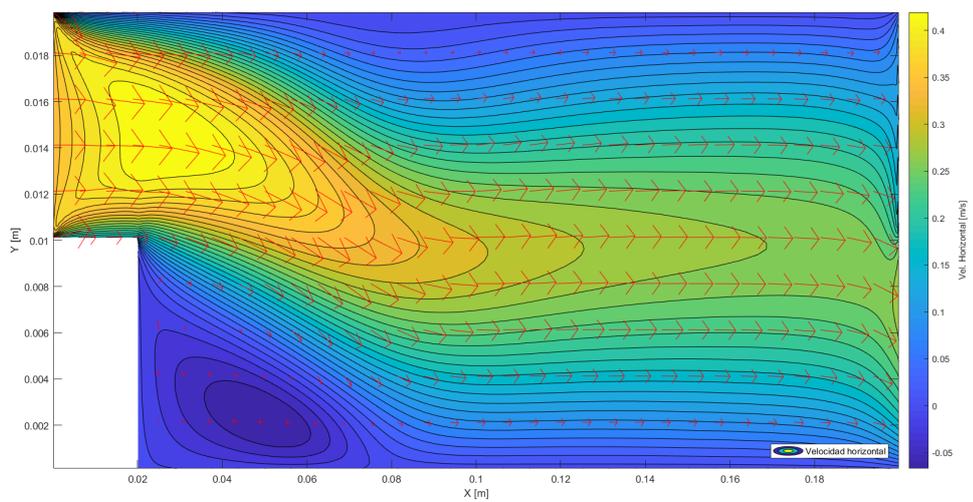
(a) Evolución de los residuos



(b) Evolución de perfil de vel. horizontal

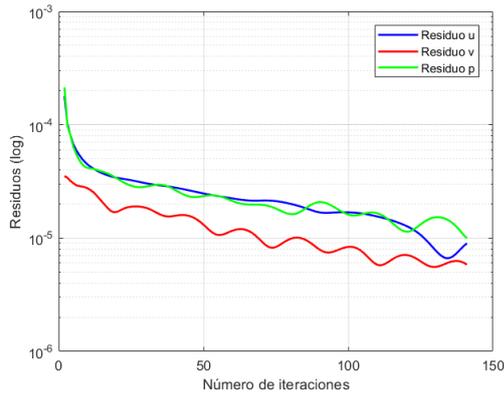


(c) Campo de presiones y líneas de corriente

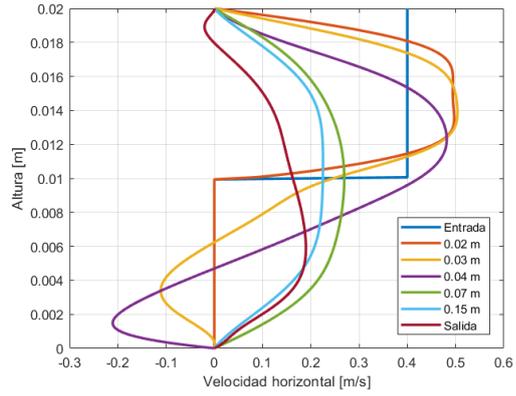


(d) Campo de velocidades

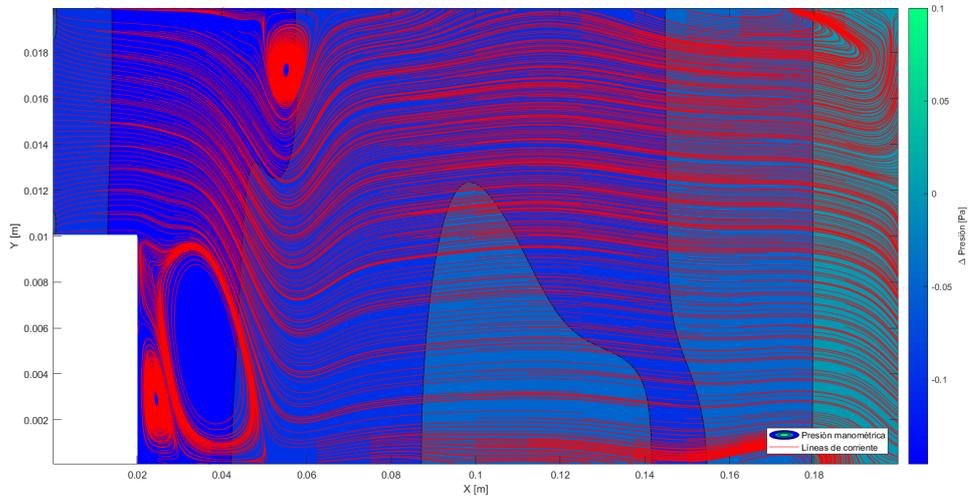
Figura 4.14 Resultados para $Re = 350$.



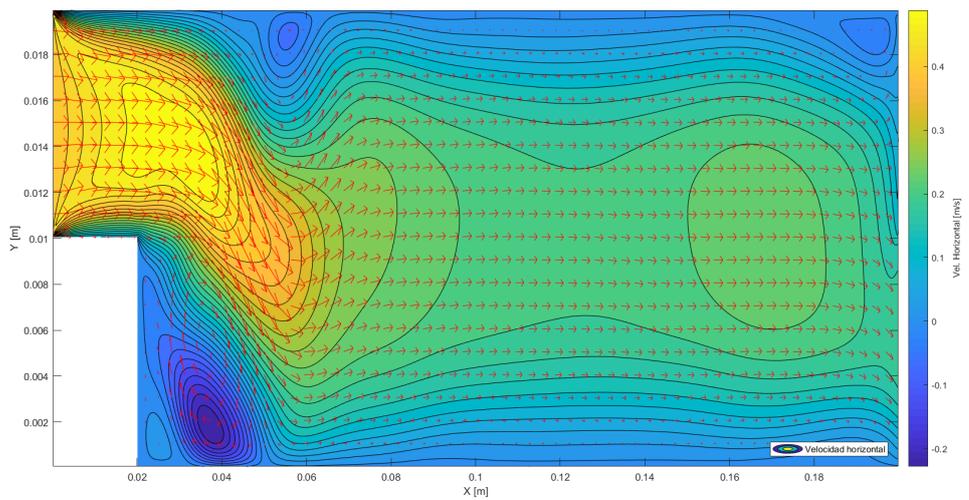
(a) Evolución de los residuos



(b) Evolución de perfil de vel. horizontal

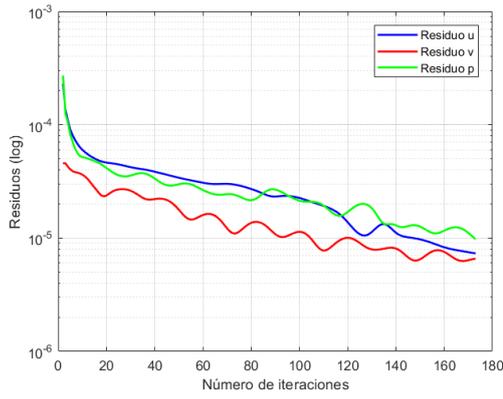


(c) Campo de presiones y líneas de corriente

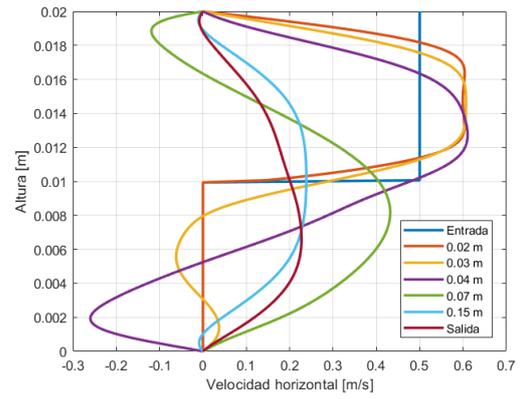


(d) Campo de velocidades

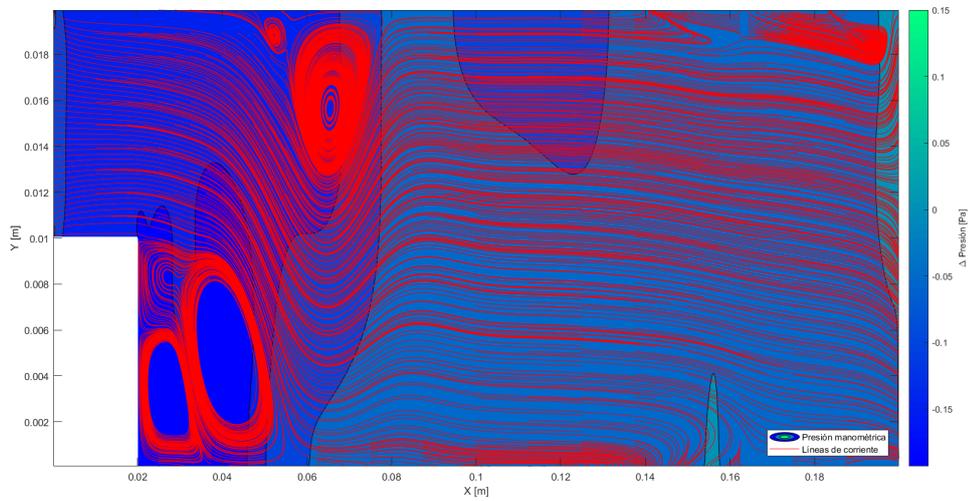
Figura 4.15 Resultados para $Re = 400$.



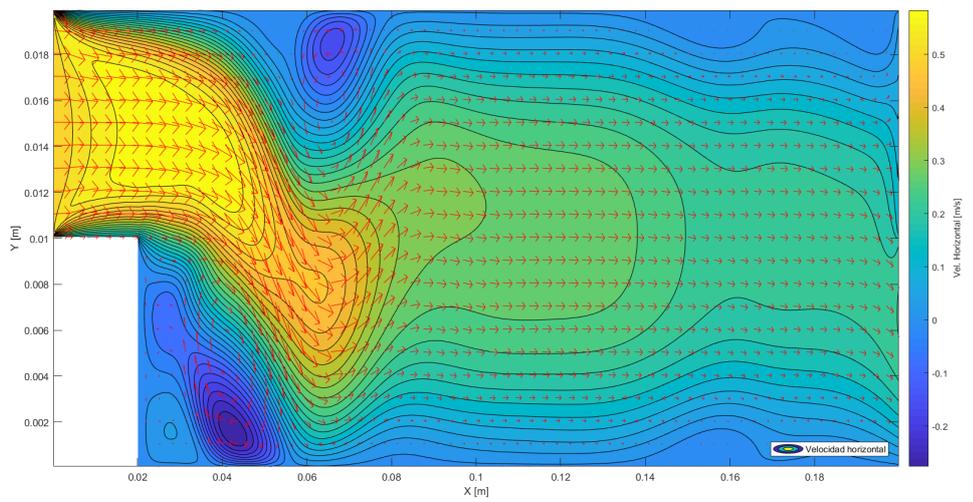
(a) Evolución de los residuos



(b) Evolución de perfil de vel. horizontal

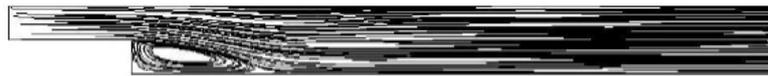


(c) Campo de presiones y líneas de corriente

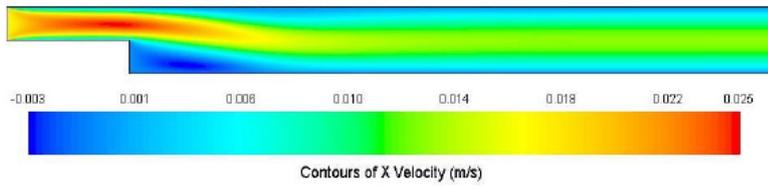


(d) Campo de velocidades

Figura 4.16 Resultados para $Re = 500$.



(a) Líneas de corriente

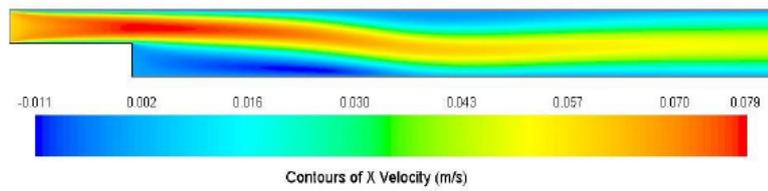


(b) Velocidad horizontal

Figura 4.17 Resultados de Hossain et al. [15] para $Re = 100$.



(a) Líneas de corriente

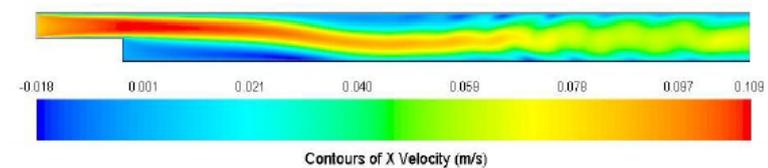


(b) Velocidad horizontal

Figura 4.18 Resultados de Hossain et al. [15] para $Re = 350$.



(a) Líneas de corriente



(b) Velocidad horizontal

Figura 4.19 Resultados de Hossain et al. [15] para $Re = 500$.

A nivel cualitativo, al comparar las líneas de corriente y las velocidades horizontales de ambos estudios, se observa que los resultados numéricos son bastante similares. Ambas investigaciones muestran tendencias similares en la aparición de zonas de recirculación a medida que aumenta la turbulencia. También se evidencia que la velocidad horizontal se frena e invierte en estas zonas de

recirculación..

Ahora bien, para evaluar el nivel de precisión de los valores a nivel cuantitativo, se tuvo que medir y adimensionalizar las longitudes de reinyección del flujo de la primera zona de desprendimiento junto al escalón, cuya notación en la mayoría de la bibliografía se denota por X_r . Los datos y su comparación se presentan en la figura 4.20.

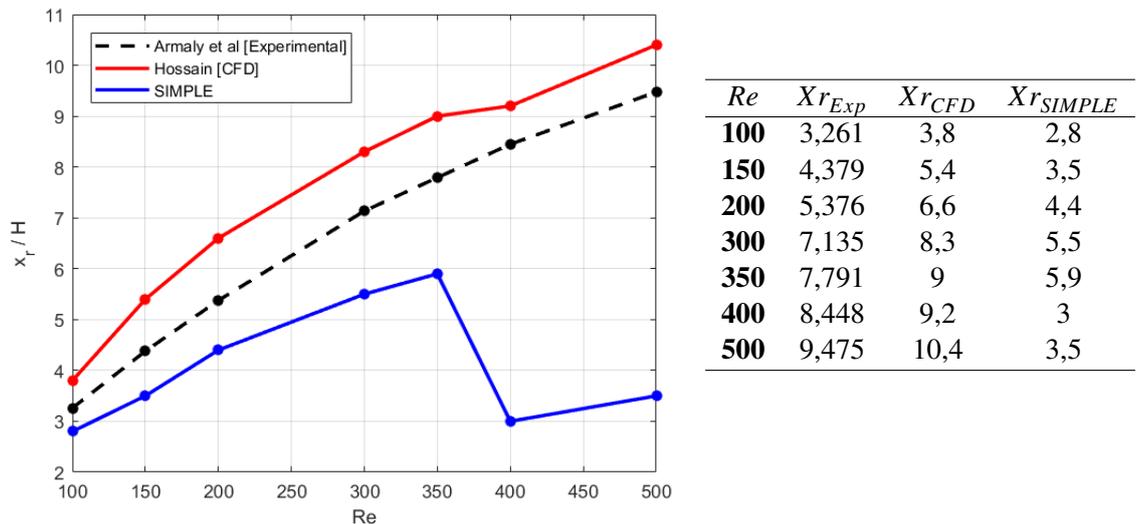
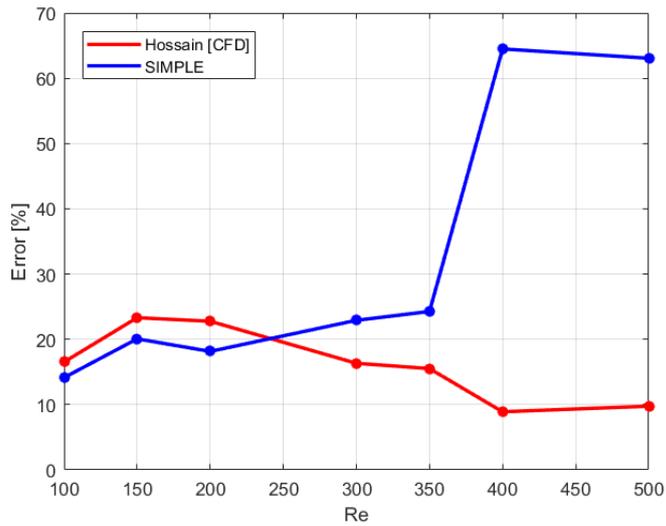


Figura 4.20 Comparación de la longitud de reinyección X_r .

Al analizar las curvas, se aprecia claramente que siguen la misma tendencia, lo que explica por qué, a nivel cualitativo, los resultados mostrados en las imágenes son tan parejos y presentan los mismos fenómenos de recirculación e inestabilidad provocados por el aumento de la turbulencia. Sin embargo, a nivel cuantitativo, las distancias x_r obtenidas se encuentran por debajo de los valores predichos numéricamente por Hossain y también por debajo de los valores medidos experimentalmente. Esta desviación es especialmente pronunciada a partir de valores de $Re > 350$, donde comienza a generarse una segunda zona de recirculación en la parte superior (II), así como una zona secundaria de recirculación dentro de la primera zona (I).

Para cuantificar estas desviaciones, se vuelve a calcular el error relativo respecto a los datos experimentales, comparando así el error que comente Hossain con los errores del código SIMPLE desarrollado en este trabajo. Estos valores quedan recogidos en la figura 4.21.

Sorprende que para valores donde el flujo es muy laminar ($100 \leq Re \leq 200$), los errores del método implementado sean incluso inferiores a los valores proporcionados por ANSYS Fluent. Sin embargo, se aprecia claramente que al aumentar el Reynolds, los errores también aumentan, divergiendo significativamente para $Re \geq 400$ en adelante. Esto probablemente se debe a que el método SIMPLE usado en ANSYS Fluent por Hossain et al. tiene incorporado algún modelo para caracterizar la turbulencia, mientras que el método implementado en Matlab no cuenta con ningún modelo de ese tipo, y por tanto, las hipótesis impuestas inicialmente dejan de ser válidas. Este es un punto de mejora evidente para futuros desarrollos: incorporar un modelo de turbulencia, como el $k - \epsilon$, para ampliar el rango de validez de este algoritmo.



Re	e_{CFD}	e_{SIMPLE}
100	16,53 %	14,14 %
150	23,32 %	20,07 %
200	22,77 %	18,15 %
300	16,33 %	22,92 %
350	15,52 %	24,27 %
400	8,90 %	64,49 %
500	9,76 %	63,06 %

Figura 4.21 Comparación del error relativo respecto a los datos experimentales.

5 Conclusiones

El desarrollo de la Dinámica de Fluidos Computacional (CFD) ha sido notable a lo largo de estos años, consolidándose como una herramienta fundamental en la ingeniería moderna. El acceso a estas herramientas de CFD ha permitido un significativo ahorro de tiempo y costes en la optimización de diseños en diversas industrias, como la automotriz, naval, energética y, especialmente, en la industria aeroespacial. Como se ha visto en este estudio, su evolución está estrechamente vinculada al aumento de la capacidad de computación de los ordenadores. Es de esperar que tenga un futuro muy prometedor y un gran salto evolutivo con la incorporación de tecnologías emergentes como la inteligencia artificial y la computación cuántica.

En este estudio se ha analizado detalladamente el método SIMPLE (*Semi-Implicit Method for Pressure Linked Equations*), por ser uno de los principales pilares sobre los que se sostienen los métodos modernos implementados en códigos comerciales como por ejemplo ANSYS Fluent. Como se ha expuesto a lo largo de este documento, el método SIMPLE presenta varias ventajas destacables frente a otros métodos para la resolución de las ecuaciones de Navier-Stokes. Es conceptualmente fácil de entender y permite el desarrollo de algoritmos prácticos para resolver problemas reales en cualquier lenguaje de programación. La metodología de discretización en volúmenes finitos representa un paso importante en la resolución de dominios fluidos, donde evaluar las propiedades de cada volumen, frente a otros métodos genéricos de resolución de ecuaciones diferenciales como MacCormack. Según lo expuesto en el capítulo 4, se consiguen resultados cuantitativos aceptables y cualitativos muy buenos, ideales para un entorno académico donde se busca comprender y visualizar los fenómenos globales del comportamiento de fluidos, permitiendo fijar mucho mejor los conocimientos adquiridos sobre mecánica de fluidos y aerodinámica. Además, aunque en este trabajo no se ha implementado, el método también permite incorporar otras variables físicas al problema, como densidad, temperatura o turbulencia.

Profundizando en las características observadas del método implementado, se llega a la conclusión de que la ecuación que representa un desafío considerable para alcanzar la convergencia es la corrección de las presiones, y esto se debe en gran medida a su estrecha relación con la ecuación de continuidad. Esta ecuación sirve como vínculo crucial entre las dos ecuaciones fundamentales de cantidad de movimiento. Al ser el componente central que conecta y equilibra las fuerzas actuantes en el flujo, cualquier discrepancia o inconsistencia en su resolución tiene un impacto significativo en la convergencia general del sistema. La complejidad inherente a esta ecuación radica en su papel de coordinación entre la distribución de presiones y la conservación de la masa dentro del flujo. La necesidad de alcanzar un equilibrio preciso entre estas variables, especialmente en condiciones de flujo complejas o turbulencia creciente, puede generar dificultades adicionales en el proceso de convergencia. Esencialmente, la corrección de las presiones actúa como un punto crítico de estabilización en el análisis del flujo, donde la resolución precisa de esta ecuación es fundamental

para obtener resultados confiables y consistentes en la simulación.

La otra clave esencial para el funcionamiento del método SIMPLE aquí implementado, es la incorporación del método PWIM (*Pressure-Weighted Interpolation Method*). Gracias a él, es posible emplear un mallado estructurado co-ubicado, ya que evita el problema de tablero de ajedrez de las oscilaciones de presión. Este problema de tablero de ajedrez es resuelto mediante PWIM porque tiene en cuenta para la interpolación de las velocidades en las caras, no solo las velocidades, si no además las presiones de los volúmenes adyacentes. Tal como se mostró en el desarrollo de las ecuaciones, el uso del mallado estructurado co-ubicado facilita enormemente el planteamiento y proyección de las ecuaciones en dichos volúmenes finitos, necesarias para la formación de los sistemas de ecuaciones que se deben resolver de forma iterativa.

Se ha constatado que, como era evidente, la complejidad de la geometría que se desea simular afecta directamente la dificultad y el tiempo necesarios para aplicar las condiciones de contorno en los sistemas de ecuaciones. En estos casos, es necesario subdividir mucho más el dominio para recoger adecuadamente las condiciones de contorno. Además, se ha corroborado la relación directa entre la discretización del dominio y los tiempos de computación. Sin un modelo adecuado para flujos turbulentos, fue necesario aumentar la discretización del dominio, buscando un equilibrio entre la tolerancia para el criterio de convergencia, la discretización y los tiempos de simulación. Para tener una idea de los tiempos, en sistemas sencillos, los tiempos de simulación fueron de unos pocos minutos, mientras que en mallados de mayor complejidad, los tiempos de cálculo superaron las cuatro horas.

Aunque se consideró la introducción de una ecuación de estado para relacionar la densidad y presión, finalmente se descartó esta opción, ya que requería incluir la ecuación de energía para calcular las temperaturas, puesto que los problemas con temperatura constante no son relevantes en aplicaciones prácticas. Por este motivo, se decidió dejarlo como una futura línea de estudio.

El código implementado en este estudio basado en el método SIMPLE fue evaluado y validado para tres geometrías diferentes: canal de sección constante, cavidad impulsada y canal con cambio de sección, logrando buenos resultados en todos los casos, y que fueron contrastados con ejemplos presentes en la literatura. En el caso del canal de sección constante se estudió la evolución de los perfiles de velocidad horizontal a lo largo del canal para determinar la longitud de entrada en función del número de Reynolds, alcanzando resultados similares a los estimados por Atkinson, Chen y Durst. Respecto a la cavidad impulsada, se compararon la posición del centro de rotación dentro de la cavidad y los perfiles de velocidad horizontal en la mitad de la sección, logrando una precisión muy buena. Por último, para el canal con cambio de sección, se compararon las imágenes de las líneas de corriente frente a las obtenidas por simulación con ANSYS Fluent por Hossain. También se comparó la longitud de reinyección del flujo tras el escalón con los datos experimentales y CFD, donde quedaron claras las limitaciones del código implementado con el aumento de la turbulencia.

Revisando el objetivo principal de este estudio, se considera que ha sido cumplido satisfactoriamente. El trabajo presentado constituye un manual académico adecuado para estudiantes o investigadores interesados en iniciarse en la dinámica de fluidos computacional (CFD). Resume los conceptos fundamentales necesarios para entender los desarrollos en este campo, ya que la literatura a menudo supone ser muy extensa y difusa. El estudio, explica detalladamente el método SIMPLE desde las ecuaciones de Navier-Stokes hasta el desarrollo del método iterativo y, aplica dicho método a ejemplos sencillos que permite familiarizarse con los sistemas de ecuaciones, condiciones de contorno y códigos implementados. Además, el Anexo B ofrece una guía práctica sobre las modificaciones necesarias en las ecuaciones según las condiciones de contorno más comunes, lo

que facilitará a los estudiantes e investigadores desarrollar sus propios códigos. En resumen, con este trabajo se ha logrado profundizar en la base de los programas CFD comerciales, para que dejen de ser empleados como una mera caja negra, y entender los procesos que siguen para resolver y simular la dinámica de los dominios fluidos.

Con las hipótesis establecidas al inicio de este estudio, el rango de aplicación se limita a geometrías sencillas 2D, malladas con elementos rectangulares, para flujos laminares a bajo número de Reynolds y flujos incompresibles. Por este motivo, las futuras líneas de investigación deben ser orientadas a superar estos límites, priorizando según la necesidad de los dominios a resolver:

- **Extensión a dominios 3D:** con la configuración actual y manteniendo un mallado ortogonal, bastaría con incorporar la tercera componente de velocidad w y la tercera ecuación de momento para un mallado hexaédrico.
- **Incorporación de un modelo de turbulencia:** como por ejemplo el modelo $k-\varepsilon$, para aumentar el rango de Reynolds y poder simular flujos más turbulentos. Sería cuestión de incorporar el modelo de viscosidad cinemática modelada por $k-\varepsilon$ y al final de cada iteración, tras la actualización de velocidades y presiones, actualizar también estos parámetros.
- **Simulación de flujos compresibles:** bastaría con incluir la ecuación de energía y estado para modelar la variabilidad de la densidad.
- **Adaptación a mallados no estructurados:** el objetivo sería poder usar mallados triangulares (2D) o tetraédricos (3D) para conseguir reproducir geometrías mucho más complejas. Esto conllevaría a volver a coger las ecuaciones generales y volver a proyectarlas según la orientación de las caras de cada triángulo, así como implementar una enumeración de los triángulos para posteriormente plantear los sistemas de ecuaciones que se deben resolver iterativamente.

Estas líneas de investigación permitirán que el código implementado en este estudio, basado en el método SIMPLE, continúe evolucionando y ampliando sus capacidades hasta equipararse con las funcionalidades más avanzadas de los softwares comerciales de simulación de dinámica de fluidos computacional.

Apéndice A

Método Line-by-line

El método line-by-line [2] es una técnica fundamental en el ámbito de los métodos computacionales, particularmente en la resolución de problemas que involucran ecuaciones diferenciales parciales (EDPs) y sistemas de ecuaciones lineales. Fue sugerido por Patankar en 1980. Este método está basado en la combinación de dos métodos: el algoritmo TDMA para problemas unidimensionales y el método iterativo de Gauss-Seidel. Se caracteriza por descomponer el dominio del problema en una serie de líneas (o segmentos), donde se aplican algoritmos y técnicas numéricas para resolver el problema en cada una de ellas de manera secuencial. Estos segmentos se barren de forma horizontal o vertical en cada iteración, determinando el valor de todos los nodos de dicha columna o fila.

Suponga que se encuentra en la iteración k , y se pretende resolver la fila j , donde las incógnitas son $\phi_{i,j}^k$ ($i = 1, 2, \dots, n$). El resto de variables en los nodos de la fila superior e inferior se consideran valores conocidos. En el caso de la fila inferior, los valores $\phi_{i,j-1}^k$ ($i = 1, 2, \dots, n$), acaban de ser calculados y actualizados en esta misma iteración k . En cuanto a los valores nodales de la fila superior $\phi_{i,j+1}^{k-1}$ ($i = 1, 2, \dots, n$), fueron calculados en el barrido anterior, en la iteración $k - 1$. La relación entre todas estas variables nodales se puede expresar de forma genérica como la ecuación discretizada A.1.

$$a_{i-1,j} \phi_{i-1,j}^k + a_{i,j} \phi_{i,j}^k + a_{i+1,j} \phi_{i+1,j}^k = d_{i,j} - a_{i,j-1} \phi_{i,j-1}^k + a_{i,j+1} \phi_{i,j+1}^{k-1} \quad (\text{A.1})$$

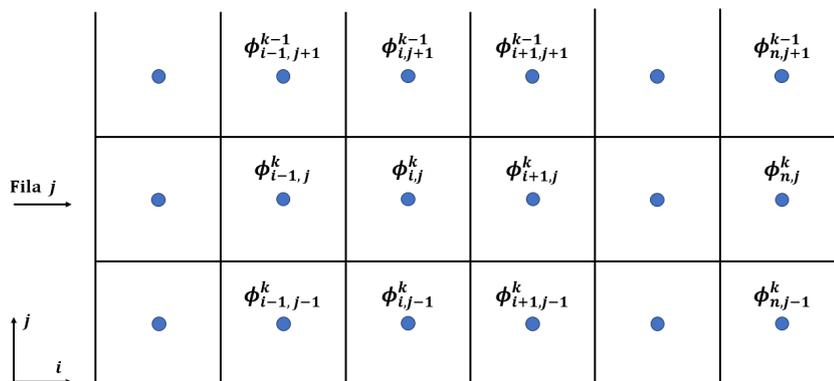


Figura A.1 Mallado para resolución mediante el método line-by-line con barrido vertical.

Ahora solo es cuestión de describir la ecuación para poder aplicar el método TDMA, es decir, se debe convertir en un sistema tridiagonal. Aplicando las relaciones mostradas a continuación, se observa como se puede construir una matriz A tridiagonal conformada por los factores que multiplican a las incógnitas, un vector d donde se encuentran las fuentes, y un vector x con las incógnitas de los valores nodales de la fila j .

$$A_{i,j} \phi_{i-1,j}^k + B_{i,j} \phi_{i,j}^k + C_{i,j} \phi_{i+1,j}^k = D_{i,j} \quad (\text{A.2})$$

$$A_{i,j} = a_{i-1,j} \quad i = 2, 3, \dots, n$$

$$B_{i,j} = a_{i,j} \quad i = 1, 2, \dots, n$$

$$C_{i,j} = a_{i+1,j} \quad i = 1, 2, \dots, n-1$$

$$D_{i,j} = d_{i,j} - a_{i,j-1} \phi_{i,j-1}^k + a_{i,j+1} \phi_{i,j+1}^{k-1} \quad i = 1, 2, \dots, n$$

$$A x = d \quad (\text{A.3})$$

$$A = \begin{bmatrix} B_{1,j} & C_{1,j} & & & 0 \\ A_{2,j} & B_{2,j} & C_{2,j} & & \\ & A_{3,j} & B_{3,j} & \ddots & \\ & & \ddots & \ddots & C_{n-1,j} \\ 0 & & & A_{n,j} & B_{n,j} \end{bmatrix}, \quad x = \begin{pmatrix} \phi_{1,j}^k \\ \phi_{2,j}^k \\ \phi_{3,j}^k \\ \vdots \\ \phi_{n,j}^k \end{pmatrix}, \quad d = \begin{pmatrix} D_{1,j} \\ D_{2,j} \\ D_{3,j} \\ \vdots \\ D_{n,j} \end{pmatrix}$$

Una vez resueltos y actualizados los valores nodales $\phi_{i,j}^k$ ($i = 1, 2, \dots, n$) de la fila j , se repite este mismo proceso para las sucesivas filas hasta terminar el barrido. Tras esto, el método recomienda que se realice otro barrido, pero esta vez por columnas, consiguiendo así una mejor convergencia.

Algunas ventajas que tiene este método son:

- **Eficiencia Computacional:** Al resolver el problema en segmentos pequeños, el método line-by-line puede ser más eficiente computacionalmente que enfoques globales que requieren la discretización de todo el dominio simultáneamente.
- **Flexibilidad:** El método line-by-line es altamente flexible y puede adaptarse a una amplia variedad de problemas, desde la simulación de fluidos hasta la propagación de ondas electromagnéticas.
- **Paralelización:** La naturaleza secuencial del método line-by-line permite su fácil paralelización, lo que lo hace adecuado para implementaciones en sistemas de cómputo distribuido o con unidades de procesamiento gráfico (GPU).

Pero también, tiene algunas limitaciones y consideraciones:

- Acoplamiento entre Segmentos: El acoplamiento entre segmentos puede introducir desafíos adicionales, especialmente en problemas no lineales o altamente no lineales, donde la convergencia puede ser más difícil de alcanzar.
- Precisión: La precisión de la solución puede variar dependiendo de la densidad de la discretización y la naturaleza del problema. Es importante realizar análisis de sensibilidad y estudios de convergencia para garantizar la validez de los resultados obtenidos.

Como se comentó en el capítulo 2, dentro de las funciones para la resolución de las ecuaciones de cantidad de movimiento y corrección de presiones, es necesario implementar la identificación de estas variables y coeficientes, de manera que se pueda formar la matriz tridiagonal A y el vector d . Una vez hecho esto, el sistema tridiagonal se resuelve con la función propia $A \setminus d$ implementada en Matlab.

Apéndice B

Condiciones de contorno

El propósito de este Anexo B es proporcionar una guía detallada de las ecuaciones principales utilizadas en el algoritmo SIMPLE. En primer lugar, se presentan las ecuaciones generales necesarias para las celdas interiores. Luego, según la condición de contorno específica que se aplique, se indican los ajustes necesarios en las ecuaciones para implementar dicha condición en el código. Estos ajustes se resaltan en rojo para una fácil identificación.

Para utilizar esta guía, primero se debe buscar la condición de contorno deseada y luego consultar las ecuaciones correspondientes que se desean resolver, ya sean:

- Ecuación de cantidad de movimiento en X
- Ecuación de cantidad de movimiento en Y
- Interpolación PWIM para velocidades en las caras
- Corrección de presiones (Ec. continuidad)
- Actualización de velocidades

Es importante tener en cuenta que, en el caso específico de las esquinas del dominio fluido, será necesario combinar adecuadamente los cambios provocados en las ecuaciones debido a la interacción de ambas condiciones de contorno.

B.1 Ecuaciones generales para celdas interiores

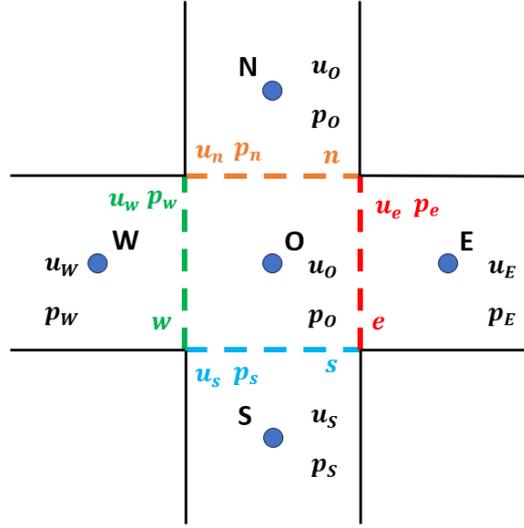


Figura B.1 Celda interior de referencia.

Ecuación de cantidad de movimiento en X

$$A_O^X \hat{u}_O + A_W^X \hat{u}_W + A_E^X \hat{u}_E + A_N^X \hat{u}_N + A_S^X \hat{u}_S = S_O^X \quad (\text{B.1})$$

$$A_O^X = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.2})$$

$$A_E^X = - \left[\frac{|(\rho u)_e| - (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} \right] \Delta y \quad (\text{B.3})$$

$$A_W^X = - \left[\frac{|(\rho u)_w| + (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y \quad (\text{B.4})$$

$$A_N^X = - \left[\frac{|(\rho v)_n| - (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} \right] \Delta x \quad (\text{B.5})$$

$$A_S^X = - \left[\frac{|(\rho v)_s| + (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.6})$$

$$S_O^X = \frac{1}{2} (p_W - p_E) \Delta y \quad (\text{B.7})$$

Ecuación de cantidad de movimiento en Y

$$A_O^Y \hat{v}_O + A_W^Y \hat{v}_W + A_E^Y \hat{v}_E + A_N^Y \hat{v}_N + A_S^Y \hat{v}_S = S_O^Y \quad (\text{B.8})$$

$$A_O^Y = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.9})$$

$$A_E^Y = - \left[\frac{|(\rho u)_e| - (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} \right] \Delta y \quad (\text{B.10})$$

$$A_W^Y = - \left[\frac{|(\rho u)_w| + (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y \quad (\text{B.11})$$

$$A_N^Y = - \left[\frac{|(\rho v)_n| - (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} \right] \Delta x \quad (\text{B.12})$$

$$A_S^Y = - \left[\frac{|(\rho v)_s| + (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.13})$$

$$S_O^Y = \frac{1}{2} (p_S - p_N) \Delta x \quad (\text{B.14})$$

Interpolación PWIM para velocidades en las caras

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{1}{2} \frac{p_{EE} - p_O}{2 A_{O|E}^X} \Delta y + \frac{1}{2} \frac{p_E - p_W}{2 A_{O|O}^X} \Delta y - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E - p_O}{2} \Delta y \quad (\text{B.15})$$

$$\hat{v}_n = \frac{\hat{v}_O + \hat{v}_N}{2} + \frac{1}{2} \frac{p_{NN} - p_O}{2 A_{O|N}^Y} \Delta x + \frac{1}{2} \frac{p_N - p_S}{2 A_{O|O}^Y} \Delta x - \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] \frac{p_N - p_O}{2} \Delta x \quad (\text{B.16})$$

$$\hat{u}_w|_O = \hat{u}_e|_W \quad (\text{B.17})$$

$$\hat{v}_s|_O = \hat{v}_n|_S \quad (\text{B.18})$$

Corrección de presiones (Ec. continuidad)

$$A_O^p p'_O + A_W^p p'_W + A_E^p p'_E + A_N^p p'_N + A_S^p p'_S = S_O^p \quad (\text{B.19})$$

$$A_O^p = \frac{\rho_e \Delta y^2}{2} \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] + \frac{\rho_w \Delta y^2}{2} \left[\frac{1}{A_{O|W}^X} + \frac{1}{A_{O|O}^X} \right] + \frac{\rho_n \Delta x^2}{2} \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] + \frac{\rho_s \Delta x^2}{2} \left[\frac{1}{A_{O|S}^Y} + \frac{1}{A_{O|O}^Y} \right] \quad (\text{B.20})$$

$$A_E^p = -\frac{\rho_e \Delta y^2}{2} \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \quad (\text{B.21})$$

$$A_W^p = -\frac{\rho_w \Delta y^2}{2} \left[\frac{1}{A_{O|W}^X} + \frac{1}{A_{O|O}^X} \right] \quad (\text{B.22})$$

$$A_N^p = -\frac{\rho_n \Delta x^2}{2} \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] \quad (\text{B.23})$$

$$A_S^p = -\frac{\rho_s \Delta x^2}{2} \left[\frac{1}{A_{O|S}^Y} + \frac{1}{A_{O|O}^Y} \right] \quad (\text{B.24})$$

$$S_O^p = -\dot{m}_{NB} = -[(\rho_e \hat{u}_e - \rho_w \hat{u}_w) \Delta y + (\rho_n \hat{v}_n - \rho_s \hat{v}_s) \Delta x] \quad (\text{B.25})$$

Actualización de presiones y velocidades

$$p_O^{k+1} = p_O^k + \omega_p p'_O \quad (\text{B.26})$$

$$u_O^{k+1} = \hat{u}_O + \omega_{uv} \frac{p'_W - p'_E}{2 A_{O|O}^X} \Delta y \quad (\text{B.27})$$

$$v_O^{k+1} = \hat{v}_O + \omega_{uv} \frac{p'_S - p'_N}{2 A_{O|O}^Y} \Delta x \quad (\text{B.28})$$

$$u_e^{k+1} = \hat{u}_e + \omega_{uv} \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p'_O - p'_E}{2} \Delta y \quad (\text{B.29})$$

$$v_n^{k+1} = \hat{v}_n + \omega_{uv} \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] \frac{p'_O - p'_N}{2} \Delta x \quad (\text{B.30})$$

$$u_w|_O = u_e|_W \quad (\text{B.31})$$

$$v_s|_O = v_n|_S \quad (\text{B.32})$$

B.2 Pared izquierda sin deslizamiento

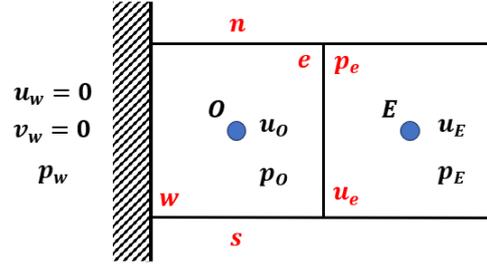


Figura B.2 Pared izquierda sin deslizamiento.

Ecuación de cantidad de movimiento en X

$$A_O^X = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{3\mu_O}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.33})$$

$$A_E^X = - \left[\frac{|(\rho u)_e| - (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{\mu_O}{3\Delta x} \right] \Delta y \quad (\text{B.34})$$

$$A_W^X = 0 \quad (\text{B.35})$$

$$\text{Aprox. 1º Orden} \quad S_O^X = \frac{p_O - p_E}{2} \Delta y \quad (\text{B.36})$$

$$\text{Aprox. 2º Orden} \quad S_O^X = (p_O - p_E) \Delta y \quad (\text{B.37})$$

Ecuación de cantidad de movimiento en Y

$$A_O^Y = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{3\mu_O}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.38})$$

$$A_E^Y = - \left[\frac{|(\rho u)_e| - (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{\mu_O}{3\Delta x} \right] \Delta y \quad (\text{B.39})$$

$$A_W^Y = 0 \quad (\text{B.40})$$

Interpolación PWIM para velocidades en las caras

Aprox. 1º Orden

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{1}{2} \frac{p_{EE} - p_O}{2 A_{O|E}^X} \Delta y + \frac{\Delta y}{2 A_{O|O}^X} \left[\frac{p_E - p_O}{2} \right] - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E - p_O}{2} \Delta y \quad (\text{B.41})$$

Aprox. 2º Orden

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{1}{2} \frac{p_{EE} - p_O}{2 A_{O|E}^X} \Delta y + \frac{\Delta y}{2 A_{O|O}^X} [p_E - p_O] - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E - p_O}{2} \Delta y \quad (\text{B.42})$$

$$\hat{u}_w|_O = 0 \quad (\text{B.43})$$

Corrección de presiones (Ec. continuidad)

$$A_O^p = \frac{\rho_e \Delta y^2}{2} \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] + \frac{\rho_n \Delta x^2}{2} \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] + \frac{\rho_s \Delta x^2}{2} \left[\frac{1}{A_{O|S}^Y} + \frac{1}{A_{O|O}^Y} \right] \quad (\text{B.44})$$

$$A_W^p = 0 \quad (\text{B.45})$$

$$S_O^p = -[(\rho_e \hat{u}_e) \Delta y + (\rho_n \hat{v}_n - \rho_s \hat{v}_s) \Delta x] \quad (\text{B.46})$$

Actualización de velocidades

$$u_w|_O = 0 \quad (\text{B.47})$$

B.3 Pared derecha sin deslizamiento

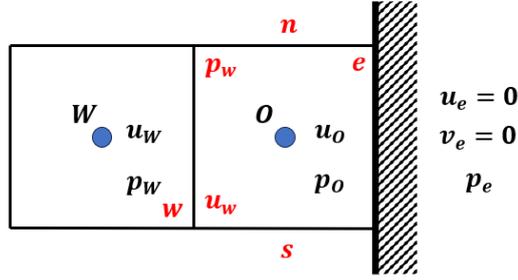


Figura B.3 Pared derecha sin deslizamiento.

Ecuación de cantidad de movimiento en X

$$A_O^X = \left[\frac{3\mu_O}{\Delta x} + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.48})$$

$$A_E^X = 0 \quad (\text{B.49})$$

$$A_W^X = - \left[\frac{|(\rho u)_w| + (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} + \frac{3\mu_O}{\Delta x} \right] \Delta y \quad (\text{B.50})$$

$$\text{Aprox. 1}^\circ \text{ Orden} \quad S_O^X = \frac{p_w - p_o}{2} \Delta y \quad (\text{B.51})$$

$$\text{Aprox. 2}^\circ \text{ Orden} \quad S_O^X = (p_w - p_o) \Delta y \quad (\text{B.52})$$

Ecuación de cantidad de movimiento en Y

$$A_O^Y = \left[\frac{3\mu_O}{\Delta x} + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.53})$$

$$A_E^Y = 0 \quad (\text{B.54})$$

$$A_W^Y = - \left[\frac{|(\rho u)_w| + (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} + \frac{\mu_O}{3\Delta x} \right] \Delta y \quad (\text{B.55})$$

Interpolación PWIM para velocidades en las caras

$$\hat{u}_e = 0 \quad (\text{B.56})$$

Para la columna $i = n - 1$:

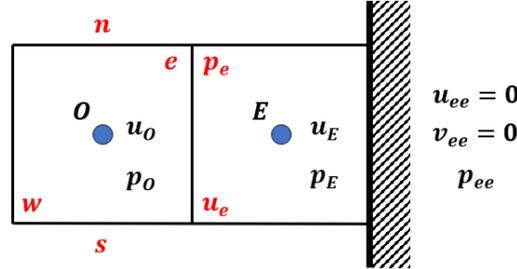


Figura B.4 Celda cerca de la pared derecha.

Aprox. 1º Orden

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{\Delta y}{2 A_{O|E}^X} \left[\frac{p_E - p_O}{2} \right] + \frac{1}{2} \frac{p_E - p_W}{2 A_{O|O}^X} \Delta y - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E - p_O}{2} \Delta y \quad (\text{B.57})$$

Aprox. 2º Orden

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{\Delta y}{2 A_{O|E}^X} [p_E - p_O] + \frac{1}{2} \frac{p_E - p_W}{2 A_{O|O}^X} \Delta y - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E - p_O}{2} \Delta y \quad (\text{B.58})$$

Corrección de presiones (Ec. continuidad)

$$A_O^p = \frac{\rho_w \Delta y^2}{2} \left[\frac{1}{A_{O|W}^X} + \frac{1}{A_{O|O}^X} \right] + \frac{\rho_n \Delta x^2}{2} \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] + \frac{\rho_s \Delta x^2}{2} \left[\frac{1}{A_{O|S}^Y} + \frac{1}{A_{O|O}^Y} \right] \quad (\text{B.59})$$

$$A_E^p = 0 \quad (\text{B.60})$$

$$S_O^p = - [(-\rho_w \hat{u}_w) \Delta y + (\rho_n \hat{v}_n - \rho_s \hat{v}_s) \Delta x] \quad (\text{B.61})$$

Actualización de velocidades

$$u_e^{k+1} = 0 \quad (\text{B.62})$$

B.4 Pared inferior sin deslizamiento

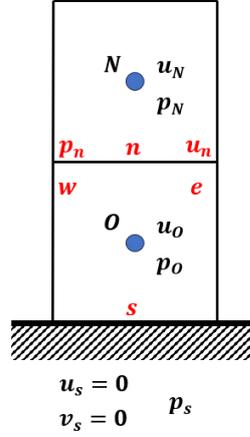


Figura B.5 Pared inferior sin deslizamiento.

Ecuación de cantidad de movimiento en X

$$\begin{aligned}
 A_O^X = & \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \\
 & + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{3\mu_O}{\Delta y} \right] \Delta x
 \end{aligned} \tag{B.63}$$

$$A_N^X = - \left[\frac{|(\rho v)_n| - (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{\mu_O}{3\Delta y} \right] \Delta x \tag{B.64}$$

$$A_S^X = 0 \tag{B.65}$$

Ecuación de cantidad de movimiento en Y

$$\begin{aligned}
 A_O^Y = & \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \\
 & + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{3\mu_O}{\Delta y} \right] \Delta x
 \end{aligned} \tag{B.66}$$

$$A_N^Y = - \left[\frac{|(\rho v)_n| - (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{3\mu_O}{\Delta y} \right] \Delta x \tag{B.67}$$

$$A_S^Y = 0 \tag{B.68}$$

$$\text{Aprox. 1º Orden} \quad S_O^Y = \frac{p_O - p_N}{2} \Delta x \quad (\text{B.69})$$

$$\text{Aprox. 2º Orden} \quad S_O^Y = (p_O - p_N) \Delta x \quad (\text{B.70})$$

Interpolación PWIM para velocidades en las caras

Aprox. 1º Orden

$$\hat{v}_n = \frac{\hat{v}_O + \hat{v}_N}{2} + \frac{1}{2} \frac{p_{NN} - p_O}{2 A_{O|N}^Y} \Delta y + \frac{\Delta x}{2 A_{O|O}^Y} \left[\frac{p_N - p_O}{2} \right] - \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] \frac{p_N - p_O}{2} \Delta y \quad (\text{B.71})$$

Aprox. 2º Orden

$$\hat{v}_n = \frac{\hat{v}_O + \hat{v}_N}{2} + \frac{1}{2} \frac{p_{NN} - p_O}{2 A_{O|N}^Y} \Delta y + \frac{\Delta x}{2 A_{O|O}^Y} [p_N - p_O] - \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] \frac{p_N - p_O}{2} \Delta y \quad (\text{B.72})$$

$$\hat{v}_s|_O = 0 \quad (\text{B.73})$$

Corrección de presiones (Ec. continuidad)

$$A_O^p = \frac{\rho_e \Delta y^2}{2} \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] + \frac{\rho_w \Delta y^2}{2} \left[\frac{1}{A_{O|W}^X} + \frac{1}{A_{O|O}^X} \right] + \frac{\rho_n \Delta x^2}{2} \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] \quad (\text{B.74})$$

$$A_S^p = 0 \quad (\text{B.75})$$

$$S_O^p = -[(\rho_e \hat{u}_e - \rho_w \hat{u}_w) \Delta y + (\rho_n \hat{v}_n) \Delta x] \quad (\text{B.76})$$

Actualización de velocidades

$$v_s|_O = 0 \quad (\text{B.77})$$

B.5 Pared superior sin deslizamiento

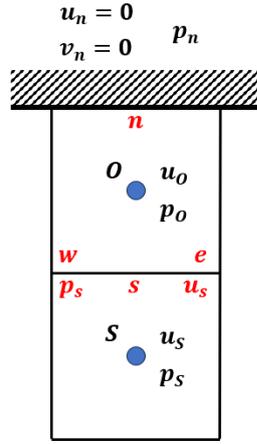


Figura B.6 Pared superior sin deslizamiento.

Ecuación de cantidad de movimiento en X

$$A_O^X = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \left[\frac{3\mu_O}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.78})$$

$$A_N^X = 0 \quad (\text{B.79})$$

$$A_S^X = - \left[\frac{|(\rho v)_s| + (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} + \frac{\mu_O}{3\Delta y} \right] \Delta x \quad (\text{B.80})$$

Ecuación de cantidad de movimiento en Y

$$A_O^Y = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \left[\frac{3\mu_O}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.81})$$

$$A_N^Y = 0 \quad (\text{B.82})$$

$$A_S^Y = - \left[\frac{|(\rho v)_s| + (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} + \frac{\mu_O}{3\Delta y} \right] \Delta x \quad (\text{B.83})$$

$$S_O^y = \frac{1}{2}(p_S - p_N) \Delta x \quad (\text{B.84})$$

$$\text{Aprox. 1}^\circ \text{ Orden} \quad S_O^y = \frac{p_S - p_O}{2} \Delta x \quad (\text{B.85})$$

$$\text{Aprox. 2}^\circ \text{ Orden} \quad S_O^y = (p_S - p_O) \Delta x \quad (\text{B.86})$$

Interpolación PWIM para velocidades en las caras

$$\hat{v}_n = 0 \quad (\text{B.87})$$

Para la fila $j = m - 1$:

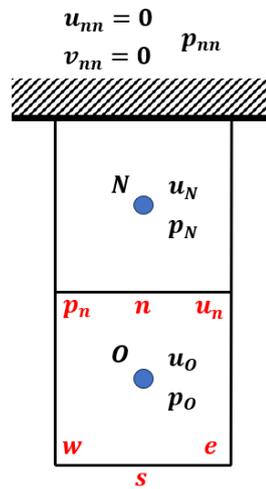


Figura B.7 Celda cerca de la pared superior.

Aprox. 1º Orden

$$\hat{v}_n = \frac{\hat{v}_O + \hat{v}_N}{2} + \frac{\Delta x}{2 A_{O|N}^y} \left[\frac{p_N - p_O}{2} \right] + \frac{1}{2} \frac{p_N - p_S}{2 A_{O|O}^y} \Delta y - \left[\frac{1}{A_{O|N}^y} + \frac{1}{A_{O|O}^y} \right] \frac{p_N - p_O}{2} \Delta y \quad (\text{B.88})$$

Aprox. 2º Orden

$$\hat{v}_n = \frac{\hat{v}_O + \hat{v}_N}{2} + \frac{\Delta x}{2 A_{O|N}^y} [p_N - p_O] + \frac{1}{2} \frac{p_N - p_S}{2 A_{O|O}^y} \Delta y - \left[\frac{1}{A_{O|N}^y} + \frac{1}{A_{O|O}^y} \right] \frac{p_N - p_O}{2} \Delta y \quad (\text{B.89})$$

Corrección de presiones (Ec. continuidad)

$$A_O^p = \frac{\rho_e \Delta y^2}{2} \left[\frac{1}{A_O^X|_E} + \frac{1}{A_O^X|_O} \right] + \frac{\rho_w \Delta y^2}{2} \left[\frac{1}{A_O^X|_W} + \frac{1}{A_O^X|_O} \right] + \frac{\rho_s \Delta x^2}{2} \left[\frac{1}{A_O^Y|_S} + \frac{1}{A_O^Y|_O} \right] \quad (\text{B.90})$$

$$A_N^p = 0 \quad (\text{B.91})$$

$$S_O^p = -[(\rho_e \hat{u}_e - \rho_w \hat{u}_w) \Delta y + (-\rho_s \hat{v}_s) \Delta x] \quad (\text{B.92})$$

Actualización de presiones

$$v_n^{k+1} = 0 \quad (\text{B.93})$$

B.6 Pared superior con deslizamiento

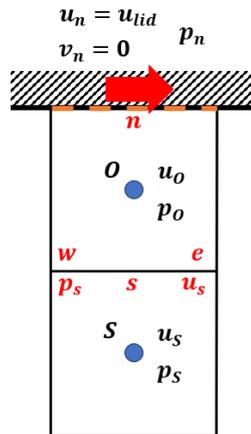


Figura B.8 Pared superior con deslizamiento.

Ecuación de cantidad de movimiento en X

$$A_O^X \hat{u}_O + A_W^X \hat{u}_W + A_E^X \hat{u}_E + A_N^X \hat{u}_N + A_S^X \hat{u}_S = S_O^X + \mu_O \frac{8\Delta x}{3\Delta y} u_{lid} \quad (\text{B.94})$$

$$A_O^X = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \left[\frac{3\mu_O}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.95})$$

$$A_N^X = 0 \quad (\text{B.96})$$

$$A_S^X = - \left[\frac{|(\rho v)_s| + (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} + \frac{\mu_O}{3\Delta y} \right] \Delta x \quad (\text{B.97})$$

Ecuación de cantidad de movimiento en Y

$$A_O^Y = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \left[\frac{3\mu_O}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.98})$$

$$A_N^Y = 0 \quad (\text{B.99})$$

$$A_S^Y = - \left[\frac{|(\rho v)_s| + (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} + \frac{\mu_o}{3\Delta y} \right] \Delta x \quad (\text{B.100})$$

$$S_O^Y = \frac{1}{2} (p_S - p_N) \Delta x \quad (\text{B.101})$$

$$\text{Aprox. 1}^\circ \text{ Orden} \quad S_O^Y = \frac{p_S - p_O}{2} \Delta x \quad (\text{B.102})$$

$$\text{Aprox. 2}^\circ \text{ Orden} \quad S_O^Y = (p_S - p_O) \Delta x \quad (\text{B.103})$$

Interpolación PWIM para velocidades en las caras

$$\hat{v}_n = 0 \quad (\text{B.104})$$

Para la fila $j = m - 1$:

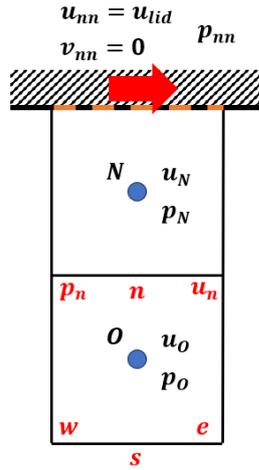


Figura B.9 Celda cerca de la pared superior con deslizamiento.

Aprox. 1º Orden

$$\hat{v}_n = \frac{\hat{v}_O + \hat{v}_N}{2} + \frac{\Delta x}{2 A_{O|N}^Y} \left[\frac{p_N - p_O}{2} \right] + \frac{1}{2} \frac{p_N - p_S}{2 A_{O|O}^Y} \Delta y - \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] \frac{p_N - p_O}{2} \Delta y \quad (\text{B.105})$$

Aprox. 2º Orden

$$\hat{v}_n = \frac{\hat{v}_O + \hat{v}_N}{2} + \frac{\Delta x}{2 A_{O|N}^Y} [p_N - p_O] + \frac{1}{2} \frac{p_N - p_S}{2 A_{O|O}^Y} \Delta y - \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] \frac{p_N - p_O}{2} \Delta y \quad (\text{B.106})$$

Corrección de presiones (Ec. continuidad)

$$A_O^p = \frac{\rho_e \Delta y^2}{2} \left[\frac{1}{A_O^X|_E} + \frac{1}{A_O^X|_O} \right] + \frac{\rho_w \Delta y^2}{2} \left[\frac{1}{A_O^X|_W} + \frac{1}{A_O^X|_O} \right] + \frac{\rho_s \Delta x^2}{2} \left[\frac{1}{A_O^Y|_S} + \frac{1}{A_O^Y|_O} \right] \quad (\text{B.107})$$

$$A_N^p = 0 \quad (\text{B.108})$$

$$S_O^p = -[(\rho_e \hat{u}_e - \rho_w \hat{u}_w) \Delta y + (-\rho_s \hat{v}_s) \Delta x] \quad (\text{B.109})$$

Actualización de presiones

$$v_n^{k+1} = 0 \quad (\text{B.110})$$

B.7 Entrada a velocidad constante

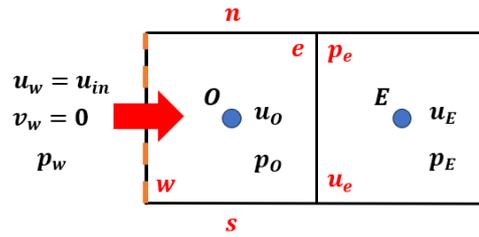


Figura B.10 Entrada del fluido a velocidad constante.

Ecuación de cantidad de movimiento en X

$$A_O^X \hat{u}_O + A_W^X \hat{u}_W + A_E^X \hat{u}_E + A_N^X \hat{u}_N + A_S^X \hat{u}_S = S_O^X + \left[\rho_{in} u_{in} u_{in} + \frac{8\mu_{in}}{3\Delta x} u_{in} \right] \Delta y \quad (\text{B.111})$$

$$A_O^X = \left[\frac{|\rho u|_e + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{3\mu_{in}}{\Delta x} \right] \Delta y + \left[\frac{|\rho v|_n + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|\rho v|_s - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.112})$$

$$A_E^X = - \left[\frac{|\rho u|_e - (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{\mu_{in}}{3\Delta x} \right] \Delta y \quad (\text{B.113})$$

$$A_W^X = 0 \quad (\text{B.114})$$

$$\text{Aprox. 1º Orden} \quad S_O^X = \frac{p_O - p_E}{2} \Delta y \quad (\text{B.115})$$

$$\text{Aprox. 2º Orden} \quad S_O^X = (p_O - p_E) \Delta y \quad (\text{B.116})$$

Ecuación de cantidad de movimiento en Y

$$A_O^Y = \left[\frac{|\rho u|_e + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{3\mu_{in}}{\Delta x} \right] \Delta y + \left[\frac{|\rho v|_n + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|\rho v|_s - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.117})$$

$$A_E^Y = - \left[\frac{|\rho u|_e - (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{\mu_{in}}{3\Delta x} \right] \Delta y \quad (\text{B.118})$$

$$A_W^Y = 0 \quad (\text{B.119})$$

Interpolación PWIM para velocidades en las caras

Aprox. 1º Orden

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{1}{2} \frac{p_{EE} - p_O}{2 A_{O|E}^X} \Delta y + \frac{\Delta y}{2 A_{O|O}^X} \left[\frac{p_E - p_O}{2} \right] - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E - p_O}{2} \Delta y \quad (\text{B.120})$$

Aprox. 2º Orden

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{1}{2} \frac{p_{EE} - p_O}{2 A_{O|E}^X} \Delta y + \frac{\Delta y}{2 A_{O|O}^X} [p_E - p_O] - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E - p_O}{2} \Delta y \quad (\text{B.121})$$

$$\hat{u}_w|_O = u_{in} \quad (\text{B.122})$$

Corrección de presiones (Ec. continuidad)

$$A_O^p = \frac{\rho_e \Delta y^2}{2} \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] + \frac{\rho_n \Delta x^2}{2} \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] + \frac{\rho_s \Delta x^2}{2} \left[\frac{1}{A_{O|S}^Y} + \frac{1}{A_{O|O}^Y} \right] \quad (\text{B.123})$$

$$A_W^p = 0 \quad (\text{B.124})$$

$$S_O^p = -[(\rho_e \hat{u}_e - \rho_{in} u_{in}) \Delta y + (\rho_n \hat{v}_n - \rho_s \hat{v}_s) \Delta x] \quad (\text{B.125})$$

Actualización de velocidades

$$u_w|_O = u_{in} \quad (\text{B.126})$$

B.8 Entrada a presión constante

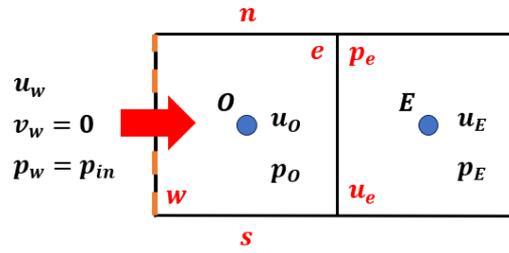


Figura B.11 Entrada del fluido a presión constante.

Ecuación de cantidad de movimiento en X

$$\text{Aprox. 1}^\circ \text{ Orden} \quad u_{in} = u_O^k \quad (\text{B.127})$$

$$\text{Aprox. 2}^\circ \text{ Orden} \quad u_{in} = \frac{3}{2} u_O^k - \frac{1}{2} u_E^k \quad (\text{B.128})$$

$$A_O^X \hat{u}_O + A_W^X \hat{u}_W + A_E^X \hat{u}_E + A_N^X \hat{u}_N + A_S^X \hat{u}_S = S_O^X + \left[\rho_{in} u_{in} u_{in} + \frac{8\mu_{in}}{3\Delta x} u_{in} \right] \Delta y \quad (\text{B.129})$$

$$A_O^X = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{3\mu_{in}}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.130})$$

$$A_E^X = - \left[\frac{|(\rho u)_e| - (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{\mu_{in}}{3\Delta x} \right] \Delta y \quad (\text{B.131})$$

$$A_W^X = 0 \quad (\text{B.132})$$

$$S_O^X = \left[p_{in} - \frac{p_O + p_E}{2} \right] \Delta y \quad (\text{B.133})$$

Ecuación de cantidad de movimiento en Y

$$A_O^Y = \left[\frac{|(\rho u)_e| + (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{3\mu_{in}}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (\text{B.134})$$

$$A_E^Y = - \left[\frac{|(\rho u)_e| - (\rho u)_e}{2} + \frac{\mu_e}{\Delta x} + \frac{\mu_{in}}{3\Delta x} \right] \Delta y \quad (\text{B.135})$$

$$A_W^Y = 0 \quad (\text{B.136})$$

Interpolación PWIM para velocidades en las caras

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{1}{2} \frac{p_{EE} - p_O}{2 A_{O|E}^X} \Delta y + \frac{\Delta y}{2 A_{O|O}^X} \left[\frac{p_E + p_O}{2} - p_{in} \right] - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E - p_O}{2} \Delta y \quad (\text{B.137})$$

$$\text{Aprox. 1º Orden} \quad \hat{u}_w|_O = \hat{u}_O \quad (\text{B.138})$$

$$\text{Aprox. 2º Orden} \quad \hat{u}_w|_O = \frac{3}{2} \hat{u}_O - \frac{1}{2} \hat{u}_E \quad (\text{B.139})$$

Corrección de presiones (Ec. continuidad)

$$A_W^p = 0 \quad (\text{B.140})$$

$$\text{Aprox. 1º Orden} \quad S_O^p = - [(\rho_e \hat{u}_e - \rho_{in} \hat{u}_O) \Delta y + (\rho_n \hat{v}_n - \rho_s \hat{v}_s) \Delta x] \quad (\text{B.141})$$

$$\text{Aprox. 2º Orden} \quad S_O^p = - \left[\left(\rho_e \hat{u}_e - \rho_{in} \left[\frac{3}{2} \hat{u}_O - \frac{1}{2} \hat{u}_E \right] \right) \Delta y + (\rho_n \hat{v}_n - \rho_s \hat{v}_s) \Delta x \right] \quad (\text{B.142})$$

Actualización de velocidades

$$\text{Aprox. 1º Orden} \quad u_w|_O = u_O \quad (\text{B.143})$$

$$\text{Aprox. 2º Orden} \quad u_w|_O = \frac{3}{2} u_O - \frac{1}{2} u_E \quad (\text{B.144})$$

B.9 Salida a presión constante

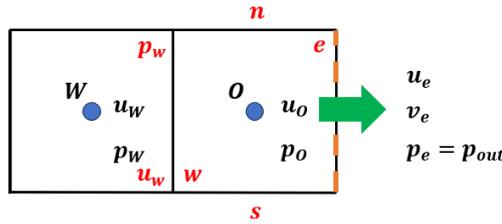


Figura B.12 Salida del fluido a presión constante.

Ecuación de cantidad de movimiento en X

$$A_O^X = \left[\rho_o u_o + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (B.145)$$

$$A_E^X = 0 \quad (B.146)$$

$$S_O^X = \left[\frac{p_o + p_w}{2} + p_{out} \right] \Delta y \quad (B.147)$$

Ecuación de cantidad de movimiento en Y

$$A_O^Y = \left[\frac{3\mu_o}{\Delta x} + \frac{|(\rho u)_w| - (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} \right] \Delta y + \left[\frac{|(\rho v)_n| + (\rho v)_n}{2} + \frac{\mu_n}{\Delta y} + \frac{|(\rho v)_s| - (\rho v)_s}{2} + \frac{\mu_s}{\Delta y} \right] \Delta x \quad (B.148)$$

$$A_E^Y = 0 \quad (B.149)$$

$$A_W^Y = - \left[\frac{|(\rho u)_w| + (\rho u)_w}{2} + \frac{\mu_w}{\Delta x} + \frac{\mu_o}{3\Delta x} \right] \Delta y \quad (B.150)$$

Interpolación PWIM para velocidades en las caras

$$\hat{u}_e = \hat{u}_O \quad (\text{B.151})$$

Para la columna $i = n - 1$:

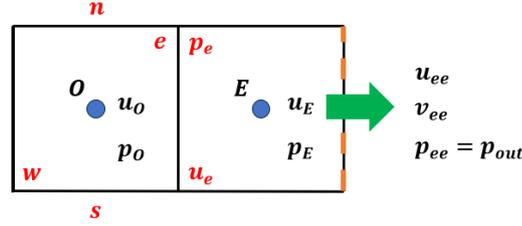


Figura B.13 Celda cerca de la salida del fluido a presión constante.

$$\hat{u}_e = \frac{\hat{u}_O + \hat{u}_E}{2} + \frac{\Delta y}{2 A_{O|E}^X} \left[p_{out} - \frac{p_O + p_E}{2} \right] + \frac{1}{2} \frac{p_E - p_W}{2 A_{O|O}^X} \Delta y - \left[\frac{1}{A_{O|E}^X} + \frac{1}{A_{O|O}^X} \right] \frac{p_E - p_O}{2} \Delta y \quad (\text{B.152})$$

Corrección de presiones (Ec. continuidad)

$$A_O^p = \frac{\rho_O \Delta y^2}{2 A_{O|O}^X} + \frac{\rho_w \Delta y^2}{2} \left[\frac{1}{A_{O|W}^X} + \frac{1}{A_{O|O}^X} \right] + \frac{\rho_n \Delta x^2}{2} \left[\frac{1}{A_{O|N}^Y} + \frac{1}{A_{O|O}^Y} \right] + \frac{\rho_s \Delta x^2}{2} \left[\frac{1}{A_{O|S}^Y} + \frac{1}{A_{O|O}^Y} \right] \quad (\text{B.153})$$

$$A_E^p = 0 \quad (\text{B.154})$$

$$A_W^p = -\frac{\rho_w \Delta y^2}{2} \left[\frac{1}{A_{O|W}^X} + \frac{1}{A_{O|O}^X} \right] + \frac{\rho_O \Delta y^2}{2 A_{O|O}^X} \quad (\text{B.155})$$

$$S_O^p = -[(\rho_O \hat{u}_O - \rho_w \hat{u}_w) \Delta y + (\rho_n \hat{v}_n - \rho_s \hat{v}_s) \Delta x] \quad (\text{B.156})$$

Actualización de velocidades

$$u_e^{k+1} = u_O^{k+1} \quad (\text{B.157})$$

Apéndice C

Códigos desarrollados

En este Anexo C se presentan los códigos principales implementados en Matlab para la resolución del caso de un canal de sección constante utilizando el algoritmo SIMPLE (Semi-Implicit Method for Pressure-Linked Equations). Cada sección de este anexo corresponde a una de las funciones descritas en el diagrama mostrado en la figura 2.3.

Los códigos proporcionados están ampliamente comentados para facilitar su comprensión y asegurar que sean autoexplicativos. Además, se destacan las secciones que deben modificarse según las condiciones de contorno específicas del problema que se desea resolver.

Los códigos se presentan en el siguiente orden:

1. Código principal del algoritmo SIMPLE
2. Función de cálculo de los coeficientes de enlace y fuentes para las ecuaciones de cantidad de movimiento
3. Función para resolución de ecuación de cantidad de movimiento en X
4. Función para resolución de ecuación de cantidad de movimiento en Y
5. Función PWIM de interpolación de velocidades en las caras
6. Función de cálculo de los coeficientes de enlace y fuentes para la ecuación de continuidad
7. Función para resolución de ecuación de continuidad
8. Función de actualización de velocidades y presiones
9. Función de post-procesado
10. Función auxiliar: TDMA

C.1 Código principal del algoritmo SIMPLE

```

=====
%% Programa principal
=====
% Este script contiene los parámetros principales del problema que se quiere
% resolver y habrá que configurarlo según el problema que se quiera
% solucionar.
%
% Se utilizará el algoritmo SIMPLE para resolver el dominio fluido que se
% configure en el script según las condiciones de contorno, usando un
% mallado estructurado y co-ubicado.
%
% Se crearán funciones auxiliares según lo indicado el en algoritmo SIMPLE
% para realizar cálculos específicos según el apartado.
=====
%% Problema 1 Canal longitudinal de sección constante

% El problema que se pretende resolver se trata de un conducto longitudinal
% de sección constante. El flujo entra con una velocidad horizontal u_in y
% sale a una presión conocida p_out

clear all, clc
=====
%% Parámetros del problema

% Dimensiones del dominio fluido
H=0.01;           % [m] Dimensión característica
Long_x=5*H;      % [m] Longitud total del conducto
Long_y=H;        % [m] Altura total del conducto

% Velocidad de entrada del fluido
U_in=0.1;        % [m/s] 0.1 0.2 0.3 0.4 0.45

% Presión de salida del fluido
p_out=0;         % [Pa] Aprox 1 atm
=====
%% Propiedades del fluido
% Se van a considerar constantes.

rho=1;           % Densidad [kg/m3]
gam=1e-5;        % Viscosidad [kg/(m s)]

% Reynolds (Basado en la altura total del conducto)
Re=rho*U_in*Long_y/gam;
=====

```

```

%% Mallado estructurado y co-ubicado
% Ortogonal y con todas las celdas iguales

n_celdas=20;           % La proporción es que cada longitud H, tenga 20 celdas
nx=n_celdas*Long_x/H; % Número de celdas totales en dirección x
ny=n_celdas*Long_y/H; % Número de celdas totales en dirección y

dx=Long_x/nx; % Cálculo de la longitud en dirección x de la celda [m]
dy=Long_y/ny; % Cálculo de la longitud en dirección y de la celda [m]
ar=dx/dy;     % Relación de aspecto
ra=1/ar;     % Inverso de la relación de aspecto

% Centro de las celdas
centro_x=dx/2:dx:(Long_x-dx/2); % En dirección x [m]
centro_y=dy/2:dy:(Long_y-dy/2); % En dirección y [m]

=====
%% Iteracciones, tolerancias y factores de amortiguamiento

%Iteracciones
iter_CdM=2; % Iteracciones para resolver Ec. Cant. de Movimiento.
iter_P=20;  % Iteracciones para resolver la Corrección de presiones.
iter_global=2000; % Iteracciones máximas para resolver todo el problema.

% Tolerancias

% Se definen una serie de tolerancias con las que una vez alcanzadas, se
% considera que la solución a convergido, sin tener que llegar al número
% máximo de iteracciones.

tol_interna=1e-3; % Tolerancia para Ec. Cant. Movimiento y Presiones.
tol_externa=1e-10; % Tolerancia para el problema completo.

% Factores de amortiguamiento
omega_uv=0.8; % Factor de amortiguamiento de la velocidad
omega_p=0.2; % Factor de amortiguamiento de la presión
omega_inercial_uv=0.2; % Factor de amortiguamiento de la inercia para la velocidad

=====
%% Condiciones de contorno

u_entrada=U_in; v_entrada=0; % Entrada al dominio
u_izq=0; v_izq=0; % Pared izquierda
u_inf=0; v_inf=0; % Pared inferior
u_sup=0; v_sup=0; % Pared superior
p_salida=p_out; % Salida del dominio

=====

```

```

%% Inicialización de variables

% Incognitas
% Velocidades y presión en las celdas (centro)
u(1:nx,1:ny)=u_entrada*ones(nx,ny);    % Velocidad horizontal [m/s]
v(1:nx,1:ny)=zeros;                    % Velocidad vertical [m/s]
p(1:nx,1:ny)=p_out*ones(nx,ny);        % Presión manométrica [Pa]

% Velocidades en las caras de las celdas
u_fw(1:nx,1:ny)=u_entrada*ones(nx,ny); % Velocidad horizontal Cara w [m/s]
u_fe(1:nx,1:ny)=u_entrada*ones(nx,ny); % Velocidad horizontal Cara e [m/s]
v_fs(1:nx,1:ny)=zeros;                 % Velocidad vertical Cara s [m/s]
v_fn(1:nx,1:ny)=zeros;                 % Velocidad vertical Cara n [m/s]

% Velocidad y presión en las celdas para la corrección
u_som(1:nx,1:ny)=zeros;    % Vel. horizontal calculada en Ec. Cant Mov. [m/s]
v_som(1:nx,1:ny)=zeros;    % Vel. vertical calculada en Ec. Cant Mov. [m/s]
p_cc(1:nx,1:ny)=zeros;     % Presión de corrección p' [Pa]

% Velocidades en las caras de las celdas interpolada con PWIM para la
% corrección de la presión
u_fw_som(1:nx,1:ny)=zeros; % Velocidad horizontal Cara w interpolada [m/s]
u_fe_som(1:nx,1:ny)=zeros; % Velocidad horizontal Cara e interpolada [m/s]
v_fs_som(1:nx,1:ny)=zeros; % Velocidad vertical Cara s interpolada [m/s]
v_fn_som(1:nx,1:ny)=zeros; % Velocidad vertical Cara n interpolada [m/s]

% Coeficientes de enlace y fuentes
% Ec. Momento en X
Aox(1:nx,1:ny)=zeros;
Awx(1:nx,1:ny)=zeros;
Aex(1:nx,1:ny)=zeros;
Anx(1:nx,1:ny)=zeros;
Asx(1:nx,1:ny)=zeros;
Sx(1:nx,1:ny)=zeros;

% Ec. Momento en Y
Aoy(1:nx,1:ny)=zeros;
Awy(1:nx,1:ny)=zeros;
Aey(1:nx,1:ny)=zeros;
Any(1:nx,1:ny)=zeros;
Asy(1:nx,1:ny)=zeros;
Sy(1:nx,1:ny)=zeros;

% Ec. corrección de presiones
Aop(1:nx,1:ny)=zeros;
Awp(1:nx,1:ny)=zeros;
Aep(1:nx,1:ny)=zeros;
Anp(1:nx,1:ny)=zeros;
Asp(1:nx,1:ny)=zeros;
Sp(1:nx,1:ny)=zeros;

```

```

% Residuos
Res_u=NaN(1, iter_global);    % Residuo para Ec. Momento en X
Res_v=NaN(1, iter_global);    % Residuo para Ec. Momento en Y
Res_p=NaN(1, iter_global);    % Residuo para Corrección de presiones

% Densidad
rho_o(1:nx,1:ny)=rho;        % Densidad [kg/m3]

=====
%% Configuración de la gráfica
% Se configura la gráfica para ir representando los residuos según se va
% resolviendo las ecuaciones en las diferentes iteraciones para comprobar
% que los resultados convergen.

figure;
xlabel('Iteración');
ylabel('Residuos');
title(['Evolución de los residuos Re = ' num2str(Re)]);
set(gca, 'YScale', 'log');
grid on;
hold on;

% Inicialización de la línea para representar los residuos
Curva_res_u = plot(1:iter_global, Res_u, 'b.-');
Curva_res_v = plot(1:iter_global, Res_v, 'r.-');
Curva_res_p = plot(1:iter_global, Res_p, 'g.-');
legend('Residual u', 'Residual v', 'Residual p');

=====
%% Algoritmo SIMPLE para la resolución

% Se llaman a cada una de las funciones para resolver el dominio
% fluido de forma iterativa.

% Se inicializan los valores para entrar al bucle.
k=1;
Res_u_ultimo=1;
Res_v_ultimo=1;
Res_p_ultimo=1;

% Bucle externo
while k<iter_global &&...           % Nº Iteraciones máximo
    (Res_u_ultimo>=tol_externa || ... % Tolerancia externa no alcanzada
    Res_v_ultimo>=tol_externa ||... % en los 3 residuales.
    Res_p_ultimo>=tol_externa)

% Se muestra por pantalla el número de la iteración
disp(['Nº Iteración ' num2str(k)]);

```

```

% Funciones por fases

% Cálculo de coeficientes de enlace y fuentes para Ec. Cant. Movimiento
% [AFECTADA POR COND. CONTORNO]
[Aox, Awx, Aex, Anx, Asx, Aoy, Aoy, Awy, Aey, Any, Asy,...
Sx, Sy]=ECM_cal_coef_enlace_y_fuentes(p, p_out, u, u_fw, u_fe,...
v_fs, v_fn, dx, dy, rho_o, gam);

% Resolución Ec. Cant. Mov. en X [AFECTADA POR COND. CONTORNO]
[u_som, Res_u_ultimo]=sol_ecm_x(u, p, Aox, Awx, Aex, Anx, Asx,...
Sx, dx, dy, iter_CdM, omega_inercial_uv, gam, u_entrada, rho_o);

% Resolución Ec. Cant. Mov. en Y [AFECTADA POR COND. CONTORNO]
[v_som, Res_v_ultimo]=sol_ecm_y(v, p, Aoy, Awy, Aey, Any, Asy,...
Sy, dx, iter_CdM, omega_inercial_uv);

% Interpolación en las caras con PWIM [AFECTADA POR COND. CONTORNO]
[u_fw_som,u_fe_som,v_fs_som,v_fn_som]=Interp_PWIM(u_som,v_som, p,...
Aox, Aoy, dx, dy, u_entrada, v_sup, v_inf, p_out);

% Cálculo de coeficientes de enlace y fuentes para Corrección de
% presión (Ec. Continuidad) [AFECTADA POR COND. CONTORNO]
[Aop, Awp, Aep, Anp, Asp, Sp]=CPress_cal_coef_enlace_y_fuentes(rho_o,...
u_entrada, u_som, u_fw_som,u_fe_som,v_fs_som,v_fn_som,...
Aox, Aoy, dx, dy);

% Resolución Corrección de presión
[p_cc, Res_p_ultimo]=sol_cpess(Aop, Awp, Aep, Anp, Asp, Sp, iter_P);

% Actualización de velocidades [AFECTADA POR COND. CONTORNO]
[u,v,u_fw,u_fe,v_fs,v_fn]=actualizar_vel(u_som, v_som, u_fe_som,...
v_fn_som, p_cc, Aox, Aoy, dx, dy, omega_uv, u_entrada, v_sup, v_inf);

% Actualización de presiones
[p]=actualizar_p(p, p_cc, omega_p);

% Se actualizan los residuos en el vector.
Res_u(k)=Res_u_ultimo;
Res_v(k)=Res_v_ultimo;
Res_p(k)=Res_p_ultimo;

disp(['Residuo u ' num2str(Res_u_ultimo)]);
disp(['Residuo v ' num2str(Res_v_ultimo)]);
disp(['Residuo p ' num2str(Res_p_ultimo)]);
disp(' ')

% Actualiza la línea con los nuevos datos
set(Curva_res_u, 'YData', Res_u);
set(Curva_res_v, 'YData', Res_v);
set(Curva_res_p, 'YData', Res_p);

```

```
%Pausa para visualizar en tiempo real (ajusta según sea necesario)
pause(0.1);

k=k+1; % Se actualiza el número de la iteración.

end

hold off;

=====
%% Guardar valores para Post-procesado

NumIteraciones=k-1;
% Se organizan los Residuos en un vector
Residuos=[(1:NumIteraciones); Res_u(1:NumIteraciones);...
          Res_v(1:NumIteraciones); Res_p(1:NumIteraciones)]';

% Guardar en archivo .mat los resultados para su postprocesado
save(['Resultados_Re_', num2str(Re), '.mat'], 'Re','NumIteraciones',...
     'Residuos', 'u', 'v', 'p');
disp(['Resultados guardados Re = ', num2str(Re) ])
```

C.2 Función de cálculo de los coeficientes de enlace y fuentes para las ecuaciones de cantidad de movimiento

```

%=====
%% Calculo de coef. Enlace y Fuentes para Ec. Cant. Movimiento
%=====
% Problema de canal longitudinal de sección constante
%=====

function [Aox, Awx, Aex, Anx, Asx, Aoy, Awy, Aey, Any, Asy,...
         Sx, Sy]=ECM_cal_coef_enlace_y_fuentes(p, p_out, u, u_fw, u_fe,...
         v_fs, v_fn, dx, dy, rho_o, gam)
%=====
% Se reorganizan las presiones y densidades para facilitar la programación
% de las expresiones

%% Presiones

[nx, ny]=size(p); % Se saca las dimensiones de la matriz;

p_W=zeros(nx,ny); p_W(2:nx,1:ny)=p(1:nx-1,1:ny);
p_E=zeros(nx,ny); p_E(1:nx-1,1:ny)=p(2:nx,1:ny);
p_S=zeros(nx,ny); p_S(1:nx,2:ny)=p(1:nx,1:ny-1);
p_N=zeros(nx,ny); p_N(1:nx,1:ny-1)=p(1:nx,2:ny);

%% Densidad
rho_fw=rho_o;
rho_fe=rho_o;
rho_fn=rho_o;
rho_fs=rho_o;

%=====
%% Inicialización

% Hay que inicializar los valores de los coeficientes de enlace y fuentes
% para guardar espacio en la memoria.

% Ec. Momento en X
Aox(1:nx,1:ny)=zeros;
Awx(1:nx,1:ny)=zeros;
Aex(1:nx,1:ny)=zeros;
Anx(1:nx,1:ny)=zeros;
Asx(1:nx,1:ny)=zeros;
Sx(1:nx,1:ny)=zeros;

% Ec. Momento en Y
Aoy(1:nx,1:ny)=zeros;
Awy(1:nx,1:ny)=zeros;
Aey(1:nx,1:ny)=zeros;

```

```

Any(1:nx,1:ny)=zeros;
Asy(1:nx,1:ny)=zeros;
Sy(1:nx,1:ny)=zeros;

%=====
%% Cálculo de coeficientes y fuentes

% [CONDICIONES DE CONTORNO]
% Dependen de las condiciones de contorno establecidas hay que distinguir
% la posición en la que se encuentra la celda.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
j=1; % Primera fila / Pared inferior

        i=1; % Zona 4 & 1 | Esquina inferior izquierda

                % Coef. y fuente para X
                ce=rho_fe(i,j)*u_fe(i,j);
                cw=rho_fw(i,j)*u_fw(i,j);
                cn=rho_fn(i,j)*v_fn(i,j);
                cs=rho_fs(i,j)*v_fs(i,j);
                % Por condición de contorno al aproximar, se añade el
                % término gam/(3*dx));
                Aex(i,j)=-((abs(ce)-ce)/2+gam/dx+gam/(3*dx))*dy;
                % Por condición de contorno, se anula este término
                Awx(i,j)=0;
                % Por condición de contorno al aproximar, se añade el
                % término gam/(3*dy)
                Anx(i,j)=-((abs(cn)-cn)/2+gam/dy+gam/(3*dy))*dx;
                % Por condición de contorno, u_fs=0 y v_fs=0, se anula
                % este término
                Asx(i,j)=0;
                % Por condición de contorno, Awx(i,j) se sustituye por
                % 3*gam/dx*dy y Asx(i,j) se sustituye por 3*gam/dy*dx
                Aox(i,j)=((abs(ce)+ce)/2+gam/dx+3*gam/dx)*dy+...
                        ((abs(cn)+cn)/2+gam/dy+3*gam/dy)*dx;
                % Por condición de contorno p_fw se aproxima en
                % 2º Orden por 3/2*p_o-1/2*p_E
                Sx(i,j)=(p(i,j)-p_E(i,j))*dy;          % Fuente X

                % Coef. y fuente para Y
                % Para este caso concreto, los coef. X e Y son iguales.
                Aoy(i,j)=Aox(i,j);
                Aey(i,j)=Aex(i,j);
                Awy(i,j)=Awx(i,j);
                Any(i,j)=Anx(i,j);
                Asy(i,j)=Asx(i,j);
                % Por condición de contorno p_fs se aproxima en
                % 2º Orden por 3/2*p_o-1/2*p_N
                Sy(i,j)=(p(i,j)-p_N(i,j))*dx;          % Fuente Y

```

```

for i=2:nx-1 % Zona 4 | Columnas interiores / Pared inferior

    % Coef. y fuente para X
    ce=rho_fe(i,j)*u_fe(i,j);
    cw=rho_fw(i,j)*u_fw(i,j);
    cn=rho_fn(i,j)*v_fn(i,j);
    cs=rho_fs(i,j)*v_fs(i,j);
    Aex(i,j)=-((abs(ce)-ce)/2+gam/dx)*dy;
    Awx(i,j)=-((abs(cw)+cw)/2+gam/dx)*dy;
    % Por condición de contorno al aproximar, se añade el
    % término gam/(3*dy)
    Anx(i,j)=-((abs(cn)-cn)/2+gam/dy+gam/(3*dy))*dx;
    % Por condición de contorno, u_fs=0 y v_fs=0, se anula
    % este término
    Asx(i,j)=0;
    % Por condición de contorno, Asx(i,j) se sustituye por
    % 3*gam/dy*dx y se suma (abs(cn)-cn)/2+gam/dy
    Aox(i,j)=( (abs(ce)+ce)/2+gam/dx +...
                (abs(cw)-cw)/2+gam/dx)*dy +...
                ((abs(cn)+cn)/2+gam/dy+3*gam/dy)*dx;
    Sx(i,j)=1/2*(p_W(i,j)-p_E(i,j))*dy;      % Fuente X

    % Coef. y fuente para Y
    % Para este caso concreto, los coef. X e Y son iguales.
    Aoy(i,j)=Aox(i,j);
    Aey(i,j)=Aex(i,j);
    Awy(i,j)=Awx(i,j);
    Any(i,j)=Anx(i,j);
    Asy(i,j)=Asx(i,j);
    % Por condición de contorno p_fs se aproxima en
    % 2º Orden por 3/2*p_o-1/2*p_N
    Sy(i,j)=(p(i,j)-p_N(i,j))*dx;      % Fuente Y

end

i=nx; % Zona 4 & 3 | Salida

    % Coef. y fuente para X
    ce=rho_fe(i,j)*u_fe(i,j);
    cw=rho_fw(i,j)*u_fw(i,j);
    cn=rho_fn(i,j)*v_fn(i,j);
    cs=rho_fs(i,j)*v_fs(i,j);
    % Por condición de contorno, Aex se anula.
    Aex(i,j)=0;
    Awx(i,j)=-((abs(cw)+cw)/2+gam/dx)*dy;
    % Por condición de contorno al aproximar, se añade el
    % término gam/(3*dy)
    Anx(i,j)=-((abs(cn)-cn)/2+gam/dy+gam/(3*dy))*dx;
    % Por condición de contorno, u_fs=0 y v_fs=0, se anula
    % este término

```

```

Asx(i,j)=0;
% Por condición de contorno Aex se sustituye por
% rho_o(i,j)*u(i,j)*dy
Aox(i,j)=rho_o(i,j)*u(i,j)*dy + ((abs(cw)-cw)/2+gam/dx)*dy...
      +((abs(cn)+cn)/2+gam/dy+3*gam/dy)*dx ;
% Por condición de contorno p_fe = p_out presión de la salida
Sx(i,j)=((p(i,j)+p_W(i,j))/2-p_out)*dy;      % Fuente X

% Coef. y fuente para Y
% Por condición de contorno, v_fe=0, se anula este término
Aey(i,j)=0;
% Por condición de contorno al aproximar, se añade el
% término gam/(3*dx)
Awy(i,j)=-((abs(cw)+cw)/2+gam/dx+gam/(3*dx))*dy;
% Son iguales a los términos en x
Any(i,j)=Anx(i,j);
% Son iguales a los términos en x
Asy(i,j)=0;
% Por condición de contorno, Aey(i,j) se sustituye por
% 3*gam/dx*dy, más ((abs(cw)-cw)/2+gam/dx
Aoy(i,j)=((abs(cw)-cw)/2+gam/dx+3*gam/dx)*dy + ...
      ((abs(cn)+cn)/2+gam/dy+3*gam/dy)*dx ;
% Por condición de contorno p_fs se aproxima en
% 2º Orden por 3/2*p_o-1/2*p_N
Sy(i,j)=(p(i,j)-p_N(i,j))*dx;      % Fuente Y

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for j=2:ny-1 % Filas entre las paredes inferiores y superior

i=1; % Zona 1 | Entrada
% Coef. y fuente para X
ce=rho_fe(i,j)*u_fe(i,j);
cw=rho_fw(i,j)*u_fw(i,j);
cn=rho_fn(i,j)*v_fn(i,j);
cs=rho_fs(i,j)*v_fs(i,j);
% Por condición de contorno al aproximar, se añade el
% término gam/(3*dx));
Aex(i,j)=-((abs(ce)-ce)/2+gam/dx+gam/(3*dx))*dy;
% Por condición de contorno, u_fw=0 y v_fw=0, se anula
% este término
Awx(i,j)=0;
Anx(i,j)=-((abs(cn)-cn)/2+gam/dy)*dx;
Asx(i,j)=-((abs(cs)+cs)/2+gam/dy)*dx;
% Por condición de contorno, Awx(i,j) se sustituye por
% 3*gam/dx*dy, más ((abs(ce)-ce)/2+gam/dx
Aox(i,j)=((abs(ce)+ce)/2+gam/dx+3*gam/dx)*dy + ...
      ((abs(cn)+cn)/2+gam/dy + (abs(cs)-cs)/2+gam/dy)*dx;
% Por condición de contorno p_fw se aproxima en
% 2º Orden por 3/2*p_o-1/2*p_E
Sx(i,j)=(p(i,j)-p_E(i,j))*dy;      % Fuente X

```

```

% Coef. y fuente para Y
% Para este caso concreto, los coef. X e Y son iguales.
Aoy(i,j)=Aox(i,j);
Aey(i,j)=Aex(i,j);
Awy(i,j)=Awx(i,j);
Any(i,j)=Anx(i,j);
Asy(i,j)=Asx(i,j);
Sy(i,j)=1/2*(p_S(i,j)-p_N(i,j))*dx;      % Fuente Y

for i=2:nx-1 % Zona interior | Celdas interiores

% Coef. y fuente para X
ce=rho_fe(i,j)*u_fe(i,j);
cw=rho_fw(i,j)*u_fw(i,j);
cn=rho_fn(i,j)*v_fn(i,j);
cs=rho_fs(i,j)*v_fs(i,j);
Aex(i,j)=-((abs(ce)-ce)/2+gam/dx)*dy;
Awx(i,j)=-((abs(cw)+cw)/2+gam/dx)*dy;
Anx(i,j)=-((abs(cn)-cn)/2+gam/dy)*dx;
Asx(i,j)=-((abs(cs)+cs)/2+gam/dy)*dx;
Aox(i,j)= ((abs(ce)+ce)/2+gam/dx + (abs(cw)-cw)/2+gam/dx)*dy...
          +((abs(cn)+cn)/2+gam/dy + (abs(cs)-cs)/2+gam/dy )*dx ;
Sx(i,j)=1/2*(p_W(i,j)-p_E(i,j))*dy;      % Fuente X

% Coef. y fuente para Y
% Para este caso concreto, los coef. X e Y son iguales.
Aoy(i,j)=Aox(i,j);
Aey(i,j)=Aex(i,j);
Awy(i,j)=Awx(i,j);
Any(i,j)=Anx(i,j);
Asy(i,j)=Asx(i,j);
Sy(i,j)=1/2*(p_S(i,j)-p_N(i,j))*dx;      % Fuente Y
end

i=nx; % Zona 3 | Salida

% Coef. y fuente para X
ce=rho_fe(i,j)*u_fe(i,j);
cw=rho_fw(i,j)*u_fw(i,j);
cn=rho_fn(i,j)*v_fn(i,j);
cs=rho_fs(i,j)*v_fs(i,j);
% Por condición de contorno, Aex se anula.
Aex(i,j)=0;
Awx(i,j)=-((abs(cw)+cw)/2+gam/dx)*dy;
Anx(i,j)=-((abs(cn)-cn)/2+gam/dy)*dx;
Asx(i,j)=-((abs(cs)+cs)/2+gam/dy)*dx;
% Por condición de contorno Aex se sustituye
% por rho_o(i,j)*u(i,j)*dy

```

```

Aox(i,j)=rho_o(i,j)*u(i,j)*dy + ((abs(cw)-cw)/2+gam/dx)*dy...
      +((abs(cn)+cn)/2+gam/dy + (abs(cs)-cs)/2+gam/dy)*dx;
% Por condición de contorno p_fe = p_out presión de la salida
Sx(i,j)=((p(i,j)+p_W(i,j))/2-p_out)*dy; % Fuente X

% Coef. y fuente para Y
% Por condición de contorno, v_fe=0, se anula este término
Aey(i,j)=0;
% Por condición de contorno al aproximar, se añade el
% término gam/(3*dx)
Awy(i,j)=-((abs(cw)+cw)/2+gam/dx+gam/(3*dx))*dy;
% Son iguales a los términos en x
Any(i,j)=Anx(i,j);
Asy(i,j)=Asx(i,j);
% Por condición de contorno, Aey(i,j) se sustituye por
% 3*gam/dx*dy, más ((abs(cw)-cw)/2+gam/dx
Aoy(i,j)=((abs(cw)-cw)/2+gam/dx+3*gam/dx)*dy +...
      ((abs(cn)+cn)/2+gam/dy + (abs(cs)-cs)/2+gam/dy)*dx ;
Sy(i,j)=1/2*(p_S(i,j)-p_N(i,j))*dx; % Fuente Y

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
j=ny; % Pared superior

i=1; % Zona 1 & 2 | Entrada Esquina superior izquierda

% Coef. y fuente para X
ce=rho_fe(i,j)*u_fe(i,j);
cw=rho_fw(i,j)*u_fw(i,j);
cn=rho_fn(i,j)*v_fn(i,j);
cs=rho_fs(i,j)*v_fs(i,j);
% Por condición de contorno al aproximar, se añade el
% término gam/(3*dx));
Aex(i,j)=-((abs(ce)-ce)/2+gam/dx+gam/(3*dx))*dy;
% Por condición de contorno, u_fw=0 y v_fw=0, se anula
% este término
Awx(i,j)=0;
% Por condición de contorno, u_fn=0 y v_fn=0, se anula
% este término
Anx(i,j)=0;
% Por condición de contorno al aproximar, se añade el
% término gam/(3*dy)
Asx(i,j)=-((abs(cs)+cs)/2+gam/dy+gam/(3*dy))*dx;
% Por condición de contorno, Awx(i,j) se sustituye por
% 3*gam/dx*dy y Anx(i,j) se sustituye por 3*gam/dy*dx
Aox(i,j)=((abs(ce)+ce)/2+gam/dx+3*gam/dx)*dy+...
      ((abs(cs)-cs)/2+gam/dy+3*gam/dy)*dx;
% Por condición de contorno p_fw se aproxima en
% 2º Orden por 3/2*p_o-1/2*p_E
Sx(i,j)=(p(i,j)-p_E(i,j))*dy; % Fuente X

```

```

% Coef. y fuente para Y
% Para este caso concreto, los coef. X e Y son iguales.
Aoy(i,j)=Aox(i,j);
Aey(i,j)=Aex(i,j);
Awy(i,j)=Awx(i,j);
Any(i,j)=Anx(i,j);
Asy(i,j)=Asx(i,j);
% Por condición de contorno p_fn se aproxima en
% 2º Orden por 3/2*p_o-1/2*p_S
Sy(i,j)=(p_S(i,j)-p(i,j))*dx; % Fuente Y

for i=2:nx-1 % Zona 2 | Pared superior columnas interiores

% Coef. y fuente para X
ce=rho_fe(i,j)*u_fe(i,j);
cw=rho_fw(i,j)*u_fw(i,j);
cn=rho_fn(i,j)*v_fn(i,j);
cs=rho_fs(i,j)*v_fs(i,j);
Aex(i,j)=-((abs(ce)-ce)/2+gam/dx)*dy;
Awx(i,j)=-((abs(cw)+cw)/2+gam/dx)*dy;
Anx(i,j)=0;
% Por condición de contorno al aproximar, se añade el
% término gam/(3*dy)
Asx(i,j)=-((abs(cs)+cs)/2+gam/dy+gam/(3*dy))*dx;
% Por condición de contorno, Anx(i,j) se sustituye por
% 3*gam/dy*dx y se suma (abs(cs)-cs)/2+gam/dy
Aox(i,j)=( (abs(ce)+ce)/2+gam/dx +...
( abs(cw)-cw)/2+gam/dx)*dy +(3*gam/dy+(abs(cs)-cs)/2+gam/dy)*dx;
Sx(i,j)=1/2*(p_W(i,j)-p_E(i,j))*dy; % Fuente X

% Coef. y fuente para Y
% Para este caso concreto, los coef. X e Y son iguales.
Aoy(i,j)=Aox(i,j);
Aey(i,j)=Aex(i,j);
Awy(i,j)=Awx(i,j);
Any(i,j)=Anx(i,j);
Asy(i,j)=Asx(i,j);
% Por condición de contorno p_fn se aproxima en
% 2º Orden por 3/2*p_o-1/2*p_S
Sy(i,j)=(p_S(i,j)-p(i,j))*dx; % Fuente Y
end

i=nx; % Zona 3 & 2 | Salida Esquina superior derecha

% Coef. y fuente para X
ce=rho_fe(i,j)*u_fe(i,j);
cw=rho_fw(i,j)*u_fw(i,j);
cn=rho_fn(i,j)*v_fn(i,j);
cs=rho_fs(i,j)*v_fs(i,j);

```

```

% Por condición de contorno, se anula este término
Aex(i,j)=0;
Awx(i,j)=-((abs(cw)+cw)/2+gam/dx)*dy;
% El valor de este término se incluye en la fuente
Anx(i,j)=0;
% Por condición de contorno al aproximar, se añade el
% término gam/(3*dy)
Asx(i,j)=-((abs(cs)+cs)/2+gam/dy+gam/(3*dy))*dx;
% Por condición de contorno, Aex(i,j) se sustituye por
% 3*gam/dx*dy y Anx(i,j) se sustituye por 3*gam/dy*dx
Aox(i,j)=rho_o(i,j)*u(i,j)*dy + ((abs(cw)-cw)/2+gam/dx)*dy ...
        +((abs(cs)-cs)/2+gam/dy+3*gam/dy)*dx;
% Por condición de contorno p_fe = p_out presión de
% la salida
Sx(i,j)=((p(i,j)+p_W(i,j))/2-p_out)*dy;    % Fuente X

% Coef. y fuente para Y
Aey(i,j)=0;
% Por condición de contorno al aproximar, se añade el
% término gam/(3*dx)
Awy(i,j)=-((abs(cw)+cw)/2+gam/dx+gam/(3*dx))*dy;
Any(i,j)=0;
Asy(i,j)=Asx(i,j);
% Por condición de contorno, Aey(i,j) se sustituye por
% 3*gam/dx*dy, más ((abs(cw)-cw)/2+gam/dx
Aoy(i,j)=((abs(cw)-cw)/2+gam/dx+3*gam/dx)*dy +...
        ((abs(cs)-cs)/2+gam/dy+3*gam/dy)*dx ;
% Por condición de contorno p_fn se aproxima en
% 2º Orden por 3/2*p_o-1/2*p_S
Sy(i,j)=(p_S(i,j)-p(i,j))*dx;    % Fuente Y

```

end

C.3 Función para resolución de ecuación de cantidad de movimiento en X

```

%=====
%% Resolución Ec. Cant. Mov. en X
%=====
%
% Función para el cálculo de las velocidades horizontales del campo fluido
% u_som y su residuo
%
%=====
%% Problema de canal longitudinal de sección constante

function [u, Res_u_ultimo]=sol_ecm_x(u, p, Aox, Awx, Aex, Anx, Asx, Sx, ...
    dx, dy, iter_CdM, omega_inercial_uv, gam, u_entrada, rho_o)

%=====
% Se reorganizan las presiones para facilitar la programación de las
% expresiones

%% Presiones

[nx, ny]=size(p); % Se saca las dimensiones de la matriz;

p_W=zeros(nx,ny); p_W(2:nx,1:ny)=p(1:nx-1,1:ny);
p_E=zeros(nx,ny); p_E(1:nx-1,1:ny)=p(2:nx,1:ny);

%% Velocidades
% Se reorganizan las velocidades para facilitar la programación de las
% expresiones.
%
% IMPORTANTE: en este caso, solo es útil para el cálculo del residuo. No
% se puede seguir usando esta separación de las incógnitas en la resolución
% de las Ec. de Cant. Movimiento, porque la velocidad u(i,j) se va
% actualizando constantemente y eso implicaría actualizar tras cada
% iteración 5 variables u_som, u_W_som, u_E_som, u_N_som, u_S_som. Por lo
% tanto lo más óptimo es solo actualizar u(i,j) y usar las equivalencias
% entre W, E, N, S, con los índices i, j.

u_W_som=zeros(nx,ny); u_W_som(2:nx,1:ny)=u(1:nx-1,1:ny);
u_E_som=zeros(nx,ny); u_E_som(1:nx-1,1:ny)=u(2:nx,1:ny);
u_S_som=zeros(nx,ny); u_S_som(1:nx,2:ny)=u(1:nx,1:ny-1);
u_N_som=zeros(nx,ny); u_N_som(1:nx,1:ny-1)=u(1:nx,2:ny);

%=====
%% Cálculo del Residuo
% Se calcula en la primera iteración, recorriendo todas las celdas
% interiores. NO se tienen en cuenta las celdas de los bordes donde hay
% condiciones especiales.

```

```

res=0;
for i=2:nx-1          % Celdas interiores
    for j=2:ny-1
        res=res+(Aox(i,j)*u(i,j)+Awx(i,j)*u_W_som(i,j)+Aex(i,j)*u_E_som(i,j)+...
                Asx(i,j)*u_S_som(i,j)+Anx(i,j)*u_N_som(i,j)...
                - 1/2*(p_W(i,j)-p_E(i,j))*dy)^2;
    end
end
end

res=sqrt(res);

Res_u_ultimo=res;

%=====
%% Resolución del sistema de ecuaciones

    for k=1:iter_CdM %Iteramos estos barridos en filas y columnas

%=====
%% Barrido por filas

j=1; % Primera fila / Pared inferior

    i=1; % Zona 4 & 1 | Esquina inferior izquierda

        b(i)=(1+omega_inercial_uv)*Aox(i,j);
        c(i)=Aex(i,j);
        % Hay que añadir el término
        % (rho_o(i,j)*u_entrada*u_entrada+8*gam/(3*dx))*dy, en la entrada,
        % por condicion de contorno Axw. Hay que quitar este término porque
        % es cero. Asx(i,j)*u(i,j-1)
        d(i)=Sx(i,j)-Anx(i,j)*u(i,j+1)+(rho_o(i,j)*u_entrada*u_entrada+...
            8*gam*u_entrada/(3*dx))*dy;
        % Se incluye estos términos para hacer el cálculo de la correccion.
        d(i)=d(i)-Aox(i,j)*u(i,j)-Aex(i,j)*u(i+1,j);

for i=2:nx-1 % Zona 4 | Columnas interiores / Pared inferior

    a(i-1)=Awx(i,j);
    b(i)=(1+omega_inercial_uv)*Aox(i,j);
    c(i)=Aex(i,j);
    % Hay que quitar este término porque es cero. Asx(i,j)*u(i,j-1)
    d(i)=Sx(i,j)-Anx(i,j)*u(i,j+1);
    % Se incluye estos términos para hacer el cálculo de la correccion.
    d(i)=d(i)-Aox(i,j)*u(i,j)-Aex(i,j)*u(i+1,j)-Awx(i,j)*u(i-1,j);
end

```

```

i=nx; % Zona 4 & 3 | Salida
    a(i-1)=Awx(i,j);
    b(i)=(1+omega_inercial_uv)*Aox(i,j);
    d(i)=Sx(i,j)-Anx(i,j)*u(i,j+1);
    % Se incluye estos términos para hacer el cálculo de la correccion.
    d(i)=d(i)-Aox(i,j)*u(i,j)-Awx(i,j)*u(i-1,j);

% Resolución con TDMA
x_sol=tdma(a,b,c,d);

% Se actualiza los valores de la velocidad de esa fila
for i=1:nx
    u(i,j)=u(i,j)+x_sol(i);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for j=2:ny-1 % Filas entre las paredes inferiores y superior

    i=1; % Zona 1 | Entrada

        b(i)=(1+omega_inercial_uv)*Aox(i,j);
        c(i)=Aex(i,j);
        % Hay que añadir el término
        % (rho_o(i,j)*u_entrada*u_entrada+8*gam/(3*dx))*dy,
        % en la entrada, por condicion de contorno Awx.
        d(i)=Sx(i,j)-Anx(i,j)*u(i,j+1)-Asx(i,j)*u(i,j-1)+...
            (rho_o(i,j)*u_entrada*u_entrada+8*gam*u_entrada/(3*dx))*dy;
        % Se incluye estos términos para hacer el cálculo de la correccion.
        d(i)=d(i)-Aox(i,j)*u(i,j)-Aex(i,j)*u(i+1,j);

    for i=2:nx-1 % Zona interior | Celdas interiores

        a(i-1)=Awx(i,j);
        b(i)=(1+omega_inercial_uv)*Aox(i,j);
        c(i)=Aex(i,j);
        d(i)=Sx(i,j)-Anx(i,j)*u(i,j+1)-Asx(i,j)*u(i,j-1);
        % Se incluye estos términos para hacer el cálculo de la correccion.
        d(i)=d(i)-Aox(i,j)*u(i,j)-Aex(i,j)*u(i+1,j)-Awx(i,j)*u(i-1,j);

    end

    i=nx; % Zona 3 | Salida

        a(i-1)=Awx(i,j);
        b(i)=(1+omega_inercial_uv)*Aox(i,j);
        d(i)=Sx(i,j)-Anx(i,j)*u(i,j+1)-Asx(i,j)*u(i,j-1);
        % Se incluye estos términos para hacer el cálculo de la correccion.
        d(i)=d(i)-Aox(i,j)*u(i,j)-Awx(i,j)*u(i-1,j);

```

```

% Resolución con TDMA
x_sol=tdma(a,b,c,d);

% Se actualiza los valores de la velocidad de esa fila
for i=1:nx
    u(i,j)=u(i,j)+x_sol(i);
end

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
j=ny; % Última fila / Pared superior

i=1; % Zona 2 & 1 | Esquina superior izquierda

b(i)=(1+omega_inercial_uv)*Aox(i,j);
c(i)=Aex(i,j);
% Hay que añadir el término
% (rho_o(i,j)*u_entrada*u_entrada+8*gam/(3*dx))*dy,
% en la entrada, por condicion de contorno Axw. Hay que quitar
% el término Anx(i,j)*u(i,j+1) porque no hay más valores de ny
d(i)=Sx(i,j)-Asx(i,j)*u(i,j-1)+...
    (rho_o(i,j)*u_entrada*u_entrada+8*gam*u_entrada/(3*dx))*dy;
% Se incluye estos términos para hacer el cálculo de la correccion.
d(i)=d(i)-Aox(i,j)*u(i,j)-Aex(i,j)*u(i+1,j);

for i=2:nx-1 % Zona 2 | Columnas interiores / Pared superior

a(i-1)=Awx(i,j);
b(i)=(1+omega_inercial_uv)*Aox(i,j);
c(i)=Aex(i,j);
% Hay que quitar el término Anx(i,j)*u(i,j+1) porque no hay más
% valores de ny
d(i)=Sx(i,j)-Asx(i,j)*u(i,j-1);
% Se incluye estos términos para hacer el cálculo de la correccion.
d(i)=d(i)-Aox(i,j)*u(i,j)-Aex(i,j)*u(i+1,j)-Awx(i,j)*u(i-1,j);

end

i=nx; % Zona 2 & 3 | Salida Esquina superior derecha

a(i-1)=Awx(i,j);
b(i)=(1+omega_inercial_uv)*Aox(i,j);
% Hay que quitar el término Anx(i,j)*u(i,j+1) porque no hay más
% valores de ny
d(i)=Sx(i,j)-Asx(i,j)*u(i,j-1);
% Se incluye estos términos para hacer el cálculo de la correccion.
d(i)=d(i)-Aox(i,j)*u(i,j)-Awx(i,j)*u(i-1,j);

```

```

% Resolución con TDMA
x_sol=tdma(a,b,c,d);

% Se actualiza los valores de la velocidad de esa fila
for i=1:nx
    u(i,j)=u(i,j)+x_sol(i);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Barrido por columnas

% Reseteamos las variables
clear a b c d x_sol

i=1; % Primera columna / Entrada

    j=1; % Zona 1 & 4 | Esquina inferior izquierda entrada

        b(j)=(1+omega_inercial_uv)*Aox(i,j);
        c(j)=Anx(i,j);
        % Hay que añadir el término
        % (rho_o(i,j)*u_entrada*u_entrada+8*gam/(3*dx))*dy,
        % en la entrada, por condicion de contorno Axw.
        % Hay que quitar el término Awx(i,j)*u(i-1,j) porque
        % no hay terminos menores de i=1
        d(j)=Sx(i,j)-Aex(i,j)*u(i+1,j)+...
            (rho_o(i,j)*u_entrada*u_entrada+8*gam*u_entrada/(3*dx))*dy;
        % Se incluye estos términos para hacer el cálculo
        % de la correccion.
        d(j)=d(j)-Aox(i,j)*u(i,j)-Anx(i,j)*u(i,j+1);

    for j=2:ny-1 % Zona 1 | Entrada

        a(j-1)=Asx(i,j);
        b(j)=(1+omega_inercial_uv)*Aox(i,j);
        c(j)=Anx(i,j);
        % Hay que añadir el término
        % (rho_o(i,j)*u_entrada*u_entrada+8*gam/(3*dx))*dy,
        % en la entrada, por condicion de contorno Axw.
        % Hay que quitar el término Awx(i,j)*u(i-1,j) porque
        % no hay terminos menores de i=1
        d(j)=Sx(i,j)-Aex(i,j)*u(i+1,j)+...
            (rho_o(i,j)*u_entrada*u_entrada+8*gam*u_entrada/(3*dx))*dy;
        % Se incluye estos términos para hacer el cálculo
        % de la correccion.
        d(j)=d(j)-Aox(i,j)*u(i,j)-Anx(i,j)*u(i,j+1)-Asx(i,j)*u(i,j-1);
    end
end

```

```

j=ny; % Zona 1 & 2 | Esquina superior izquierda Entrada

a(j-1)=Asx(i,j);
b(j)=(1+omega_inercial_uv)*Aox(i,j);
% Hay que añadir el término
% (rho_o(i,j)*u_entrada*u_entrada+8*gam/(3*dx))*dy,
% en la entrada, por condicion de contorno Axw.
% Hay que quitar el término Awx(i,j)*u(i-1,j) porque
% no hay terminos menores de i=1
d(j)=Sx(i,j)-Aex(i,j)*u(i+1,j)+...
    (rho_o(i,j)*u_entrada*u_entrada+8*gam*u_entrada/(3*dx))*dy;
% Se incluye estos términos para hacer el cálculo
% de la correccion.
% Hay que quitar el término Anx(i,j)*u(i,j+1),
% no hay valores mayores a j=ny
d(j)=d(j)-Aox(i,j)*u(i,j)-Asx(i,j)*u(i,j-1);

% Resolución con TDMA
x_sol=tdma(a,b,c,d);

% Se actualiza los valores de la velocidad de esa columna
for j=1:ny
    u(i,j)=u(i,j)+x_sol(j);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=2:nx-1 % Columnas interiores entre la entrada y la salida

j=1; % Zona 4 | Pared inferior

b(j)=(1+omega_inercial_uv)*Aox(i,j);
c(j)=Anx(i,j);
d(j)=Sx(i,j)-Aex(i,j)*u(i+1,j)-Awx(i,j)*u(i-1,j);
d(j)=d(j)-Aox(i,j)*u(i,j)-Anx(i,j)*u(i,j+1);

for j=2:ny-1 % Zona interior | Celdas interiores

a(j-1)=Asx(i,j);
b(j)=(1+omega_inercial_uv)*Aox(i,j);
c(j)=Anx(i,j);
d(j)=Sx(i,j)-Aex(i,j)*u(i+1,j)-Awx(i,j)*u(i-1,j);
% Se incluye estos términos para hacer el cálculo
% de la correccion.
d(j)=d(j)-Aox(i,j)*u(i,j)-Anx(i,j)*u(i,j+1)-Asx(i,j)*u(i,j-1);

end

```

```

j=ny; % Zona 2 | Pared superior

a(j-1)=Asx(i,j);
b(j)=(1+omega_inercial_uv)*Aox(i,j);
d(j)=Sx(i,j)-Aex(i,j)*u(i+1,j)-Awx(i,j)*u(i-1,j);
% Se incluye estos términos para hacer el cálculo
% de la correccion.
% Hay que quitar el término Anx(i,j)*u(i,j+1),
% no hay valores más de j=ny
d(j)=d(j)-Aox(i,j)*u(i,j)-Asx(i,j)*u(i,j-1);

% Resolución con TDMA
x_sol=tdma(a,b,c,d);

% Se actualiza los valores de la velocidad de esa columna
for j=1:ny
    u(i,j)=u(i,j)+x_sol(j);
end

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
i=nx; % Salida Zona 3

j=1; % Zona 4 & 3 | Esquina inferior izquierda

b(j)=(1+omega_inercial_uv)*Aox(i,j);
c(j)=Anx(i,j);
% Hay que quitar el término -Aex(i,j)*u(i+1,j),
% porque no hay valores para i=nx+1
d(j)=Sx(i,j)-Awx(i,j)*u(i-1,j);
% Se incluye estos términos para hacer el cálculo
% de la correccion.
d(j)=d(j)-Aox(i,j)*u(i,j)-Anx(i,j)*u(i,j+1);

for j=2:ny-1 % Zona 3 | Salida

a(j-1)=Asx(i,j);
b(j)=(1+omega_inercial_uv)*Aox(i,j);
c(j)=Anx(i,j);
% Hay que quitar el término -Aex(i,j)*u(i+1,j), porque
% no hay valores para i=nx+1
d(j)=Sx(i,j)-Awx(i,j)*u(i-1,j);
% Se incluye estos términos para hacer el cálculo
% de la correccion.
d(j)=d(j)-Aox(i,j)*u(i,j)-Anx(i,j)*u(i,j+1)-Asx(i,j)*u(i,j-1);

end

```


C.4 Función para resolución de ecuación de cantidad de movimiento en Y

```

%=====
%% Resolución Ec. Cant. Mov. en Y
%=====

% Función para el cálculo de las velocidades verticales del campo fluido
% v_som y su residual

%=====
%% Problema de canal longitudinal de sección constante

function [v, Res_v_ultimo]=sol_ecm_y(v, p, Aoy, Awy, Aey, Any, Asy, Sy,...
    dx, iter_CdM, omega_inercial_uv)

%=====
% Se reorganizan las presiones para facilitar la programación de las
% expresiones

%% Presiones

[nx, ny]=size(p); % Se saca las dimensiones de la matriz;

p_S=zeros(nx,ny); p_S(1:nx,2:ny)=p(1:nx,1:ny-1);
p_N=zeros(nx,ny); p_N(1:nx,1:ny-1)=p(1:nx,2:ny);

%% Velocidades
% Se reorganizan las velocidades para facilitar la programación de las
% expresiones.

% IMPORTANTE: en este caso, solo es útil para el cálculo del residual. No
% se puede seguir usando esta separación de las incógnitas en la resolución
% de las Ec. de Cant. Movimiento, porque la velocidad v(i,j) se va
% actualizando constantemente y eso implicaría actualizar tras cada
% iteración 5 variables v_som, v_W_som, v_E_som, v_N_som, v_S_som. Por lo
% tanto lo más óptimo es solo actualizar v(i,j) y usar las equivalencias
% entre W, E, N, S, con los índices i, j.

v_W_som=zeros(nx,ny); v_W_som(2:nx,1:ny)=v(1:nx-1,1:ny);
v_E_som=zeros(nx,ny); v_E_som(1:nx-1,1:ny)=v(2:nx,1:ny);
v_S_som=zeros(nx,ny); v_S_som(1:nx,2:ny)=v(1:nx,1:ny-1);
v_N_som=zeros(nx,ny); v_N_som(1:nx,1:ny-1)=v(1:nx,2:ny);

%=====
%% Cálculo del Residual
% Se calcula en la primera iteración, recorriendo todas las celdas
% interiores. NO se tienen en cuenta las celdas de los bordes donde hay
% condiciones especiales.

```

```

res=0;
for i=2:nx-1          % Celdas interiores
    for j=2:ny-1
        res=res+(Aoy(i,j)*v(i,j)+Awy(i,j)*v_W_som(i,j)+...
        Aey(i,j)*v_E_som(i,j)+Asy(i,j)*v_S_som(i,j)+...
        Any(i,j)*v_N_som(i,j)- 1/2*(p_S(i,j)-p_N(i,j))*dx)^2;
    end
end

res=sqrt(res);

Res_v_ultimo=res;

%=====
%% Resolución del sistema de ecuaciones

for k=1:iter_CdM %Iteramos estos barridos en filas y columnas

%% Barrido por filas

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
j=1; % Primera fila / Pared inferior

i=1; % Zona 4 & 1 | Esquina inferior izquierda

    b(i)=(1+omega_inercial_uv)*Aoy(i,j);
    c(i)=Aey(i,j);
    % Hay que quitar este término porque es cero. Asy(i,j)*v(i,j-1)
    d(i)=Sy(i,j)-Any(i,j)*v(i,j+1);
    d(i)=d(i)-Aoy(i,j)*v(i,j)-Aey(i,j)*v(i+1,j);

for i=2:nx-1 % Zona 4 | Columnas interiores / Pared inferior

    a(i-1)=Awy(i,j);
    b(i)=(1+omega_inercial_uv)*Aoy(i,j);
    c(i)=Aey(i,j);
    % Hay que quitar este término porque es cero. Asy(i,j)*v(i,j-1)
    d(i)=Sy(i,j)-Any(i,j)*v(i,j+1);
    % Se incluye estos términos para hacer el cálculo de la correccion.
    d(i)=d(i)-Aoy(i,j)*v(i,j)-Aey(i,j)*v(i+1,j)-Awy(i,j)*v(i-1,j);
end

i=nx; % Zona 4 & 3 | Salida

    a(i-1)=Awy(i,j);
    b(i)=(1+omega_inercial_uv)*Aoy(i,j);
    d(i)=Sy(i,j)-Any(i,j)*v(i,j+1);
    % Se incluye estos términos para hacer el cálculo de la correccion.
    d(i)=d(i)-Aoy(i,j)*v(i,j)-Awy(i,j)*v(i-1,j);

```



```

j=ny; % Última fila / Pared superior

i=1; % Zona 2 & 1 | Esquina superior izquierda

b(i)=(1+omega_inercial_uv)*Aoy(i,j);
c(i)=Aey(i,j);
% Hay que quitar el término Any(i,j)*v(i,j+1) porque no hay más
% valores de ny
d(i)=Sy(i,j)-Asy(i,j)*v(i,j-1);
% Se incluye estos términos para hacer el cálculo de la correccion.
d(i)=d(i)-Aoy(i,j)*v(i,j)-Aey(i,j)*v(i+1,j);

for i=2:nx-1 % Zona 2 | Columnas interiores / Pared superior

a(i-1)=Awy(i,j);
b(i)=(1+omega_inercial_uv)*Aoy(i,j);
c(i)=Aey(i,j);
% Hay que quitar el término Any(i,j)*v(i,j+1) porque no hay más
% valores de ny
d(i)=Sy(i,j)-Asy(i,j)*v(i,j-1);
% Se incluye estos términos para hacer el cálculo de la correccion.
d(i)=d(i)-Aoy(i,j)*v(i,j)-Aey(i,j)*v(i+1,j)-Awy(i,j)*v(i-1,j);
end

i=nx; % Zona 2 & 3 | Salida Esquina superior derecha

a(i-1)=Awy(i,j);
b(i)=(1+omega_inercial_uv)*Aoy(i,j);
% Hay que quitar el término Any(i,j)*v(i,j+1) porque no hay más
% valores de ny
d(i)=Sy(i,j)-Asy(i,j)*v(i,j-1);
% Se incluye estos términos para hacer el cálculo de la correccion.
d(i)=d(i)-Aoy(i,j)*v(i,j)-Awy(i,j)*v(i-1,j);

% Resolución con TDMA
x_sol=tdma(a,b,c,d);

% Se actualiza los valores de la velocidad de esa fila
for i=1:nx
v(i,j)=v(i,j)+x_sol(i);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Barrido por columnas

% Reseteamos las variables
clear a b c d x_sol

```



```

for i=2:nx-1 % Columnas interiores entre la entrada y la salida

    j=1; % Zona 4 | Pared inferior

        b(j)=(1+omega_inercial_uv)*Aoy(i,j);
        c(j)=Any(i,j);
        d(j)=Sy(i,j)-Aey(i,j)*v(i+1,j)-Awy(i,j)*v(i-1,j);
        % Se incluye estos términos para hacer el cálculo
        % de la correccion.
        d(j)=d(j)-Aoy(i,j)*v(i,j)-Any(i,j)*v(i,j+1);

    for j=2:ny-1 % Zona interior | Celdas interiores

        a(j-1)=Asy(i,j);
        b(j)=(1+omega_inercial_uv)*Aoy(i,j);
        c(j)=Any(i,j);
        d(j)=Sy(i,j)-Aey(i,j)*v(i+1,j)-Awy(i,j)*v(i-1,j);
        % Se incluye estos términos para hacer el cálculo
        % de la correccion.
        d(j)=d(j)-Aoy(i,j)*v(i,j)-Any(i,j)*v(i,j+1)-Asy(i,j)*v(i,j-1);
    end

    j=ny; % Zona 2 | Pared superior

        a(j-1)=Asy(i,j);
        b(j)=(1+omega_inercial_uv)*Aoy(i,j);
        d(j)=Sy(i,j)-Aey(i,j)*v(i+1,j)-Awy(i,j)*v(i-1,j);
        % Se incluye estos términos para hacer el cálculo
        % de la correccion. Hay que quitar el término
        % Anx(i,j)*u(i,j+1), no hay valores más de j=ny
        d(j)=d(j)-Aoy(i,j)*v(i,j)-Asy(i,j)*v(i,j-1);

    % Resolución con TDMA
    x_sol=tdma(a,b,c,d);

    % Se actualiza los valores de la velocidad de esa columna
    for j=1:ny
        v(i,j)=v(i,j)+x_sol(j);
    end

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
i=nx; % Salida Zona 3

    j=1; % Zona 4 & 3 | Esquina inferior izquierda

        b(j)=(1+omega_inercial_uv)*Aoy(i,j);
        c(j)=Any(i,j);

```


C.5 Función PWIM de interpolación de velocidades en las caras

```

%=====
%% Interpolación de la velocidades en las caras con PWIM
%=====

% Una vez resuelta las Ec. Cant. Mov. y conocido el campo de velocidades en
% cada uno de los centros de las celdas, se proecede con PWIM a interpolar
% el valor de las velocidades en las caras.

%=====
%% Problema de canal longitudinal de sección constante

function [u_fw_som,u_fe_som,v_fs_som,v_fn_som]=Interp_PWIM(u_som,v_som, p,...
    Aox, Aoy, dx, dy, u_entrada, v_sup, v_inf, p_out)

%=====
%% Presiones
% Se reorganizan las presiones para facilitar la programación de las
% expresiones

[nx, ny]=size(p); % Se saca las dimensiones de la matriz;

p_W=zeros(nx,ny); p_W(2:nx,1:ny)=p(1:nx-1,1:ny);
p_E=zeros(nx,ny); p_E(1:nx-1,1:ny)=p(2:nx,1:ny);
p_S=zeros(nx,ny); p_S(1:nx,2:ny)=p(1:nx,1:ny-1);
p_N=zeros(nx,ny); p_N(1:nx,1:ny-1)=p(1:nx,2:ny);
p_EE=zeros(nx,ny); p_EE(1:nx-2,1:ny)=p(3:nx,1:ny);
p_NN=zeros(nx,ny); p_NN(1:nx,1:ny-2)=p(1:nx,3:ny);

%% Velocidades
% Se reorganizan las velocidades para facilitar la programación de las
% expresiones
u_W_som=zeros(nx,ny); u_W_som(2:nx,1:ny)=u_som(1:nx-1,1:ny);
u_E_som=zeros(nx,ny); u_E_som(1:nx-1,1:ny)=u_som(2:nx,1:ny);
v_S_som=zeros(nx,ny); v_S_som(1:nx,2:ny)=v_som(1:nx,1:ny-1);
v_N_som=zeros(nx,ny); v_N_som(1:nx,1:ny-1)=v_som(1:nx,2:ny);

%% Coeficientes de enlace
% Se reorganizan los coeficientes para facilitar la programación de las
% expresiones

Aox_W=zeros(nx,ny); Aox_W(2:nx,1:ny)=Aox(1:nx-1,1:ny);
Aox_E=zeros(nx,ny); Aox_E(1:nx-1,1:ny)=Aox(2:nx,1:ny);
Aoy_S=zeros(nx,ny); Aoy_S(1:nx,2:ny)=Aoy(1:nx,1:ny-1);
Aoy_N=zeros(nx,ny); Aoy_N(1:nx,1:ny-1)=Aoy(1:nx,2:ny);

%=====

```

```

%% Interpolación de velocidades en las caras

% [CONDICIONES DE CONTORNO]
% Dependen de las condiciones de contorno establecidas hay que distinguir
% la posición en la que se encuentra la celda.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
j=1; % Primera fila / Pared inferior

    i=1; % Zona 4 & 1 | Entrada Esquina inferior izquierda

    % Se descomponen los elementos para hacer más sencillo
    Uoe=(u_som(i,j)+u_E_som(i,j))/2;
    % Aplicando la aproximación de 2ºOrden para p_w
    Pew=(p_E(i,j)-p(i,j))/(2*Aox(i,j))*dy;
    Peeo=1/2*(p_EE(i,j)-p(i,j))/(2*Aox_E(i,j))*dy;
    Peo=-(1/Aox_E(i,j)+1/Aox(i,j))*(p_E(i,j)-p(i,j))/2*dy;
    u_fe_som(i,j)=Uoe+Pew+Peeo+Peo; % Vel. cara Este

    Von=(v_som(i,j)+v_N_som(i,j))/2;
    % Aplicando la aproximación de 2ºOrden para p_s
    Pns=(p_N(i,j)-p(i,j))/(2*Aoy(i,j))*dx;
    Pnno=1/2*(p_NN(i,j)-p(i,j))/(2*Aoy_N(i,j))*dx;
    Pno=-(1/Aoy_N(i,j)+1/Aoy(i,j))*(p_N(i,j)-p(i,j))/2*dx;
    v_fn_som(i,j)=Von+Pns+Pnno+Pno; % Vel. cara Norte

    % Velocidad de la cara izquierda definida por condicion
    % de contorno Velocidad de entrada
    u_fw_som(i,j)=u_entrada; % Vel. cara Oeste
    % Velocidad de la cara inferior definida por condicion
    % de contorno
    v_fs_som(i,j)=v_inf; % Vel. cara Sur

for i=2:nx-2 % Zona 4 | Columnas interiores / Pared inferior

    % Se descomponen los elementos para hacer más sencillo
    Uoe=(u_som(i,j)+u_E_som(i,j))/2;
    Pew=1/2*(p_E(i,j)-p_W(i,j))/(2*Aox(i,j))*dy;
    Peeo=1/2*(p_EE(i,j)-p(i,j))/(2*Aox_E(i,j))*dy;
    Peo=-(1/Aox_E(i,j)+1/Aox(i,j))*(p_E(i,j)-p(i,j))/2*dy;
    u_fe_som(i,j)=Uoe+Pew+Peeo+Peo; % Vel. cara Este

    Von=(v_som(i,j)+v_N_som(i,j))/2;
    % Aplicando la aproximación de 2ºOrden para p_s
    Pns=(p_N(i,j)-p(i,j))/(2*Aoy(i,j))*dx;
    Pnno=1/2*(p_NN(i,j)-p(i,j))/(2*Aoy_N(i,j))*dx;
    Pno=-(1/Aoy_N(i,j)+1/Aoy(i,j))*(p_N(i,j)-p(i,j))/2*dx;
    v_fn_som(i,j)=Von+Pns+Pnno+Pno; % Vel. cara Norte

```

```

% Tiene que ser la misma que la cara Este de la celda
% de la izquierda
u_fw_som(i,j)=u_fe_som(i-1,j); % Vel. cara Oeste
% Velocidad de la cara inferior definida por condicion
% de contorno
v_fs_som(i,j)=v_inf; % Vel. cara Sur

end

i=nx-1; % Zona 4 & 6 | Cerca de la salida p_EE

% Se descomponen los elementos para hacer más sencillo
Uoe=(u_som(i,j)+u_E_som(i,j))/2;
Pew=1/2*(p_E(i,j)-p_W(i,j))/(2*Aox(i,j))*dy;
% Aplicando la presión p_EE = p_out
Peeo=(p_out-(p(i,j)+p_E(i,j))/2)/(2*Aox_E(i,j))*dy;
Peo=-(1/Aox_E(i,j)+1/Aox(i,j))*(p_E(i,j)-p(i,j))/2*dy;
u_fe_som(i,j)=Uoe+Pew+Peeo+Peo; % Vel. cara Este

Von=(v_som(i,j)+v_N_som(i,j))/2;
% Aplicando la aproximación de 2ºOrden para p_s
Pns=(p_N(i,j)-p(i,j))/(2*Aoy(i,j))*dx;
Pnno=1/2*(p_NN(i,j)-p(i,j))/(2*Aoy_N(i,j))*dx;
Pno=-(1/Aoy_N(i,j)+1/Aoy(i,j))*(p_N(i,j)-p(i,j))/2*dx;
v_fn_som(i,j)=Von+Pns+Pnno+Pno; % Vel. cara Norte

% Tiene que ser la misma que la cara Este de la celda
% de la izquierda
u_fw_som(i,j)=u_fe_som(i-1,j); % Vel. cara Oeste
% Velocidad de la cara inferior definida por condicion
% de contorno
v_fs_som(i,j)=v_inf; % Vel. cara Sur

i=nx; % Zona 4 & 3 | Salida

% Se descomponen los elementos para hacer más sencillo
% Por condición de contorno, por coherencia, u_fe_som = u_som
u_fe_som(i,j)=u_som(i,j); % Vel. cara Este

Von=(v_som(i,j)+v_N_som(i,j))/2;
% Aplicando la aproximación de 2ºOrden para p_s
Pns=(p_N(i,j)-p(i,j))/(2*Aoy(i,j))*dx;
Pnno=1/2*(p_NN(i,j)-p(i,j))/(2*Aoy_N(i,j))*dx;
Pno=-(1/Aoy_N(i,j)+1/Aoy(i,j))*(p_N(i,j)-p(i,j))/2*dx;
v_fn_som(i,j)=Von+Pns+Pnno+Pno; % Vel. cara Norte

% Tiene que ser la misma que la cara Este de la celda
% de la izquierda
u_fw_som(i,j)=u_fe_som(i-1,j); % Vel. cara Oeste

```

```

% Velocidad de la cara inferior definida por
% condicion de contorno
v_fs_som(i,j)=v_inf; % Vel. cara Sur

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for j=2:ny-2 % Filas interiores

    i=1; % Zona 1 | Entrada

    % Se descomponen los elementos para hacer más sencillo
    Uoe=(u_som(i,j)+u_E_som(i,j))/2;
    % Aplicando la aproximación de 2º Orden para p_w
    Pew=(p_E(i,j)-p(i,j))/(2*Aox(i,j))*dy;
    Peeo=1/2*(p_EE(i,j)-p(i,j))/(2*Aox_E(i,j))*dy;
    Peo=-(1/Aox_E(i,j)+1/Aox(i,j))*(p_E(i,j)-p(i,j))/2*dy;
    u_fe_som(i,j)=Uoe+Pew+Peeo+Peo; % Vel. cara Este

    Von=(v_som(i,j)+v_N_som(i,j))/2;
    Pns=1/2*(p_N(i,j)-p_S(i,j))/(2*Aoy(i,j))*dx;
    Pnno=1/2*(p_NN(i,j)-p(i,j))/(2*Aoy_N(i,j))*dx;
    Pno=-(1/Aoy_N(i,j)+1/Aoy(i,j))*(p_N(i,j)-p(i,j))/2*dx;
    v_fn_som(i,j)=Von+Pns+Pnno+Pno; % Vel. cara Norte

    % Velocidad de la cara izquierda definida por condicion de
    % contorno Velocidad de entrada
    u_fw_som(i,j)=u_entrada; % Vel. cara Oeste
    % Tiene que ser la misma que la cara Norte de la celda de abajo
    v_fs_som(i,j)=v_fn_som(i,j-1); % Vel. cara Sur

for i=2:nx-2 % Celdas interiores

    % Se descomponen los elementos para hacer más sencillo
    Uoe=(u_som(i,j)+u_E_som(i,j))/2;
    Pew=1/2*(p_E(i,j)-p_W(i,j))/(2*Aox(i,j))*dy;
    Peeo=1/2*(p_EE(i,j)-p(i,j))/(2*Aox_E(i,j))*dy;
    Peo=-(1/Aox_E(i,j)+1/Aox(i,j))*(p_E(i,j)-p(i,j))/2*dy;
    u_fe_som(i,j)=Uoe+Pew+Peeo+Peo; % Vel. cara Este

    Von=(v_som(i,j)+v_N_som(i,j))/2;
    Pns=1/2*(p_N(i,j)-p_S(i,j))/(2*Aoy(i,j))*dx;
    Pnno=1/2*(p_NN(i,j)-p(i,j))/(2*Aoy_N(i,j))*dx;
    Pno=-(1/Aoy_N(i,j)+1/Aoy(i,j))*(p_N(i,j)-p(i,j))/2*dx;
    v_fn_som(i,j)=Von+Pns+Pnno+Pno; % Vel. cara Norte

    % Tiene que ser la misma que la cara Este de la celda
    % de la izquierda
    u_fw_som(i,j)=u_fe_som(i-1,j); % Vel. cara Oeste
    % Tiene que ser la misma que la cara Norte de la celda de abajo
    v_fs_som(i,j)=v_fn_som(i,j-1); % Vel. cara Sur

end

```

```

i=nx-1; % Zona 6 | Cerca de la salida   p_EE

% Se descomponen los elementos para hacer más sencillo
Uoe=(u_som(i,j)+u_E_som(i,j))/2;
Pew=1/2*(p_E(i,j)-p_W(i,j))/(2*Aox(i,j))*dy;
% Aplicando la presión p_EE = p_out
Peeo=(p_out-(p(i,j)+p_E(i,j))/2)/(2*Aox_E(i,j))*dy;
Peo=- (1/Aox_E(i,j)+1/Aox(i,j))*(p_E(i,j)-p(i,j))/2*dy;
u_fe_som(i,j)=Uoe+Pew+Peeo+Peo; % Vel. cara Este

Von=(v_som(i,j)+v_N_som(i,j))/2;
Pns=1/2*(p_N(i,j)-p_S(i,j))/(2*Aoy(i,j))*dx;
Pnno=1/2*(p_NN(i,j)-p(i,j))/(2*Aoy_N(i,j))*dx;
Pno=- (1/Aoy_N(i,j)+1/Aoy(i,j))*(p_N(i,j)-p(i,j))/2*dx;
v_fn_som(i,j)=Von+Pns+Pnno+Pno; % Vel. cara Norte

% Tiene que ser la misma que la cara Este de la celda
% de la izquierda
u_fw_som(i,j)=u_fe_som(i-1,j); % Vel. cara Oeste
% Tiene que ser la misma que la cara Norte de la celda de abajo
v_fs_som(i,j)=v_fn_som(i,j-1); % Vel. cara Sur

i=nx; % Zona 3 | Salida

% Se descomponen los elementos para hacer más sencillo
% Por condición de contorno, por coherencia, u_fe_som = u_som
u_fe_som(i,j)=u_som(i,j); % Vel. cara Este

Von=(v_som(i,j)+v_N_som(i,j))/2;
Pns=1/2*(p_N(i,j)-p_S(i,j))/(2*Aoy(i,j))*dx;
Pnno=1/2*(p_NN(i,j)-p(i,j))/(2*Aoy_N(i,j))*dx;
Pno=- (1/Aoy_N(i,j)+1/Aoy(i,j))*(p_N(i,j)-p(i,j))/2*dx;
v_fn_som(i,j)=Von+Pns+Pnno+Pno; % Vel. cara Norte

% Tiene que ser la misma que la cara Este de la celda
% de la izquierda
u_fw_som(i,j)=u_fe_som(i-1,j); % Vel. cara Oeste
% Tiene que ser la misma que la cara Norte de la celda de abajo
v_fs_som(i,j)=v_fn_som(i,j-1); % Vel. cara Sur

end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
j=ny-1; % Fila cerca de la pared superior p_NN

i=1; % Zona 1 | Entrada

% Se descomponen los elementos para hacer más sencillo
Uoe=(u_som(i,j)+u_E_som(i,j))/2;

```

```

% Aplicando la aproximación de 2ºOrden para p_w
Pew=(p_E(i,j)-p(i,j))/(2*Aox(i,j))*dy;
Peeo=1/2*(p_EE(i,j)-p(i,j))/(2*Aox_E(i,j))*dy;
Peo=-(1/Aox_E(i,j)+1/Aox(i,j))*(p_E(i,j)-p(i,j))/2*dy;
u_fe_som(i,j)=Uoe+Pew+Peeo+Peo; % Vel. cara Este

Von=(v_som(i,j)+v_N_som(i,j))/2;
Pns=1/2*(p_N(i,j)-p_S(i,j))/(2*Aoy(i,j))*dx;
% Al estar cerca de la pared, p_NN se tiene que aproximar por
% Taylor (aproximación de 2ºOrden)
Pnno=dx/(2*Aoy_N(i,j))*(p_N(i,j)-p(i,j));
Pno=-(1/Aoy_N(i,j)+1/Aoy(i,j))*(p_N(i,j)-p(i,j))/2*dx;
v_fn_som(i,j)=Von+Pns+Pnno+Pno; % Vel. cara Norte
% Velocidad de la cara izquierda definida por condicion
% de contorno Velocidad de entrada
u_fw_som(i,j)=u_entrada; % Vel. cara Oeste
% Tiene que ser la misma que la cara Norte de la celda de abajo
v_fs_som(i,j)=v_fn_som(i,j-1); % Vel. cara Sur

for i=2:nx-2 % Zona 5 & Celdas interiores

% Se descomponen los elementos para hacer más sencillo
Uoe=(u_som(i,j)+u_E_som(i,j))/2;
Pew=1/2*(p_E(i,j)-p_W(i,j))/(2*Aox(i,j))*dy;
Peeo=1/2*(p_EE(i,j)-p(i,j))/(2*Aox_E(i,j))*dy;
Peo=-(1/Aox_E(i,j)+1/Aox(i,j))*(p_E(i,j)-p(i,j))/2*dy;
u_fe_som(i,j)=Uoe+Pew+Peeo+Peo; % Vel. cara Este

Von=(v_som(i,j)+v_N_som(i,j))/2;
Pns=1/2*(p_N(i,j)-p_S(i,j))/(2*Aoy(i,j))*dx;
% Al estar cerca de la pared, p_NN se tiene que aproximar
% por Taylor (aproximación de 2ºOrden)
Pnno=dx/(2*Aoy_N(i,j))*(p_N(i,j)-p(i,j));
Pno=-(1/Aoy_N(i,j)+1/Aoy(i,j))*(p_N(i,j)-p(i,j))/2*dx;
v_fn_som(i,j)=Von+Pns+Pnno+Pno; % Vel. cara Norte

% Tiene que ser la misma que la cara Este de la celda
% de la izquierda
u_fw_som(i,j)=u_fe_som(i-1,j); % Vel. cara Oeste
% Tiene que ser la misma que la cara Norte de la celda de abajo
v_fs_som(i,j)=v_fn_som(i,j-1); % Vel. cara Sur

end

i=nx-1; % Zona 6 & 5 | Cerca de la salida p_EE

% Se descomponen los elementos para hacer más sencillo
Uoe=(u_som(i,j)+u_E_som(i,j))/2;
Pew=1/2*(p_E(i,j)-p_W(i,j))/(2*Aox(i,j))*dy;
% Aplicando la presión p_EE = p_out

```

```

Peeo=(p_out-(p(i,j)+p_E(i,j))/2)/(2*Aox_E(i,j))*dy;
Peo=- (1/Aox_E(i,j)+1/Aox(i,j))*(p_E(i,j)-p(i,j))/2*dy;
u_fe_som(i,j)=Uoe+Pew+Peeo+Peo; % Vel. cara Este

Von=(v_som(i,j)+v_N_som(i,j))/2;
Pns=1/2*(p_N(i,j)-p_S(i,j))/(2*Aoy(i,j))*dx;
% Al estar cerca de la pared, p_NN se tiene que aproximar
% por Taylor (aproximación de 2ºOrden)
Pnno=dx/(2*Aoy_N(i,j))*(p_N(i,j)-p(i,j));
Pno=- (1/Aoy_N(i,j)+1/Aoy(i,j))*(p_N(i,j)-p(i,j))/2*dx;
v_fn_som(i,j)=Von+Pns+Pnno+Pno; % Vel. cara Norte

% Tiene que ser la misma que la cara Este de la celda
% de la izquierda
u_fw_som(i,j)=u_fe_som(i-1,j); % Vel. cara Oeste
% Tiene que ser la misma que la cara Norte de la celda de abajo
v_fs_som(i,j)=v_fn_som(i,j-1); % Vel. cara Sur

i=nx; % Zona 3 & 5 | Salida

% Se descomponen los elementos para hacer más sencillo
% Por condición de contorno, por coherencia, u_fe_som = u_som
u_fe_som(i,j)=u_som(i,j); % Vel. cara Este

Von=(v_som(i,j)+v_N_som(i,j))/2;
Pns=1/2*(p_N(i,j)-p_S(i,j))/(2*Aoy(i,j))*dx;
% Al estar cerca de la pared, p_NN se tiene que aproximar
% por Taylor (aproximación de 2ºOrden)
Pnno=dx/(2*Aoy_N(i,j))*(p_N(i,j)-p(i,j));
Pno=- (1/Aoy_N(i,j)+1/Aoy(i,j))*(p_N(i,j)-p(i,j))/2*dx;
v_fn_som(i,j)=Von+Pns+Pnno+Pno; % Vel. cara Norte

% Tiene que ser la misma que la cara Este de la celda
% de la izquierda
u_fw_som(i,j)=u_fe_som(i-1,j); % Vel. cara Oeste
% Tiene que ser la misma que la cara Norte de la celda de abajo
v_fs_som(i,j)=v_fn_som(i,j-1); % Vel. cara Sur

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

j=ny; % Zona 2 Pared superior

i=1; % Zona 2 & 1 | Entrada

Uoe=(u_som(i,j)+u_E_som(i,j))/2;
% Aplicando la aproximación de 2ºOrden para p_w
Pew=(p_E(i,j)-p(i,j))/(2*Aox(i,j))*dy;
Peeo=1/2*(p_EE(i,j)-p(i,j))/(2*Aox_E(i,j))*dy;
Peo=- (1/Aox_E(i,j)+1/Aox(i,j))*(p_E(i,j)-p(i,j))/2*dy;
u_fe_som(i,j)=Uoe+Pew+Peeo+Peo; % Vel. cara Este

```

```

% Velocidad de la cara superior definida por
% condicion de contorno
v_fn_som(i,j)=v_sup;          % Vel. cara Norte

% Velocidad de la cara izquierda definida por condicion
% de contorno Velocidad de entrada
u_fw_som(i,j)=u_entrada;     % Vel. cara Oeste
% Tiene que ser la misma que la cara Norte de la
% celda de abajo
v_fs_som(i,j)=v_fn_som(i,j-1); % Vel. cara Sur

for i=2:nx-2 % Zona 2 Pared superior

% Se descomponen los elementos para hacer más sencillo
Uoe=(u_som(i,j)+u_E_som(i,j))/2;
Pew=1/2*(p_E(i,j)-p_W(i,j))/(2*Aox(i,j))*dy;
Peeo=1/2*(p_EE(i,j)-p(i,j))/(2*Aox_E(i,j))*dy;
Peo=-(1/Aox_E(i,j)+1/Aox(i,j))*(p_E(i,j)-p(i,j))/2*dy;
u_fe_som(i,j)=Uoe+Pew+Peeo+Peo; % Vel. cara Este

% Velocidad de la cara superior definida por
% condicion de contorno
v_fn_som(i,j)=v_sup;          % Vel. cara Norte

% Tiene que ser la misma que la cara Este de la celda
% de la izquierda
u_fw_som(i,j)=u_fe_som(i-1,j); % Vel. cara Oeste
% Tiene que ser la misma que la cara Norte de la
% celda de abajo
v_fs_som(i,j)=v_fn_som(i,j-1); % Vel. cara Sur

end

i=nx-1; % Zona 2 & 6 | Cerca de la salida p_EE

% Se descomponen los elementos para hacer más sencillo
Uoe=(u_som(i,j)+u_E_som(i,j))/2;
Pew=1/2*(p_E(i,j)-p_W(i,j))/(2*Aox(i,j))*dy;
% Aplicando la presión p_EE = p_out
Peeo=(p_out-(p(i,j)+p_E(i,j))/2)/(2*Aox_E(i,j))*dy;
Peo=-(1/Aox_E(i,j)+1/Aox(i,j))*(p_E(i,j)-p(i,j))/2*dy;
u_fe_som(i,j)=Uoe+Pew+Peeo+Peo; % Vel. cara Este

% Velocidad de la cara superior definida por
% condicion de contorno
v_fn_som(i,j)=v_sup;          % Vel. cara Norte

```



```

j=1; % Primera fila / Pared inferior

    i=1; % Zona 4 & 1 | Esquina inferior izquierda

        Aep(i,j)=-rho_fe(i,j)/2*dy^2*(1/Aox_E(i,j)+1/Aox(i,j));
        Awp(i,j)=0;
        Anp(i,j)=-rho_fn(i,j)/2*dx^2*(1/Aoy_N(i,j)+1/Aoy(i,j));
        Asp(i,j)=0;
        Aop(i,j)=-1*(Aep(i,j)+Awp(i,j)+Anp(i,j)+Asp(i,j));
        Sp(i,j)=- (dy*(rho_fe(i,j)*u_fe_som(i,j)-rho_o(i,j)*u_entrada)...
            +dx*(rho_fn(i,j)*v_fn_som(i,j)));

    for i=2:nx-1 % Zona 4 | Columnas interiores / Pared inferior

        Aep(i,j)=-rho_fe(i,j)/2*dy^2*(1/Aox_E(i,j)+1/Aox(i,j));
        Awp(i,j)=-rho_fw(i,j)/2*dy^2*(1/Aox_W(i,j)+1/Aox(i,j));
        Anp(i,j)=-rho_fn(i,j)/2*dx^2*(1/Aoy_N(i,j)+1/Aoy(i,j));
        Asp(i,j)=0;
        Aop(i,j)=-1*(Aep(i,j)+Awp(i,j)+Anp(i,j)+Asp(i,j));
        Sp(i,j)=- (dy*(rho_fe(i,j)*u_fe_som(i,j)-rho_fw(i,j)*u_fw_som(i,j))...
            +dx*(rho_fn(i,j)*v_fn_som(i,j)));

    end

    i=nx; % Zona 4 & 3 | Salida

        Aep(i,j)=0;
        % Por condicion de contorno hay que agregar el
        % término rho_o(i,j)/(2*Aox(i,j))*dy^2
        Awp(i,j)=-rho_fw(i,j)/2*dy^2*(1/Aox_W(i,j)+1/Aox(i,j))+...
            rho_o(i,j)/(2*Aox(i,j))*dy^2;
        Anp(i,j)=-rho_fn(i,j)/2*dx^2*(1/Aoy_N(i,j)+1/Aoy(i,j));
        Asp(i,j)=0;
        % Por condicion de contorno hay que agregar el
        % término rho_o(i,j)/(2*Aox(i,j))*dy^2
        Aop(i,j)= rho_fw(i,j)/2*dy^2*(1/Aox_W(i,j)+1/Aox(i,j)) +...
            rho_fn(i,j)/2*dx^2*(1/Aoy_N(i,j)+1/Aoy(i,j)) +...
            rho_o(i,j)/(2*Aox(i,j))*dy^2;
        Sp(i,j)=- (dy*(rho_o(i,j)*u_som(i,j)-rho_fw(i,j)*u_fw_som(i,j))+...
            dx*(rho_fn(i,j)*v_fn_som(i,j)) );

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for j=2:ny-1 % Filas entre las paredes inferiores y superior

    i=1; % Zona 1 | Entrada

        Aep(i,j)=-rho_fe(i,j)/2*dy^2*(1/Aox_E(i,j)+1/Aox(i,j));
        Awp(i,j)=0;
        Anp(i,j)=-rho_fn(i,j)/2*dx^2*(1/Aoy_N(i,j)+1/Aoy(i,j));
        Asp(i,j)=-rho_fs(i,j)/2*dx^2*(1/Aoy_S(i,j)+1/Aoy(i,j));
        Aop(i,j)=-1*(Aep(i,j)+Awp(i,j)+Anp(i,j)+Asp(i,j));

```

```

    Sp(i,j)=-(dy*(rho_fe(i,j)*u_fe_som(i,j)-rho_o(i,j)*u_entrada)+...
            dx*(rho_fn(i,j)*v_fn_som(i,j)-rho_fs(i,j)*v_fs_som(i,j)));

for i=2:nx-1 % Zona interior | Celdas interiores

    Aep(i,j)=-rho_fe(i,j)/2*dy^2*(1/Aox_E(i,j)+1/Aox(i,j));
    Awp(i,j)=-rho_fw(i,j)/2*dy^2*(1/Aox_W(i,j)+1/Aox(i,j));
    Anp(i,j)=-rho_fn(i,j)/2*dx^2*(1/Aoy_N(i,j)+1/Aoy(i,j));
    Asp(i,j)=-rho_fs(i,j)/2*dx^2*(1/Aoy_S(i,j)+1/Aoy(i,j));
    Aop(i,j)=-1*(Aep(i,j)+Awp(i,j)+Anp(i,j)+Asp(i,j));
    Sp(i,j)=-(dy*(rho_fe(i,j)*u_fe_som(i,j)-rho_fw(i,j)*u_fw_som(i,j))...
            +dx*(rho_fn(i,j)*v_fn_som(i,j)-rho_fs(i,j)*v_fs_som(i,j)));

end

i=nx; % Zona 3 | Salida

    Aep(i,j)=0;
    % Por condicion de contorno hay que agregar el
    % término rho_o(i,j)/(2*Aox(i,j))*dy^2
    Awp(i,j)=-rho_fw(i,j)/2*dy^2*(1/Aox_W(i,j)+1/Aox(i,j))+...
            rho_o(i,j)/(2*Aox(i,j))*dy^2;
    Anp(i,j)=-rho_fn(i,j)/2*dx^2*(1/Aoy_N(i,j)+1/Aoy(i,j));
    Asp(i,j)=-rho_fs(i,j)/2*dx^2*(1/Aoy_S(i,j)+1/Aoy(i,j));
    % Por condicion de contorno hay que agregar el
    % término rho_o(i,j)/(2*Aox(i,j))*dy^2
    Aop(i,j)= rho_fw(i,j)/2*dy^2*(1/Aox_W(i,j)+1/Aox(i,j)) + ...
            rho_fn(i,j)/2*dx^2*(1/Aoy_N(i,j)+1/Aoy(i,j)) + ...
            rho_fs(i,j)/2*dx^2*(1/Aoy_S(i,j)+1/Aoy(i,j)) + ...
            rho_o(i,j)/(2*Aox(i,j))*dy^2;
    Sp(i,j)=-(dy*(rho_o(i,j)*u_som(i,j)-rho_fw(i,j)*u_fw_som(i,j))+...
            dx*(rho_fn(i,j)*v_fn_som(i,j)-rho_fs(i,j)*v_fs_som(i,j)));

end

j=ny; % Última fila / Pared superior

i=1; % Zona 2 & 1 | Esquina superior izquierda

    Aep(i,j)=-rho_fe(i,j)/2*dy^2*(1/Aox_E(i,j)+1/Aox(i,j));
    Awp(i,j)=0;
    Anp(i,j)=0;
    Asp(i,j)=-rho_fs(i,j)/2*dx^2*(1/Aoy_S(i,j)+1/Aoy(i,j));
    Aop(i,j)=-1*(Aep(i,j)+Awp(i,j)+Anp(i,j)+Asp(i,j));
    Sp(i,j)=-(dy*(rho_fe(i,j)*u_fe_som(i,j)-rho_o(i,j)*u_entrada)+...
            dx*(-rho_fs(i,j)*v_fs_som(i,j)));

for i=2:nx-1 % Zona 2 | Columnas interiores / Pared superior

    Aep(i,j)=-rho_fe(i,j)/2*dy^2*(1/Aox_E(i,j)+1/Aox(i,j));
    Awp(i,j)=-rho_fw(i,j)/2*dy^2*(1/Aox_W(i,j)+1/Aox(i,j));

```

```

Anp(i,j)=0;
Asp(i,j)=-rho_fs(i,j)/2*dx^2*(1/Aoy_S(i,j)+1/Aoy(i,j));
Aop(i,j)=-1*(Aep(i,j)+Awp(i,j)+Anp(i,j)+Asp(i,j));
Sp(i,j)=- (dy*(rho_fe(i,j)*u_fe_som(i,j)-rho_fw(i,j)*u_fw_som(i,j))...
          +dx*(-rho_fs(i,j)*v_fs_som(i,j)));
end

i=nx; % Zona 2 & 3 | Salida Esquina superior derecha

Aep(i,j)=0;
% Por condicion de contorno hay que agregar el
% término rho_o(i,j)/(2*Aox(i,j))*dy^2
Awp(i,j)=-rho_fw(i,j)/2*dy^2*(1/Aox_W(i,j)+1/Aox(i,j))+...
          rho_o(i,j)/(2*Aox(i,j))*dy^2;
Anp(i,j)=0;
Asp(i,j)=-rho_fs(i,j)/2*dx^2*(1/Aoy_S(i,j)+1/Aoy(i,j));
% Por condicion de contorno hay que agregar el
% término rho_o(i,j)/(2*Aox(i,j))*dy^2
Aop(i,j)= rho_fw(i,j)/2*dy^2*(1/Aox_W(i,j)+1/Aox(i,j)) +...
          rho_fs(i,j)/2*dx^2*(1/Aoy_S(i,j)+1/Aoy(i,j)) +...
          rho_o(i,j)/(2*Aox(i,j))*dy^2;
Sp(i,j)=- (dy*(rho_o(i,j)*u_som(i,j)-rho_fw(i,j)*u_fw_som(i,j))+...
          dx*(-rho_fs(i,j)*v_fs_som(i,j)));
end

```

C.7 Función para resolución de ecuación de continuidad

```

%=====
%% Cálculo de coeficientes de enlace y fuentes para Corrección de presión
%=====

% Resolución de la ecuación de continuidad
% Función para la resolución de los sistemas de ecuaciones para
% el cálculo de la corrección de presión.

%=====
%% Problema de canal longitudinal de sección constante

function [p_cc, Res_p_ultimo]=sol_cpres(Aop, Awp, Aep, Anp, Asp, Sp, iter_P)

%=====
[nx, ny]=size(Aop); % Se saca las dimensiones de la matriz;

p_cc=zeros(nx,ny); % Se inicializa en zeros porque lo que se van a calcular
                    % son correcciones de presiones, no se está iterando con
                    % la presión total p, está se actualiza después de
                    % calcular las correcciones p_cc.

%=====
%% Cálculo del Residuo
% Se calcula en la primera iteración, recorriendo todas las celdas
% interiores. NO se tienen en cuenta las celdas de los bordes donde hay
% condiciones especiales.

% Coincide con el "desbalanceo" de masa representado por Sp

res=0;
    for i=2:nx-1          % Celdas interiores
        for j=2:ny-1
            res=res+(Sp(i,j))^2;
        end
    end

res=sqrt(res);

Res_p_ultimo=res;

%=====
%% Resolución del sistema de ecuaciones

    for k=1:iter_P %Iteramos estos barridos en filas y columnas

% [CONDICIONES DE CONTORNO]
% Dependen de las condiciones de contorno establecidas hay que distinguir
% la posición en la que se encuentra la celda.

```

```

%% Barrido por filas
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
j=1; % Primera fila / Pared inferior

    i=1; % Zona 4 & 1 | Esquina inferior izquierda

        b(i)=Aop(i,j);
        c(i)=Aep(i,j);
        % No hace falta añadir más términos. Ya son correcciones. Hay
        % que quitar el término Asp(i,j)*p_cc(i,j-1), no hay
        % valores para j=0
        d(i)=Sp(i,j)-Anp(i,j)*p_cc(i,j+1);

    for i=2:nx-1 % Zona 4 | Columnas interiores / Pared inferior

        a(i-1)=Awp(i,j);
        b(i)=Aop(i,j);
        c(i)=Aep(i,j);
        % Hay que quitar el término Asp(i,j)*p_cc(i,j-1), no hay
        % valores para j=0
        d(i)=Sp(i,j)-Anp(i,j)*p_cc(i,j+1);
    end

    i=nx; % Zona 4 & 3 | Salida

        a(i-1)=Awp(i,j);
        b(i)=Aop(i,j);
        d(i)=Sp(i,j)-Anp(i,j)*p_cc(i,j+1);

% Resolución con TDMA
x_sol=tdma(a,b,c,d);

% Se actualiza los valores de la velocidad de esa fila
for i=1:nx
    p_cc(i,j)=x_sol(i);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for j=2:ny-1 % Filas entre las paredes inferiores y superior

    i=1; % Zona 1 | Entrada

        b(i)=Aop(i,j);
        c(i)=Aep(i,j);
        d(i)=Sp(i,j)-Anp(i,j)*p_cc(i,j+1)-Asp(i,j)*p_cc(i,j-1);

    for i=2:nx-1 % Zona interior | Celdas interiores

        a(i-1)=Awp(i,j);

```

```

        b(i)=Aop(i,j);
        c(i)=Aep(i,j);
        d(i)=Sp(i,j)-Anp(i,j)*p_cc(i,j+1)-Asp(i,j)*p_cc(i,j-1);
    end

    i=nx; % Zona 3 | Salida

        a(i-1)=Awp(i,j);
        b(i)=Aop(i,j);
        d(i)=Sp(i,j)-Anp(i,j)*p_cc(i,j+1)-Asp(i,j)*p_cc(i,j-1);

% Resolución con TDMA
x_sol=tdma(a,b,c,d);

% Se actualiza los valores de la velocidad de esa fila
for i=1:nx
    p_cc(i,j)=x_sol(i);
end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

j=ny; % Última fila / Pared superior

    i=1; % Zona 2 & 1 | Esquina superior izquierda

        b(i)=Aop(i,j);
        c(i)=Aep(i,j);
        % Hay que quitar los términos Anp(i,j)*p_cc(i,j+1),
        % porque no hay valores para j=ny+1
        d(i)=Sp(i,j)-Asp(i,j)*p_cc(i,j-1);

    for i=2:nx-1 % Zona 2 | Columnas interiores / Pared superior

        a(i-1)=Awp(i,j);
        b(i)=Aop(i,j);
        c(i)=Aep(i,j);
        % Hay que quitar los términos Anp(i,j)*p_cc(i,j+1),
        % porque no hay valores para j=ny+1
        d(i)=Sp(i,j)-Asp(i,j)*p_cc(i,j-1);
    end

    i=nx; % Zona 2 & 3 | Salida Esquina superior derecha

        a(i-1)=Awp(i,j);
        b(i)=Aop(i,j);
        % Hay que quitar los términos Anp(i,j)*p_cc(i,j+1),
        % porque no hay valores para j=ny+1
        d(i)=Sp(i,j)-Asp(i,j)*p_cc(i,j-1);

```

```

% Resolución con TDMA
x_sol=tdma(a,b,c,d);

% Se actualiza los valores de la velocidad de esa fila
for i=1:nx
    p_cc(i,j)=x_sol(i);
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Barrido por columnas

% Reseteamos las variables
clear a b c d x_sol

i=1; % Primera columna / Entrada

    j=1; % Zona 1 & 4 | Esquina inferior izquierda entrada

        b(j)=Aop(i,j);
        c(j)=Anp(i,j);
        % Hay que quitar los términos Awp(i,j)*p_cc(i-1,j),
        % no hay valores para i=0
        d(j)=Sp(i,j)-Aep(i,j)*p_cc(i+1,j);

    for j=2:ny-1 % Zona 1 | Entrada

        a(j-1)=Asp(i,j);
        b(j)=Aop(i,j);
        c(j)=Anp(i,j);
        % Hay que quitar los términos Awp(i,j)*p_cc(i-1,j),
        % no hay valores para i=0
        d(j)=Sp(i,j)-Aep(i,j)*p_cc(i+1,j);
    end

    j=ny; % Zona 1 & 2 | Esquina superior izquierda Entrada

        a(j-1)=Asp(i,j);
        b(j)=Aop(i,j);
        % Hay que quitar los términos Awp(i,j)*p_cc(i-1,j),
        % no hay valores para i=0
        d(j)=Sp(i,j)-Aep(i,j)*p_cc(i+1,j);

% Resolución con TDMA
x_sol=tdma(a,b,c,d);

% Se actualiza los valores de la velocidad de esa columna
for j=1:ny
    p_cc(i,j)=x_sol(j);
end

```

```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=2:nx-1 % Columnas interiores entre la entrada y la salida

    j=1; % Zona 4 | Pared inferior

        b(j)=Aop(i,j);
        c(j)=Anp(i,j);
        d(j)=Sp(i,j)-Aep(i,j)*p_cc(i+1,j)-Awp(i,j)*p_cc(i-1,j);

    for j=2:ny-1 % Zona interior | Celdas interiores

        a(j-1)=Asp(i,j);
        b(j)=Aop(i,j);
        c(j)=Anp(i,j);
        d(j)=Sp(i,j)-Aep(i,j)*p_cc(i+1,j)-Awp(i,j)*p_cc(i-1,j);
    end

    j=ny; % Zona 2 | Pared superior

        a(j-1)=Asp(i,j);
        b(j)=Aop(i,j);
        d(j)=Sp(i,j)-Aep(i,j)*p_cc(i+1,j)-Awp(i,j)*p_cc(i-1,j);

    % Resolución con TDMA
    x_sol=tdma(a,b,c,d);

    % Se actualiza los valores de la velocidad de esa columna
    for j=1:ny
        p_cc(i,j)=x_sol(j);
    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
i=nx; % Salida Zona 3

    j=1; % Zona 4 & 3 | Esquina inferior izquierda

        b(j)=Aop(i,j);
        c(j)=Anp(i,j);
        % Hay que quitar el término Aep(i,j)*p_cc(i+1,j)
        % porque no hay valores para i=nx+1
        d(j)=Sp(i,j)-Awp(i,j)*p_cc(i-1,j);

    for j=2:ny-1 % Zona 3 | Salida

        a(j-1)=Asp(i,j);
        b(j)=Aop(i,j);
        c(j)=Anp(i,j);
        % Hay que quitar el término Aep(i,j)*p_cc(i+1,j) porque
        % no hay valores para i=nx+1

```


C.8 Función de actualización de velocidades y presiones

```

%=====
%% Actualizar campo de presiones
%=====

% Una vez se consiguen las correcciones de las presiones p', con el
% amortiguamiento definido, se procede a actualizar el campo de presiones p

%=====
function [p]=actualizar_p(p, p_cc, omega_p)

p=p+omega_p.*p_cc;

end
%=====

%=====
%% Actualización del campo de velocidades
%=====

% Tras calcular las correcciones de presiones, se procede a actualizar el
% campo de velocidades en el centro de cada celda y en las caras de cada
% celda.

%=====
%% Problema de canal longitudinal de sección constante

function [u,v,u_fw,u_fe,v_fs,v_fn]=actualizar_vel(u_som, v_som, u_fe_som, ...
    v_fn_som, p_cc, Aox, Aoy, dx, dy, omega_uv,...
    u_entrada, v_sup, v_inf)

%=====
%% Presiones corregidas
% Se reorganizan las presiones para facilitar la programación

[nx, ny]=size(p_cc); % Se saca las dimensiones de la matriz;

p_cc_W=zeros(nx,ny); p_cc_W(2:nx,1:ny)=p_cc(1:nx-1,1:ny);
p_cc_E=zeros(nx,ny); p_cc_E(1:nx-1,1:ny)=p_cc(2:nx,1:ny);
p_cc_S=zeros(nx,ny); p_cc_S(1:nx,2:ny)=p_cc(1:nx,1:ny-1);
p_cc_N=zeros(nx,ny); p_cc_N(1:nx,1:ny-1)=p_cc(1:nx,2:ny);

```

```

%% Coeficientes de enlace Aox y Aoy
% Se reorganizan para facilitar la programación

Aox_E=zeros(nx,ny); Aox_E(1:nx-1,1:ny)=Aox(2:nx,1:ny);
Aoy_N=zeros(nx,ny); Aoy_N(1:nx,1:ny-1)=Aoy(1:nx,2:ny);

%=====
%% Actualización de Velocidades en el centro de las celdas

% Operaciones con cada uno de los componentes de las matrices. No es
% necesario hacer ningún bucle.

u=u_som+omega_uv.*dy.*(p_cc_W-p_cc_E)./(2.*Aox);
v=v_som+omega_uv.*dx.*(p_cc_S-p_cc_N)./(2.*Aoy);

%=====
%% Actualización de Velocidades en las caras de las celdas

% [CONDICIONES DE CONTORNO]
% Dependen de las condiciones de contorno establecidas hay que distinguir
% la posición en la que se encuentra la celda.

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

j=1; % Primera fila / Pared inferior

    i=1; % Zona 4 & 1 | Entrada Esquina inferior izquierda

        u_fe(i,j)=u_fe_som(i,j)+omega_uv*(1/Aox_E(i,j)+...
            1/Aox(i,j))*(p_cc(i,j)-p_cc_E(i,j))/2*dy; % Vel. cara Este
        v_fn(i,j)=v_fn_som(i,j)+omega_uv*(1/Aoy_N(i,j)+...
            1/Aoy(i,j))*(p_cc(i,j)-p_cc_N(i,j))/2*dx; % Vel. cara Norte
        % Por condición de contorno, es igual a la velocidad de Entrada
        u_fw(i,j)=u_entrada; % Vel. cara Oeste
        % Velocidad de la cara inferior definida por condicion de contorno
        v_fs(i,j)=v_inf; % Vel. cara Sur

    for i=2:nx-1 % Zona 4 | Columnas interiores / Pared inferior

        u_fe(i,j)=u_fe_som(i,j)+omega_uv*(1/Aox_E(i,j)...
            +1/Aox(i,j))*(p_cc(i,j)-p_cc_E(i,j))/2*dy; % Vel. cara Este
        v_fn(i,j)=v_fn_som(i,j)+omega_uv*(1/Aoy_N(i,j)+...
            1/Aoy(i,j))*(p_cc(i,j)-p_cc_N(i,j))/2*dx; % Vel. cara Norte
        % Tiene que ser la misma que la cara Este de la celda de la izquierda
        u_fw(i,j)=u_fe(i-1,j); % Vel. cara Oeste
        % Velocidad de la cara inferior definida por condicion de contorno
        v_fs(i,j)=v_inf; % Vel. cara Sur
    end

    i=nx; % Zona 4 & 3 | Salida Esquina inferior derecha

```


C.9 Función de post-procesado

```

%=====
%% Post-procesado
%=====
%
% Este script toma los datos obtenidos de la resolución del dominio fluido y
% los configura para mostrar las distintas gráficas mostradas en el
% documento:
%   - Evolución de los residuos
%   - Campo de velocidad horizontal y vectores de velocidad
%   - Campo de presiones y líneas de corriente
%   - Evolución del perfil de velocidad horizontal a lo largo del canal
%
%=====
%% Problema de canal longitudinal de sección constante

% El problema que se pretende resolver se trata de un conducto longitudinal
% de sección constante. El flujo entra con una velocidad horizontal u_in y
% sale a una presión conocida p_out

%=====
clear all, clc
close all
%=====
%% Carga de resultados

Re=100; % Valores calculados 100 200 300 400 450

load (['Resultados_Re_', num2str(Re), '.mat'])

%% Parámetros del problema

% Dimensiones del dominio fluido
H=0.01;      % [m] Dimensión característica
Long_x=5*H;  % [m] Longitud total del conducto
Long_y=H;    % [m] Altura total del conducto

%% Propiedades del fluido
% Se van a considerar constantes.

rho=1;       % Densidad [kg/m3]
gam=1e-5;    % Viscosidad [kg/(m s)]

%=====
%% Mallado estructurado y co-ubicado
% Ortogonal y con todas las celdas iguales

% Se saca las dimensiones de la matriz que coincide con el número de celdas
% en x e y

```

```

[nx, ny]=size(u);

dx=Long_x/nx;    % Cálculo de la longitud en dirección x de la celda [m]
dy=Long_y/ny;    % Cálculo de la longitud en dirección y de la celda [m]
ar=dx/dy;        % Relación de aspecto
ra=1/ar;         % Inverso de la relación de aspecto

% Centro de las celdas
centro_x=dx/2:dx:(Long_x-dx/2);    % En dirección x [m]
centro_y=dy/2:dy:(Long_y-dy/2);    % En dirección y [m]

% Parámetros del mallado para la representación gráfica
[X,Y] = meshgrid(centro_x, centro_y);
U=u;
V=v;

%=====
%% Representación de los residuos

% Se sacan los residuales de la matriz donde se almacenaron
iter=Residuales(2:end,1); % 1º columna
R_X=Residuales(2:end,2); % 2º columna
R_Y=Residuales(2:end,3); % 3º columna
R_P=Residuales(2:end,4); % 4º columna

figure;
semilogy(iter, R_X, 'b', 'LineWidth', 1.5); % Residuo EcX en rojo
hold on;
semilogy(iter, R_Y, 'r', 'LineWidth', 1.5); % Residuo EcY en verde
semilogy(iter, R_P, 'g', 'LineWidth', 1.5); % Residuo EcP en azul
xlabel('Número de iteraciones');
ylabel('Residuos (log)');
title(['Residuos para Re = ' num2str(Re)])
legend('Residuo u','Residuo v','Residuo p');
grid on; % Activar la cuadrícula en el gráfico

hold off

%=====
%% Representación campo velocidad horizontal u y vector de velocidades

% Reducimos el número de vectores de velocidad que se muestra
% Se reduce U, V a una matriz más pequeña para que se aprecien los valores
% de los vectores
N_X=zeros(1, nx/2+1);
N_Y=zeros(1, ny);
N_U=zeros(nx/2+1, ny);
N_V=zeros(nx/2+1, ny);

i_r=1;

```

```

for i=1:4:nx
    N_X(i_r)=centro_x(i);
    j_r=1;
    for j=1:2:ny
        N_X(i_r)=centro_x(i);
        N_Y(j_r)=centro_y(j);
        N_U(i_r,j_r)=U(i,j);
        N_V(i_r,j_r)=V(i,j);

        j_r=j_r+1;
    end
    i_r=i_r+1;
end

% Para coger los valores de todas las filas de la última columna
N_X(nx/2+1)=centro_x(nx);
N_Y(ny)=centro_y(ny);

N_U(nx/2+1,:)=U(nx,[1:ny]);
N_V(nx/2+1,:)=V(nx,[1:ny]);

N_U(:,ny)=U([1:2:nx nx], ny);
N_V(:,ny)=V([1:2:nx nx], ny);

figure
% Representación de la velocidad horizontal
contourf(centro_x, centro_y, u', 30);
cbar = colorbar; % Barra de color para representar la velocidad
ylabel(cbar, ' Vel. Horizontal [m/s] ');

hold on

% Representación de los vectores de velocidad
quiver(N_X,N_Y,N_U',N_V',1.5,'r')

% Establecer límites en los ejes x e y
xlim([centro_x(1) centro_x(end)]);
ylim([centro_y(1) centro_y(end)]);
xlabel('X [m]');
ylabel('Y [m]');
title(['Velocidad horizontal u para Re = ' num2str(Re)])
legend('Velocidad horizontal', 'Location', 'south',...
    'Orientation', 'horizontal');

%=====
%% Representación del campo de presiones y líneas de corriente

[X, Y] = meshgrid(centro_x, centro_y);
% Parámetros para representación de las líneas de corriente
[sx, sy] = meshgrid(X(1,1), Y(1:2:20,1));

```

```

figure
contourf(X, Y, p, 'k');
colormap(winter);          % Comando para hacerlo segun patrón de colores
cbar = colorbar;          % Barra de color para representar la presión
ylabel(cbar, ' \Delta Presión [Pa] ');

hold on

h=streamline(X,Y,U',V',sx,sy); % Representación de líneas de corriente
set(h,'Color','red');

xlabel('X [m]');
ylabel('Y [m]');
title(['Campo de presiones y Líneas de corriente para Re = ' num2str(Re)])
legend('Presión manométrica', 'Líneas de corriente',...
       'Location', 'northeast');

%=====
%% Desarrollo del perfil de velocidades a lo largo del conducto

% Cálculo del centro de las celdas
vector_altura=[0 centro_y Long_y]; % En dirección y [m]

Uin=Re*gam/(rho*Long_y);
Upared=0;          % Condición de contorno en las paredes

x=[0 11 nx/4 nx/2 75 nx]; % Posición longitudinal dentro del canal

leng={'Entrada','0.00875 m ','0.0125 m ','0.025 m ','0.0375 m ','Salida'};

figure
for i=1:length(x)
    if i==1
        U_perfil(1:ny+2)=Uin;
    else
        U_perfil=[Upared, u(x(i), :), Upared];
    end
end
hold on
plot(U_perfil,vector_altura,'LineWidth',2)
end

legend(leng);
% Añadir etiquetas y título
xlabel('Velocidad horizontal [m/s]')
ylabel('Altura [m]')
title(['Perfil de velocidad a lo largo del conducto para Re = ' num2str(Re)])
legend('Location', 'west');
grid on
box on

```

C.10 Función auxiliar: TDMA

```
=====  
%% Resolución de sistema tridiagonal (TDMA)  
=====  
  
% Al no encontrar una función para el método tridiagonal en matlab, se  
% adapta el método, a la creación de una matriz tridiagonal y a la  
% resolución mediante el comando x=A\b.  
  
=====  
function [x]=tdma(a,b,c,d)  
  
% a = diagonal inferior           | vector de N-1  
% b = diagonal principal         | vector de N  
% c = diagonal superior         | vector de N-1  
% d = valores de términos independientes | vector de N  
  
%% Se crea la matriz tridiagonal A con los vectores a, b, y c.  
  
% Verificar que los tamaños de los vectores son coherentes  
if length(a) ~= length(b) - 1 || length(c) ~= length(b) - 1  
    error('Los tamaños de los vectores a, b y c no son...  
        coherentes para una matriz tridiagonal.');
```

A=diag(b)+diag(a,-1)+diag(c,1);

```
%% Se resuelve el sistema  
  
x=A\d';  
  
end
```

Bibliografía

- [1] *The Lattice Boltzmann Method (LBM) in CFD* | SimWiki | SimScale, <https://www.simscale.com/docs/simwiki/cfd-computational-fluid-dynamics/lattice-boltzmann-method-lbm/>.
- [2] John D Anderson, Alexandra Komrakova, and Course Objectives, *Computational methods for heat and mass transfer*, 2019.
- [3] Antonio Barrero Ripoll; Miguel Pérez-Saborid Sánchez-Pastor., *Fundamentos y Aplicaciones de Mecánica de Fluidos*, McGraw-Hill/Interamericana de España, Madrid, 2005 (spa).
- [4] Ivan Peinado Asensi, *Resolución de ecuaciones de convección-difusión en 2D usando el método de las diferencias finitas compactas*, (2019).
- [5] David J. Charlesworth, *Solution of the Incompressible Navier- Stokes Equations on Unstructured Meshes*, Thesis (2003), no. August.
- [6] Shiyi Chen and Gary D. Doolen, *Lattice boltzmann method for fluid flows*, Annual Review of Fluid Mechanics **30** (1998), 329–364.
- [7] Germán Ferreira, Artur Sucena, Luís L. Ferrás, Fernando T. Pinho, and Alexandre M. Afonso, *Hydrodynamic entrance length for laminar flow in microchannels with rectangular cross section*, Fluids **6** (2021), no. 7, 1–13.
- [8] Joel H. Ferziger, *Computational Methods for Fluid Dynamics*, 4th ed. 20 ed., Springer International Publishing, Cham, 2020 (eng).
- [9] Andrés Gonzáles Mora, *Historia de la dinámica de fluidos computacionales (CFD)*, (2018), 1–5.
- [10] Pijush K. Kundu, *Fluid mechanics*, 4th ed. ed., Academic Press, Amsterdam ;, 2008 (eng).
- [11] José M. López-Herrera Sánchez, *Introducción al CFD, Mecánica de fluidos avanzada*, 2023, pp. 1–11.
- [12] Adrián Lozano, *Apuntes sobre Dinámica de Fluidos Computacional*, (2019), 1–111.
- [13] Sandip Mazumder, *CFD Lectures SIMPLE Algorithm on Staggered and Co-located Mesh*, <https://www.youtube.com/watch?v=JUbHKZRupnE&list=PLVuuXJfoPgT4gJcBAAF7uMwjFKB9aqT>.
- [14] Sandip Mazumder, *Numerical Methods for Partial Differential Equations : finite difference and finite volume methods*, Academic Press, London, 2016 (eng).

- [15] Shakerur Ridwan Mohammad A. Hossain, Md. Taibur Rahman, *Numerical Investigation of Fluid Flow Through A 2D Backward Facing Step Channel*, International Journal of Engineering Research Technology (IJERT) **2** (2013), no. 10, 3700–3708.
- [16] Jayathi Y. Murthy, *Numerical Methods in Heat , Mass and Momentum Transfer*, Lecture (2002), 1–196.
- [17] M Ramakrishna, *Elements of Computational Fluid Dynamics - unformatted*, vol. Vol. 1, 2011.
- [18] Manuel Carreño Ruiz and Miguel Pérez-Saborid Sánchez-Pastor, *Trabajo Fin de Grado Grado en Ingeniería Aeroespacial Aplicaciones del método de MacCormack a diversos problemas fluidomecánicos*, (2016).
- [19] J. Tihon, V. Pěnkavová, J. Havlica, and M. Šimčík, *The transitional backward-facing step flow in a water channel with variable expansion geometry*, Experimental Thermal and Fluid Science **40** (2012), 112–125.