

*Métodos Miméticos en la Simulación
de Flujo de Fluido sobre cuerpos
Poligonales en 2D
utilizando la técnica de Confinamiento
de la Vorticidad*

Javier Eduardo Abache Romero

14 de junio de 2005



UNIVERSIDAD DE CARABOBO
Facultad Experimental de Ciencias y Tecnología
Departamento de Computación

*Métodos Miméticos en la Simulación de Flujo de Fluido sobre
cuerpos Poligonales en 2D
utilizando la técnica de Confinamiento de la Vorticidad*

Autor:

Javier Eduardo Abache Romero

Tutor: Prof. Carlos E. Cadenas R.

Bárbula, 14 de junio de 2005

Resumen

En este trabajo se presentan los resultados preliminares de la simulación de flujo de fluidos en dos dimensiones basadas en métodos conservativos, resolviendo numéricamente las ecuaciones de Navier-Stokes y de compresibilidad artificial en 2D, haciendo uso de la librería *UCSparseLib* [?] aplicando la técnica de Confinamiento de la Vorticidad. Se presenta un esquema numérico implícito para 2D, haciendo uso del método Castillo-Grone 2-2-2 [?]. Además, se crearon herramientas necesarias para la visualización final de resultados haciendo uso de OpenGL. En cuanto a los resultados: después de varias pruebas, se eligió un grupo de seis de estas, las cuales son presentados en este trabajo, dichos casos convergieron en su totalidad a la solución deseada, mostrando en la simulación un comportamiento en concordancia con la realidad y las fuentes bibliográficas consultadas. De todos los métodos que se encuentran en *UCSparseLib* y que fueron utilizados en las pruebas: BiCGstab, BiCG, ConjGrad y GMRES; el último fue el que mostró un mejor desempeño.

Índice general

1. PLANTEAMIENTO DEL PROBLEMA	1
1.1. Planteamiento del Problema	1
1.2. Objetivo	5
1.2.1. Objetivo General	5
1.2.2. Objetivos Específicos	5
1.3. Justificación	6
1.4. Antecedentes	8
2. MARCO TEÓRICO	13
2.1. Conceptos Básicos	14
2.1.1. Mecánica de fluidos	14
2.1.2. Ecuaciones Diferenciales y Condiciones de Frontera	17

2.1.3.	Operadores de Divergencia, Gradiente y Laplaciano	20
2.2.	Métodos de Solución Numérica	20
2.2.1.	Diferencia Finita Simple	22
2.2.2.	Volumen Finito	23
2.2.3.	Elemento Finito	23
2.2.4.	Diferencia Finita Mimética	24
2.3.	Generación de Mallados	26
2.3.1.	Mallados en 2D	26
2.4.	Método Castillo-Grone 2-2-2	28
2.4.1.	Método Implícito de Crank-Nicholson	30
2.4.2.	Condiciones de Frontera en CFD	31
2.5.	Resolución de Sistemas de Ecuaciones Lineales	32
2.5.1.	Uso de la librería UCSparceLib	33
2.6.	Confinamiento de la Vorticidad	38
2.7.	Métodos de Compresibilidad Artificial	40
2.8.	OpenGL como API para la Visualización	41
2.9.	Metodología de Ciclo de Vida del desarrollo de software	43

2.9.1. Análisis	44
2.9.2. Diseño	44
2.9.3. Implementación	44
2.9.4. Pruebas	45
3. DESARROLLO	47
3.1. Solución Numérica de las Ecuaciones de Navier-Stokes y Compresibilidad Artificial en 2D, utilizando la técnica de Castillo-Grone 2-2-2 y Confinamiento de la Vorticidad	48
3.1.1. Aproximación de las Ecuaciones de Navier-Stokes, con el Término de Confinamiento de la Vorticidad y Compresibilidad Artificial, en los nodos internos del mallado	48
3.1.2. Aproximaciones de la Ecuaciones de Navier-Stokes, con el término de Confinamiento de la Vorticidad y Compresibilidad Artificial en el obstáculo y la frontera	57
4. RESULTADOS Y CONCLUSIONES	77
4.1. Resultados Numéricos	77
4.2. Conclusiones y Recomendaciones	86
4.3. Trabajos Futuros	87

Índice de figuras

2.1. Mallado Tensorial	27
2.2. Distribución de los valores de Dv y Gf en el mallado	30
4.1. Prueba 1: Fluido	78
4.2. Prueba 1: Velocidad	79
4.3. Prueba 1: Presión	79
4.4. Prueba 1: Mallado	79
4.5. Prueba 2: Fluido	80
4.6. Prueba 2: Velocidad	80
4.7. Prueba 2: Presión	80
4.8. Prueba 2: Mallado	81
4.9. Prueba 3: Fluido	81

4.10. Prueba 3: Velocidad	81
4.11. Prueba 3: Presión	82
4.12. Prueba 3: Mallado	82
4.13. Prueba 4: Fluido	83
4.14. Prueba 4: Velocidad	83
4.15. Prueba 4: Presión	83
4.16. Prueba 4: Mallado	84
4.17. Prueba 5: Fluido	84
4.18. Prueba 5: Velocidad	84
4.19. Prueba 5: Presión	85
4.20. Prueba 5: Mallado	85
4.21. Prueba 6: Fluido	85
4.22. Prueba 6: Velocidad	86
4.23. Prueba 6: Presión	86
4.24. Prueba 6: Mallado	86

Introducción

El control de flujo de fluidos ha emergido recientemente como una importante área de investigación científica y tecnológica. Esto se debe a la amplia gama de importantes aplicaciones en las que es extremadamente útil tener un conocimiento apropiado de la mecánica de fluidos. En biomecánica el flujo de sangre y fluido cerebral son de particular interés; en meteorología e ingeniería oceánica, para entender el movimiento del aire y las corrientes oceánicas, se requiere del conocimiento de la mecánica de fluidos; los ingenieros químicos deben comprender la mecánica de fluidos para diseñar los diferentes equipos de procesamiento químico; los ingenieros aeronáuticos utilizan su conocimiento de fluidos para incrementar al máximo la fuerza de elevación, reducir al mínimo el retardo de aeronaves y para diseñar motores de reacción; los ingenieros mecánicos diseñan bombas, turbinas, motores de combustión interna, compresores de aire, equipos de aire acondicionado, para el control de contaminación y plantas eléctricas con base en el conocimiento apropiado de la mecánica de fluidos; y los ingenieros civiles también utilizan los resultados obtenidos en el estudio de mecánica de fluidos para comprender el transporte de sedimentos y la erosión en ríos, la contaminación del aire y agua, y así diseñar sistemas de tuberías, plantas de tratamiento de aguas negras, canales de irrigación, sistemas de control de inundaciones, presas y estadios

deportivos cubiertos.

Surge entonces la simulación computacional de la dinámica de fluidos (*CFD*, por sus siglas en inglés) como una herramienta para la evaluación e investigación de situaciones como estas sin necesidad de realizarlas, lo que ha llevado a muchos investigadores dedicarse a resolver los problemas de orden numérico para cumplir con estos objetivos, logrando así reducir costos en tiempo, dinero y esfuerzo.

Desde hace algunas décadas, se ha trabajado en el desarrollo de teorías y métodos numéricos que son de gran importancia en la resolución de ecuaciones diferenciales de primer y segundo orden, los cuales permiten obtener aproximaciones cada vez más exactas a fin de modelar algún proceso dentro de un entorno controlado. Nuevas técnicas numéricas como los Métodos de Diferencias Finitas Miméticas, son evaluadas con el fin de brindar a los investigadores la facilidad de resolver problemas en menor tiempo y reducir los costos de muchas pruebas en comparación al desarrollo manual. En estas técnicas de recientemente desarrollo, los operadores discretos mantienen algunas propiedades fundamentales de los operadores continuos de divergencia, gradiente y laplaciano, cómo la simetría y la ley de conservación. Aún están siendo probadas a diferentes niveles desde el punto de vista conceptual y es por ello que esta investigación estaría colocando precedentes tanto en el área numérica, como en el campo de aplicación al cual está siendo enfocada.

El deseo de simular procesos reales, entre ellos los de una refinería de crudo pesado (petróleo), hace muy atractivo este tipo de investigación, ya que la visualización del comportamiento de los fluidos dentro de la refinería es un gran paso en el diseño de proyectos de esa índole; también por el uso de múltiples métodos para

resolver las ecuaciones diferenciales que se involucran en este tipo de problemas.

En el primer capítulo se da a conocer el problema a resolver, los objetivos que se persiguen, la justificación y los antecedentes en las áreas involucradas. En el capítulo dos se introducen algunos conceptos básicos de la dinámica de fluidos computacional y las ecuaciones diferenciales implicadas. Luego se exponen algunas de las formulaciones más importantes para resolver ecuaciones diferenciales. El análisis que se hace de los métodos es en términos de los procedimientos para obtener una solución aproximada, por lo que está orientado al cómo se construye la formulación y no en el por qué la formulación funciona, ya que ello escapa de los alcances del presente trabajo. Es por ello que los conceptos matemáticos se mantendrán lo más simples posible. En el capítulo tres se hace el desarrollo numérico de las ecuaciones planteadas en el capítulo uno, en términos de las formulaciones miméticas. A continuación, se presenta la implementación de las funciones necesarias, explicando las estructuras de datos creadas y los prototipos de las principales funciones. En el capítulo cuatro se muestran los resultados de las pruebas realizadas, así como también las conclusiones, recomendaciones y posibles trabajos futuros. En el apéndice A está el código fuente de las funciones implementadas.

Como un aspecto de notación, todas las variables escalares y vectores o matrices, involucrados se expresan en letras itálicas (ie: x , u , y) y letra negrilla (ie: \mathbf{u} , \mathbf{f} , \mathbf{A}) respectivamente, mientras que para aquellas variables o instrucciones involucradas en la implementación computacional se escribirán con letra tipo (i.e: `ii`, `TDMatrix`)

Capítulo 1

PLANTEAMIENTO DEL PROBLEMA

1.1. Planteamiento del Problema

El modelado de fenómenos naturales es uno de los mayores problemas en la animación por computadoras. La dificultad proviene de la complejidad de los modelos usados para describir el movimiento de dicho fenómeno, que en la mayoría de los casos involucran necesariamente la solución numérica de Ecuaciones Diferenciales (ED) Ordinarias o Parciales, que pueden ser formuladas en su mayoría mediante los operadores diferenciales invariantes de primer orden, léase *Divergencia* y *Gradiente*. Por otra parte, es posible construir muchos esquemas de diferencias finitas en base a los operadores discretos de *Divergencia* y *Gradiente*. Comúnmente, los operadores discretos tradicionales no mantienen algunas propiedades físicas de los operadores continuos; por lo tanto, ha surgido

la necesidad de desarrollar métodos que permitan obtener operadores que si mantengan las leyes de conservación, simetría, entre otros; dichos métodos han sido denominados miméticos.

Recientemente se han desarrollado técnicas matriciales [?] para generar operadores discretos miméticos tanto de la divergencia como del gradiente. Como toda nueva técnica no ha sido utilizada en muchos problemas clásicos del campo científico, lo que ha traído como consecuencia el incremento de investigadores dedicados a su aplicación en los diversos campos del quehacer de la ciencia.

En el presente trabajo se pretende simular el flujo de fluidos sobre sólidos poligonales 2D utilizando la técnica del Confinamiento de la Vorticidad basado en un sistema de ED conocidas como ecuaciones de Navier-Stokes para el fluido no estacionario:

$$\frac{\partial v_x}{\partial t} + v_x \frac{\partial v_x}{\partial x} + v_y \frac{\partial v_x}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \left(\frac{\partial^2 v_x}{\partial x^2} + \frac{\partial^2 v_x}{\partial y^2} \right) \quad (1.1)$$

$$\frac{\partial v_y}{\partial t} + v_x \frac{\partial v_y}{\partial x} + v_y \frac{\partial v_y}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \nu \left(\frac{\partial^2 v_y}{\partial x^2} + \frac{\partial^2 v_y}{\partial y^2} \right) \quad (1.2)$$

donde: ν : es la viscosidad cinemática del fluido (constante)

ρ : es la densidad del fluido (constante)

p : es la presión del fluido en (x,y)

t : es el tiempo

v : es el vector velocidad

$(x, y) \in \Omega \subset \mathbb{R}^2$ variables dependientes

Lo anterior se puede escribir en función de los operadores continuos de divergencia, gradiente y laplaciano, como:

$$\frac{Dv}{\partial t} = -\frac{1}{\rho}\nabla p + \nu\nabla^2 v \quad \text{en } \Omega \quad (1.3)$$

En donde :

$$\begin{aligned}\nabla^2 v &= (\Delta v_x, \Delta v_y)^t \\ \nabla p &= \left(\frac{\partial p}{\partial x}, \frac{\partial p}{\partial y} \right)^t \\ \frac{Dv}{\partial t} &= \frac{\partial v}{\partial t} + v \cdot \nabla v \\ \nabla v &= \left(\frac{\partial v}{\partial x}, \frac{\partial v}{\partial y} \right)^t \\ v &= (v_x, v_y)^t \\ \Delta a &= \frac{\partial^2 a}{\partial x^2} + \frac{\partial^2 a}{\partial y^2} \quad a : \mathbb{R}^2 \rightarrow \mathbb{R}\end{aligned}$$

En (??) y (??) se plantearon las ecuaciones de *Navier - Stokes* para el flujo de fluido incompresible en 2D, pero es necesario definir otra ecuación (ecuación de continuidad) que permita completar el sistema para que tenga solución única:

$$\nabla.v = 0 \quad \text{en} \quad \Omega \quad (1.4)$$

El sistema de ecuaciones al ser tratado numéricamente y realizando una discretización del dominio, se obtiene un conjunto de n ecuaciones lineales con n incógnitas, representado por la forma:

$$\mathbf{Ax}=\mathbf{b}$$

donde: $\mathbf{A} \in \mathbb{R}^{n \times n}$ es la matriz de coeficientes

$\mathbf{x} \in \mathbb{R}^n$ es el vector de incógnitas

$\mathbf{b} \in \mathbb{R}^n$ es el vector de términos independientes

Para hallar una solución numérica al sistema de ecuaciones en derivadas parciales, se optará por el Método Castillo-Grone 2-2-2 [?] para construir los operadores discretos miméticos. Al surgir un sistema de ecuaciones lineales, el mismo debe ser resuelto por alguna biblioteca numérica. De esta manera se obtienen los resultados que permitan evaluar las ventajas y desventajas del método planteado. Posteriormente, es deseable poder visualizar de alguna manera el comportamiento dinámico de la simulación a través de su progreso en el tiempo, lo que exige la utilización de técnicas de visualización científica. En este trabajo se utiliza la biblioteca numérica *UCSparseLib* como herramienta de desarrollo para el programa de simulación y OpenGL como herramienta en la creación del software de visualización.

1.2. Objetivo

1.2.1. Objetivo General

Desarrollar un conjunto de programas capaces de simular el flujo de fluido sobre cuerpos poligonales en un dominio físico 2D utilizando la técnica del confinamiento de la vorticidad.

1.2.2. Objetivos Específicos

1. Estudiar la técnica matricial para construir operadores diferenciales discretos miméticos, dinámica de fluidos, la librería OpenGL y métodos iterativos para la solución numérica de sistemas de ecuaciones lineales, mediante la revisión detallada de material bibliográfico existente en el área de Diferencias Finitas Miméticas, fluidos, visualización y solución numérica de sistemas de ecuaciones lineales, con el fin de obtener los conocimientos necesarios para la resolución del problema planteado.
2. Diseñar un conjunto de herramientas computacionales para la simulación numérica del flujo de fluido y su interacción con el cuerpo poligonal utilizando los conocimientos y elementos ya considerados.
3. Diseñar módulos de software para UCSparseLib que permitan resolver el sistema de ecuaciones diferenciales parciales, aplicando el Método Castillo-Grone 2-2-2 para construir operadores discretos miméticos.
4. Implementar el diseño previamente elaborado, utilizando para ello un lenguaje de programación previamente seleccionado y la librería OpenGL

para el manejo de gráficos, con la finalidad de comprobar la efectividad del método utilizado.

5. Evaluar el software resultante de la implementación utilizando para ello el recurso de la visualización, con el cual se observará la dinámica del fluido interactuando con el cuerpo poligonal, observando su comportamiento lógico y haciendo comparaciones con algunos casos reportados en la bibliografía disponible.

1.3. Justificación

El movimiento de fluidos es increíblemente complejo. Esto depende de la interacción del fluido con el ambiente y de las partículas que fluyen en él. La dinámica de salpicar, romper, ondas y otros interesantes fenómenos requiere que la animación del mismo incluya un modelo complejo del movimiento. Los modelos existentes de Dinámica de Fluidos Computacional tienden a disipar las estructuras vorticales del fluido, degradando la precisión general del mismo; esta disipación puede ser reducida utilizando mallados finos pero a expensas de un gran incremento en la demanda de recursos computacionales. El método del confinamiento de la Vorticidad previene la disipación de la estructuras vorticales, teniendo como resultado flujos de configuraciones bastante complejas, implementados en mallados gruesos, a un bajo costo computacional. Muchos investigadores se han dedicado a resolver problemas de flujo de fluidos desde el punto de vista numérico, debido a la utilidad de las simulaciones y animaciones en el amplio campo donde pueden ser aplicados, lo cual reduce los costos de tiempo, dinero y esfuerzo.

Surge entonces la oportunidad de evaluar una nueva técnica como son los Métodos de Diferencias Finitas Miméticas y combinarlo con el Método de Confinamiento de la Vorticidad, con el fin de brindar a los investigadores la facilidad de resolver las ecuaciones que modelen el problema que se quiere solucionar y reducir los costos de las pruebas (personal operativo, equipos, dinero, tiempo y esfuerzo) en comparación a si se hiciese de manera real.

Dentro de un paquete de simulación, el tiempo es de gran interés por parte de los investigadores. Debido a que la resolución numérica de cada sistema de ecuaciones lineales, asociado a un paso de tiempo de un problema, puede tomar de minutos a horas de cómputo; el proceso completo para una simulación podría tardar de semanas a meses, haciendo que los resultados no se obtengan en el momento deseado. Para mejorar el tiempo que consume resolver los sistemas lineales, se emplean bibliotecas de software con métodos de resolución eficientes.

La representación gráfica de tales resultados, permite comprender visualmente el comportamiento dinámico de la simulación. Para ello es necesario considerar el uso de OpenGL como la más importante biblioteca para hardware gráfico existente en la actualidad, lo que ofrece mayor soporte, portabilidad y excelentes tiempos de respuesta en este aspecto de la simulación.

1.4. Antecedentes

Hoy en día se necesitan hacer simulaciones en distintas áreas, tales como ondas de calor, ondas de choque o de lo que se trata este trabajo: flujo de fluidos, así como el problema subyacente: Obtener la solución aproximada de diversas Ecuaciones Diferenciales.

En la actualidad existen grupos interdisciplinarios que desarrollan métodos relacionados con los operadores miméticos, así como también en el campo de la dinámica de fluidos. Entre los que se pueden presentar un conjunto de trabajos en ambas áreas:

1. En 1996 M. Shashkov y S. Steinberg [?] desarrollan un nuevo algoritmo de diferencias finitas para resolver ecuaciones de difusión con coeficientes complejos sobre un mallado rectangular general. El análisis de éste artículo será importante para la construcción de las rutinas a implementar.
2. En 1998 J. M. Hyman y M. Shashkov [?] describen como incorporar las condiciones de frontera dentro de los métodos de diferencias finitas y luego construyen métodos de diferencias finitas seguros para aproximar las ecuaciones de Maxwell y las ecuaciones de difusión del campo magnético usando análogos discretos de operadores diferenciales que satisfacen las identidades y teoremas del cálculo vectorial y tensorial en forma discreta, que se usarán para entender los análogos de las propiedades del método de

diferencias finitas miméticas.

3. En el año 2001 J. E. Castillo, junto con J. M. Hyman, M. Shashkov y S. Steinberg [?] derivan aproximaciones de diferencia finita de cuarto y sexto orden para los operadores de divergencia y gradiente así como un producto interno compatible sobre mallas estrechas en una dimensión en un dominio acotado, que son de suma de importancia. En éste artículo se explica de forma detallada como obtener éstos operadores discretos, que satisfacen las propiedades del teorema de Green, y que servirán de base para obtener algunos de los operadores discretos de diferencias finitas miméticas que se desean implementar.
4. En el año 2003 J. E. Castillo, junto R. D. Grone [?] construyen técnicas que permiten la construcción de versiones discretas de los Operadores Divergencia y Gradiente que tienen aproximaciones de alto orden en la frontera. Esto garantiza un alto orden de precisión general, es decir, el mismo orden de precisión tanto en la frontera como en el interior del dominio. El aprendizaje de este artículo será crucial, ya que en el se explica la técnica para construir operadores de orden superior, que servirán para construir las ecuaciones que al resolverlas, permitirán modelar el Flujo de Fluido interactuando con el Cuerpo poligonal.
5. En 1998, B. Jiang [?], utiliza el Método de Elementos finitos de Mínimos Cuadrados basado en la formulación velocidad-presión-vorticidad de primer orden, con el fin de resolver varios problemas relacionados con flujo viscoso compresible, entre ellos una adaptación rigurosa de las ecuaciones de Navier-Stokes y sus condiciones de frontera adicionales. Se realizará una revisión de las aplicaciones que aparecen en este texto a fin de seleccionar alguno de

ellos para hacer la comparación de resultados.

6. En [?] J. Stam propone por primera vez un modelo incondicionalmente estable para producir flujos de fluidos complejos que permite la interacción de los mismos con otros elementos en un escenario de simulación en 2D y 3D. Esto lo logra a través del modelo matemático de Navier-Stokes y un uso eficiente de técnicas propias a la graficación por computadora. Se realizará una revisión, la cual servirá de base teórica para el trabajo que se quiere realizar.
7. Posteriormente en [?] J. Stam presenta la implementación de un solver para fluidos, consistente con las ecuaciones de flujo de fluidos que producen campos de velocidad que contienen estructuras rotacionales compresibles y que reaccionan dinámicamente a fuerzas ejercidas por el usuario. Este solver especializado en fluidos que cubren el espacio, permite tomar ventaja de la Transformada de Fourier, lo que simplifica considerablemente muchos aspectos del mismo y puede ser usado como primitiva de movimientos básicos en diferentes aplicaciones en computación gráfica.
8. En el 2003 M. Fagúndez y J. Medina [?] desarrollan un software para la Simulación Numérica de Flujo de Fluido Viscoso Incompresible en 2D, utilizando para ello la librería UCSparseLib [?] - para la solución numérica del sistema de ecuaciones lineales por varios métodos - y OpenGL para la construcción del software de visualización de los resultados obtenidos, mediante la resolución de las Ecuaciones de Navier-Stokes.
9. En [?] Y. Wenren , M. Fan , W. Dietz , G. Hu , C. Braun , J. Steinhoff , B. Grossman, desarrollan un modelo euleriano eficiente para la simulación de

flujos de rotorcraft utilizando la técnica de confinamiento de la vorticidad. Este artículo servirá de base teórica para entender dicha técnica.

10. En [?] W. Dietz , M. Fan , J. Steinhoff ,Y. Wenren aplican la técnica de Confinamiento de la Vorticidad para predecir el comportamiento del flujo sobre cuerpos complejos. Este artículo servirá de base teórica para entender dicha técnica.
11. M. J. Harris en su Tesis Doctoral [?], desarrolló un software de simulación y barrido en tiempo real para nubes, utilizando la Técnica de Confinamiento de la Vorticidad así como Dinamica de Fluidos. Este trabajo servirá de base teórica para el estudio de la Dinamica de Fluidos así como de la técnica de Confinamiento de la Vorticidad.

Capítulo 2

MARCO TEÓRICO

En este capítulo se introducen algunos conceptos básicos de la dinámica de fluidos computacional, los fundamentos de las ecuaciones gobernantes, y algunos modelos clásicos de ello. Luego, se mostrarán las formulaciones de los métodos de diferencias finitas miméticas, que permiten aproximar la solución, haciendo una descomposición de las ecuaciones diferenciales de segundo orden en un sistema de ecuaciones lineales. Posteriormente se analizan algunos métodos de discretización del dominio continuo (mallados), el uso de las bibliotecas UCSparcelib y OpenGL, sobre las cuales se basa la implementación propuesta, y finalmente es presentada la metodología seguida en el diseño y desarrollo de los productos de software resultantes de esta investigación.

2.1. Conceptos Básicos

2.1.1. Mecánica de fluidos

Los *fluidos* se definen como todo material que no sea sólido y que puede "fluir". Son fluidos los líquidos y los gases; aún con sus grandes diferencias, sus comportamientos como fluidos se describe con las mismas ecuaciones básicas. La diferencia entre uno y otro está en su compresibilidad. Para cualquier sustancia, el estado líquido existe a una temperatura mayor que la del estado sólido, debido a que tiene una mayor agitación térmica y las fuerzas moleculares no son suficientes para mantener a las mismas en posiciones fijas. Lo común que tienen con los sólidos es que si actúan fuerzas externas de compresión, surgen grandes fuerzas atómicas que se resisten. En el estado gaseoso las moléculas tienen un continuo movimiento al azar y ejercen fuerzas muy débiles unas con otras y la separación promedio entre las moléculas de un gas, es mucho más grande que las dimensiones de las mismas.

El estudio de la dinámica de fluidos es similar al estudio clásico de la dinámica de sólidos, en la que se estudia el movimiento bajo la acción de fuerzas aplicadas. Se hace uso de los mismos principios de: conservación de masa, conservación del momentum y conservación de la energía termodinámica. Las ecuaciones de movimiento son dinámicas y las ecuaciones de continuidad son cinemáticas.

Propiedades de los fluidos

Entre las principales propiedades relacionadas con los fluidos y analizadas en el desarrollo de este trabajo de investigación, están:

- Densidad: Es la cantidad de masa por unidad de volumen.
- Peso específico: Es el peso por unidad de volumen.
- Viscosidad: Se refiere a la pegajosidad interna de un fluido. Es una de las propiedades que influye en la potencia necesaria para mover una superficie aerodinámica a través de la atmósfera. La velocidad de deformación de un fluido está directamente ligada a su viscosidad.
- Compresibilidad: Es la deformación provocada en el fluido por cambios de presión, lo que da como resultado un incremento de densidad.
- Tension superficial: es una propiedad originada por las fuerzas de atracción entre las moléculas.

Movimientos de fluido

Los movimientos de fluido se manifiestan de diferentes maneras. Algunos pueden ser descritos con facilidad, en tanto que otros requieren de un conocimiento completo de las leyes de física. En aplicaciones de ingeniería es importante describir los movimientos de fluidos tan simplemente como puedan ser justificados. Esto en general depende de la precisión requerida y de las suposiciones simplificadoras de las ecuaciones de movimiento, las cuales son muy difíciles de

resolver. Algunas suposiciones comunes utilizadas para simplificar una situación de flujo tienen que ver con las propiedades del fluido. Por ejemplo, en ciertas condiciones, la viscosidad puede afectar el flujo de manera significativa; en otras, los efectos viscosos pueden ser omitidos, con lo que se simplifican en gran medida las ecuaciones sin que se alteren significativamente las predicciones. La compresibilidad de un gas en movimiento deberá ser tomada en cuenta si las velocidades son muy altas, mas los efectos de compresibilidad no tienen que ser tomados en cuenta para predecir las fuerzas del viento que actúan en edificios o para predecir cualquier otra cantidad física que sea un efecto directo del viento. Después de estudiar los movimientos de fluidos, las suposiciones apropiadas utilizadas deberán ser más que obvias. El análisis de problemas de flujo de fluidos complejo a menudo se simplifica mediante la visualización de patrones de flujo, los que permiten desarrollar un mayor entendimiento intuitivo y ayudan a formular el problema matemático.

Clasificación de los flujos de fluido

- Flujos viscosos e invíscidos: Un flujo puede ser clasificado de una manera general como flujo viscoso o flujo invíscido. Un flujo invíscido es aquel en el que los efectos viscosos no influyen significativamente en el flujo y por lo tanto son ignorados. En un flujo viscoso los efectos de viscosidad son importantes y no pueden ser ignorados. Para modelar un flujo inviscido analíticamente, simplemente la viscosidad se hace cero; esto obviamente hace que todos los efectos viscosos sean cero. Con base en la experiencia, se encuentra que la clase principal de flujos que pueden ser modelados como flujos invíscidos, son los flujos externos, es decir, los flujos que existen en

el exterior de un cuerpo, tales como el flujo alrededor de una superficie aerodinámica o una superficie hidrodinámica. Los flujos viscosos incluyen la amplia clase de flujos internos, tales como flujos en tubos y conductos y en canales abiertos. En flujos como esos los efectos viscosos provocan pérdidas sustanciales y responden a las inmensas cantidades de energía que deben ser utilizadas para transportar petróleo y gas por oleoductos. La condición no deslizante que produce una velocidad cero en la pared, y los esfuerzos cortantes resultantes, conducen directamente a estas pérdidas.

- Flujo laminar: En un flujo laminar el fluido fluye sin mezclado significativo de sus partículas próximas entre sí. El flujo es ordenado y predecible, el movimiento se produce en capas o láminas y las soluciones matemáticas son factibles. En este flujo las partículas se mueven en trayectorias independientes de las partículas de capas adyacentes.
- Flujo turbulento: Los movimientos del fluido varían irregularmente de tal suerte que las cantidades tales como velocidad y presión muestran una variación aleatoria con el tiempo y las coordenadas espaciales. Las cantidades físicas con frecuencia se describen mediante promedios estadísticos. En este sentido, un flujo turbulento "continuo" puede ser definido como un flujo en el que las cantidades físicas promedio dependen del tiempo y no cambian con este.

2.1.2. Ecuaciones Diferenciales y Condiciones de Frontera

La mayoría de los fenómenos naturales son descritos por modelos matemáticos. Entre estos, las ecuaciones diferenciales son muy frecuentes y existen muchos tipos

de ellas. En general, una ecuación diferencial puede escribirse como

$$A(u) = f \quad \text{en } \Omega, \tag{2.1}$$

donde A es un operador diferencial, u y f son funciones suficientemente diferenciables para satisfacer las restricciones del operador A , y u se conoce como *función solución* o *variable dependiente*, por lo que A puede verse como una función que depende de u . El operador A se dice que es lineal en su argumento u , si y solo si se cumple que

$$A(\alpha u + \beta v) = \alpha A(u) + \beta A(v). \tag{2.2}$$

para todo escalar α, β , y funciones u, v . Si $f \equiv 0$ en (??), se dice que la ecuación es *homogénea*, en caso contrario, *no homogénea*. Además, Ω en (??) se conoce como el *dominio* de la ecuación diferencial, y para este contexto siempre se definirá real. Un dominio es una colección de puntos en el espacio con la propiedad de que si p es un punto del dominio, entonces todos los puntos suficientemente cercanos a p , pertenecen al dominio. Cuando un dominio tiene una *frontera*, se denomina dominio *acotado*. La frontera se define como un conjunto de puntos tales que en su vecindad, existen puntos que pueden pertenecer o no al dominio.

Cuando la variable dependiente es función de una variable, el dominio es un segmento de recta y su frontera son los dos puntos extremos. Cuando la variable dependiente es función de dos variables, el dominio es, comúnmente, una región del plano y su frontera es la curva que encierra (o delimita) la región.

Para garantizar la unicidad de la solución de la ecuación diferencial, es necesario imponer algunas condiciones sobre la variable dependiente. Cuando estas restricciones se aplican a los valores de la función solución y/o a sus derivadas en

los puntos de la frontera, se conoce como un *problema de valor de frontera*. Las condiciones de frontera se denotan como

$$B(u) = g \quad \text{sobre } \Gamma, \quad (2.3)$$

donde B es un operador diferencial, u la variable dependiente y g es una función definida en la frontera Γ . Dependiendo de cómo se formulan las condiciones de frontera se pueden tener diferentes tipos. Cuando se condiciona únicamente el valor de la variable dependiente, se conoce como condición de *Dirichlet*, y tiene la forma

$$hu = g \quad \text{sobre } \Gamma. \quad (2.4)$$

donde h es una función definida en Γ .

Cuando la condición prescribe el valor de la derivada de la variable dependiente, se conoce como condición de *Newmann*,

$$a \frac{\partial u}{\partial n} = g \quad \text{sobre } \Gamma, \quad (2.5)$$

donde n es el vector normal unitario externo a la frontera y a una función definida en Γ .

La condición donde se involucra tanto la variable dependiente como su derivada, se conoce como condición de *Robin*, *Newmann generalizada* o *mixta*, y es de la forma

$$a \frac{\partial u}{\partial n} + hu = g \quad \text{sobre } \Gamma \quad (2.6)$$

Si en cualquiera de los casos, $g \equiv 0$, se denomina condición *homogénea*, en otro caso se denomina *no homogénea*.

2.1.3. Operadores de Divergencia, Gradiente y Laplaciano

Entre los operadores más comunes, está el operador ∇ , el cual permite obtener la Divergencia, el Gradiente y Laplaciano de funciones escalares o vectoriales según sea el caso. Cada una de estas operaciones se define como:

Gradiente:

$$\nabla u = \begin{pmatrix} \frac{\partial u}{\partial x} \\ \frac{\partial u}{\partial y} \end{pmatrix}$$

Divergencia:

$$\nabla \cdot \vec{A} = \frac{\partial AX}{\partial x} + \frac{\partial AY}{\partial y}$$

Laplaciano:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

2.2. Métodos de Solución Numérica

Un número significativo de problemas físicos y de ingeniería, incluyen ecuaciones diferenciales con derivadas parciales. Es posible obtener soluciones analíticas de tales ecuaciones físico-matemática, solo en casos especiales, o bien puede ser

extremadamente laborioso. Como una alternativa a ello, surgen los métodos numéricos para poder obtener soluciones aproximadas de la ecuación diferencial.

Las funciones incógnitas en ecuaciones diferenciales, son funciones de argumentos continuos. Por ejemplo, la función $u(x, y, z, t)$ de la ecuación de calor, retorna el valor de temperatura en cada punto del espacio en el tiempo t . Esto significa que en el momento de tiempo t , puede asociarse a cada punto con coordenadas (x, y, z) un valor de temperatura $u(x, y, z, t)$. En el caso más simple, 1D estacionario, u es la función de un argumento espacial $x = x_1$, $u(x)$ y $x \in [a, b]$. La función $u(x)$ de argumento continuo $x \in [a, b]$, es elemento de un espacio de funciones H , y alguna norma es utilizada para comparar dos funciones $u(x), v(x) \in H$. Por ejemplo,

$$\|u\|_C = \max_{a \leq x \leq b} |u(x)| \quad (2.7)$$

$$\|u\|_{L_2} = \left(\int_a^b u^2(x) dx \right)^{1/2} \quad (2.8)$$

La escogencia de la norma para la comparación de estas funciones, depende de la interpretación física de la función u .

Para comprender lo que significa la *funcion de argumentos discretos*, se puede considerar el proceso de medición en un punto del espacio por un instrumento. Como resultado de tales mediciones, se obtiene un conjunto de números, cada uno de los cuales corresponde al valor en un punto. Estos valores pueden ser enumerados de alguna manera, por ejemplo asignando un número al instrumento

que toma cada medición, y puede ser denotado como u_i . En algunas aplicaciones físicas, los instrumentos pueden arrojar valores aproximados sobre un pequeño volumen o área, por lo que en este caso, los valores no corresponden a un punto en específico. Es claro que solo el índice i no representa información sobre la ubicación del instrumento con numeración i ; es por ello que la noción de función con argumentos discretos incluye información sobre la ubicación de los puntos, volúmenes u otra figuras geométricas y los valores relacionados a estos elementos geométricos. Ejemplos de mallados y funciones de mallados son dados en la siguiente sección.

2.2.1. Diferencia Finita Simple

Este es el método más antiguo para hallar solución numérica de ecuaciones en derivadas parciales, introducido por Euler en el siglo XVIII. Es además muy fácil de aplicar en geometrías simples. En cada punto del mallado, la ecuación diferencial es aproximada reemplazando las derivadas parciales por aproximaciones en términos de los valores nodales de las funciones. Aunque en principio el método puede ser aplicado sobre cualquier tipo de mallado, es común que se desarrolle únicamente para mallados estructurados. Por medio de expandir la serie de Taylor o interpolación polinomial, se obtienen aproximaciones para la primera y segunda derivada de las variables con respecto a las coordenadas. Como ya se mencionó, en mallados estructurados, el método de diferencias finitas es simple y efectivo, y es especialmente sencillo obtener esquemas de orden alto en mallados regulares. La desventaja que presentan es que la conservación no es considerada a menos que se dedique un análisis adicional. Además, la restricción

para solo geometrias simples es una desventaja significativa en flujos complejos.

2.2.2. Volumen Finito

Este método hace uso de la forma integral de la ecuación de conservación como punto de inicio. El dominio de la solución es dividido en un número finito de volúmenes de control contiguos y las ecuaciones de conservación son aplicadas a cada volumen. Es posible adaptar el método a cualquier tipo de mallado, por lo que es apropiado para geometrías complejas. Posiblemente las aproximaciones de Volumen Finito sean las más sencillas de entender y programar, y todos los términos que necesitan ser aproximados tienen interpretación física, lo que lo hace muy popular en ingeniería. La desventaja en comparación con los esquemas de Diferencias Finitas, es que el desarrollo para órdenes mayores a dos, es mucho más difícil de desarrollar en 3 dimensiones. Esto se debe a que el enfoque de Volumen Finito requiere tres niveles de aproximación: interpolación, diferenciación e integración.

2.2.3. Elemento Finito

Es similar al método de Volumen Finito en muchas maneras. El dominio es fraccionado en un conjunto de volúmenes discretos o elementos finitos que están generalmente estructurados; en 2D es común el uso de triángulos o cuadriláteros, mientras que en 3D son tetraedros o hexaedros. La característica distintiva de este método es que las ecuaciones son multiplicadas por una *función de peso* antes de ser integrada en el dominio. En la forma más simple del método de Elementos

Finitos, la solución es aproximada por una función lineal en cada elemento, de manera de garantizar continuidad de la solución a través de las fronteras de los elementos. Tal función puede ser construida a partir de los valores en las esquinas de los elementos. Esta aproximación es entonces sustituida dentro de la integral de la ley de conservación y las ecuaciones a ser resueltas son obtenidas a través de la derivada de la integral con respecto a cada valor nodal igual a cero; esto corresponde a seleccionar la mejor solución dentro de un conjunto de funciones permitidas (la de residual mínimo). El resultado es un conjunto de ecuaciones algebraicas no lineales. Una importante ventaja de este método es la capacidad de adaptarse a geometrías arbitrarias, y existe una extensa bibliografía dedicada a la construcción de mallas para métodos de elementos finitos. Los mallas son fácilmente refinados por medio de simplemente subdividir cada elemento. Además, los métodos de elementos finitos son fáciles de analizar matemáticamente y es posible demostrar sus propiedades de optimalidad para ciertos tipos de ecuaciones. La principal desventaja, compartida con cualquier método que utilice mallas no uniformes, es que las matrices de las ecuaciones linealizadas, no son estructuradas como aquellas en mallas uniformes, haciendo más difícil hallar métodos de soluciones eficientes.

2.2.4. Diferencia Finita Mimética

Entre los métodos más universales y efectivos, usado ampliamente en la actualidad para hallar soluciones aproximadas a ecuaciones físico-matemáticas, está el método de *diferencias finitas miméticas*, el cual resuelve la ecuación original reemplazando los operadores continuos de *divergencia* y *gradiente* y *laplaciano*

por aproximaciones numéricas que conserven propiedades importantes de los operadores originales.

Las soluciones para las ecuaciones diferenciales, son conseguidas en dos etapas:

- El desarrollo del esquema de diferencias finitas (una aproximación de diferencia para la ecuación diferencial sobre un mallado).
- El cálculo de la solución de las ecuaciones en diferencias, el cual es escrito en forma de un sistema de alto orden de ecuaciones algebraicas lineales o no lineales.

La esencia del método puede ser resumido como sigue:

1. El dominio continuo (por ejemplo, un intervalo, un rectángulo o un dominio con forma arbitraria), es reemplazado por un conjunto discreto de puntos (nodos).
2. En lugar de una función con argumentos continuos, se considera una con argumentos discretos. El valor de la función es definida en los nodos del mallado o en otros elementos del mallado (por ejemplo, en celdas que tienen nodos en sus vértices) y es llamada *función de mallado*.
3. Los valores de derivadas dentro de la ecuación diferencial y en las condiciones de frontera, son aproximados por las expresiones en diferencias. Así el problema diferencial es transformado en un sistema de ecuaciones algebraicas lineales o no lineales, lo que constituye el *esquema de diferencias finitas*.

Debido a lo reciente de la creación de este metodo, aun está siendo probado a diferentes niveles desde el punto de vista conceptual y de aplicación, con ecuaciones diferenciales diversas que permitan determinar sus ventajas y desventajas. Esta investigación estará orientada a un dominio físico 1D y 2D, ya que el método de Diferencias Finitas Miméticas propuesto para resolver las ecuaciones diferenciales de *Navier-Stokes*, no posee un desarrollo completo en 3D.

2.3. Generación de Mallados

2.3.1. Mallados en 2D

El ejemplo más simple de un mallado en 2D es llamado *producto tensorial* en un rectángulo. Sea el dominio de las variables x y y el rectángulo

$$R = 0 \leq x \leq a \times 0 \leq y \leq b$$

Sobre los intervalos $0 \leq x \leq a$ y $0 \leq y \leq b$ es construido un mallado 1D como en la seccion previa:

$$0 = x_1 < x_2 < \dots < x_i < x_{i+1} < \dots < x_{M-1} < x_M = a,$$

$$0 = y_1 < y_2 < \dots < y_j < y_{j+1} < \dots < y_{N-1} < y_N = b$$

El conjunto de nodos con coordenadas (x_i, y_j) sobre el plano es llamado mallado de producto tensorial o mallado rectangular no uniforme. Claramente, el mallado consiste en los puntos en donde las lineas $x = x_i$ y $y = y_j$) se intersectan. En el caso de mallados de producto tensorial, la densidad puede ser caracterizada por

el parámetro

$$h = \max(h_x, h_y) = \max\left(\frac{a}{M-1}, \frac{b}{N-1}\right)$$

el cual tiende a cero cuando el número de nodos incrementa.

El mallado es llamado *uniforme* en alguna dirección (x , y o ambos), si el mallado 1D en esa dirección es uniforme. Si el mallado es uniforme en ambas direcciones y los pasos h_x y h_y son iguales, el mallado es denominado *cuadrado*.

Para la identificación de cada nodo, se utilizan dos índices i, j . El rectángulo con vértices $(i, j), (i + 1, j), (i + 1, j + 1), (i, j + 1)$ (llamado celda) se usa para la identificación de esa celda.

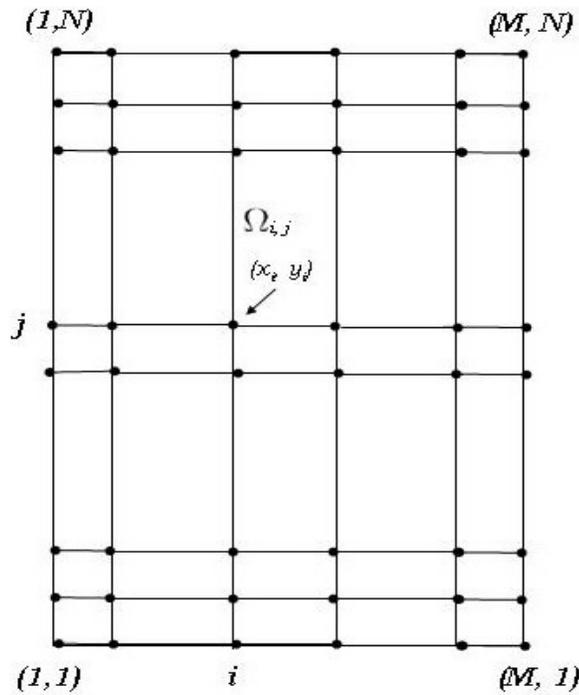


Figura 2.1: Mallado Tensorial

2.4. Método Castillo-Grone 2-2-2

En [?], J. Castillo y R. Grone muestran la manera de construir aproximaciones discretas de orden alto para la derivada en mallados uniformes unidimensionales y que satisfagan una ley de conservación global. Básicamente el problema consiste en hallar buenas aproximaciones de la divergencia ($\nabla \cdot$) y el gradiente (grad) que satisfagan el análogo discreto del teorema de la divergencia:

$$\int_{\Omega} \nabla \cdot \vec{v} f dV + \int_{\Omega} \vec{v} \text{grad} f dV = \int_{\partial\Omega} f \vec{v} \cdot \vec{n} dS \quad (2.9)$$

Para estos dos operadores existen tres ideas muy relacionadas: el *teorema de la divergencia* (??), conservación local y conservación global. La conservación local es un caso especial del teorema de la divergencia donde $f = 1$ y Ω una sola celda, mientras que la conservación global es (??) con $f = 1$ aplicado a la región completa bajo consideración. Las discretizaciones que poseen propiedades análogas a estas, son denominadas *miméticas*. Como tanto el teorema de la divergencia como la conservación local implican la conservación global, es relativamente sencillo hallar discretizaciones que satisfagan análogos a la conservación global.

En una configuración unidimensional dentro del intervalo $[0, 1]$, (??) es simplemente integración por partes:

$$\int_0^1 \frac{dv}{dx} f dx + \int_0^1 v \frac{df}{dx} dx = v(1)f(1) - v(0)f(0)$$

En las discretizaciones se utiliza el mallado de Operadores de Soporte, construido con $N > 0$, el cual es el numero de celdas, y $h = 1/N$. Los nodos o puntos en el mallado son $x_i = ih$, $0 \leq i \leq N$. Las celdas están dadas por el intervalo $[ih, (i+1)h]$ con centro $x_{i+\frac{1}{2}} = (i+\frac{1}{2})h$, $0 \leq i \leq N-1$. La divergencia discreta opera en los valores v , mientras que el gradiente discreto en los valores f .

Operadores discretos de segundo orden

La divergencia discreta más simple esta definida por

$$(\mathbf{D}v)_{i+\frac{1}{2}} = \frac{v_{i+1} - v_i}{h}, 0 \leq i \leq N-1, \quad (2.10)$$

Y el gradiente discreto esta definido por

$$(\mathbf{G}f)_0 = \frac{f_{\frac{1}{2}} - f_0}{h}, \quad (2.11)$$

$$(\mathbf{G}f)_i = \frac{f_{i+\frac{1}{2}} - f_{i-\frac{1}{2}}}{h}, 1 \leq i \leq N-1, \quad (2.12)$$

$$(\mathbf{G}f)_N = \frac{f_N - f_{N-\frac{1}{2}}}{h/2}, \quad (2.13)$$

$$(\mathbf{G}f)_0 = \frac{-\frac{4}{3}f_0 + \frac{3}{2}f_{\frac{1}{2}} - \frac{1}{6}f_{\frac{3}{2}}}{h/2}, \quad (2.14)$$

$$(\mathbf{G}f)_i = \frac{f_{i+\frac{1}{2}} - f_{i-\frac{1}{2}}}{h}, 1 \leq i \leq N-1, \quad (2.15)$$

$$(\mathbf{G}f)_N = \frac{\frac{4}{3}f_N - \frac{3}{2}f_{N-\frac{1}{2}} + \frac{1}{6}f_{N-\frac{3}{2}}}{h/2}, \quad (2.16)$$

donde nuevamente la definición de \mathbf{G} en los puntos de la frontera es la estándar dada por la aproximación de Operadores de Soporte. La divergencia \mathbf{D} es una aproximación de segundo orden, mientras que el gradiente \mathbf{G} es de segundo orden en los nodos internos y de primer orden en la frontera. La siguiente figura ilustra las posiciones de los valores de $\mathbf{D}v$ y $\mathbf{G}f$ en el mallado.

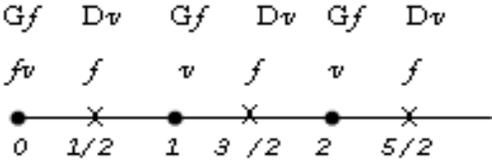


Figura 2.2: Distribución de los valores de Dv y Gf en el mallado

Los esquemas miméticos construidos con esta aproximación, conservan las propiedades fundamentales de los operadores continuos originales, permitiendo a las aproximaciones discretas de las ecuaciones en derivadas parciales imitar propiedades críticas, incluyendo las leyes de conservación y simetría en la solución de los problemas físicos subyacentes.

2.4.1. Método Implícito de Crank-Nicolson

El método de Crank-Nicolson se corresponde con la regla del trapecio para ecuaciones diferenciales ordinarias y se basa en diferencias centrales para hacer aproximaciones de segundo orden en el tiempo. Esta aproximación para la primera derivada es como sigue:

$$\partial_t v x_i^{n+1/2} = \frac{\partial_t v x_i^{n+1} + \partial_t v x_i^n}{2} \tag{2.17}$$

Los desarrollos basados en el esquema de Crank-Nicolson son incondicionalmente

estables, consistentes y convergentes para todos los valores en los pasos de tiempo, alcanzando aproximaciones considerablemente mejores que con métodos explícitos. Es utilizado conjuntamente con diferencias centrales espaciales con el propósito de alcanzar mayores precisiones.

2.4.2. Condiciones de Frontera en CFD

Todos los problemas en la dinámica de fluidos computacional son definidos en términos de las condiciones iniciales y de frontera. Es importante que el usuario especifique correctamente estos valores y comprenda el papel que desempeñan en los algoritmos numéricos. En los problemas no estacionarios, los valores iniciales de todas las variables del fluido necesitan ser establecidas en todos los puntos del dominio. Computacionalmente, esto significa inicializar apropiadamente las estructuras de datos utilizadas. Entre las condiciones de frontera comunmente utilizadas, se pueden mencionar:

- Condiciones de Entrada.
- Condiciones de Salida.
- Condiciones de Pared.
- Presiones prescritas o constantes.
- Simetría.
- Periodicidad (o condiciones de frontera cíclicas).

Haciendo uso de las condiciones de frontera, es posible establecer arbitrariamente la magnitud de una variable en un nodo del mallado con el proposito de simular situaciones especificas. Adicionalmente, es util modificar los valores de algunas variables en nodos internos, con el fin de simular obstáculos dentro del dominio. El sistema de las ecuaciones de fluido discretizado puede ser resuelto normalmente sin tener que tratar tales obstáculos de forma separada. Algunas condiciones de fronteras mas complejas, pueden incluir movimientos dependientes del tiempo, fronteras rotacionales o factores de aceleracion, condiciones especiales para flujos supersónicos, entre otras. Debido a la importancia de combinar correctamente los diferentes tipos de condiciones de frontera dentro de una simulación con el fin de obtener los resultados y aproximaciones deseadas, es tarea del usuario establecerlas partiendo de un cuidadoso analisis de los aspectos involucrados y de la experiencia que tenga en identificar las condiciones de la situación planteada.

2.5. Resolución de Sistemas de Ecuaciones Lineales

En muchos casos el resultado del proceso de discretización numérica, es un sistema de ecuaciones algebraicas lineales o no lineales dependiendo de la naturaleza de las ecuaciones diferenciales. Aun en los casos no lineales, los metodos de solucion envuelven resolver sistemas lineales. Es por ello que se hace necesario el uso de metodos eficientes para hallar solucion a tales sistemas. Las matrices resultantes de ecuaciones en derivadas parciales son siempre esparcidas, es decir, las mayorias de sus elementos son cero. Este hecho permite definir estructuras eficientes que almacenan y manipulan solo las coeficientes relevantes para resolver el sistema.

Entre los metodos directos se encuentran la Eliminación Gaussiana, Descomposición LU, Metodo de solucion Tridiagonal (*Algoritmo de Thomas*) y el metodo de Reducción Cíclica. Cualquier sistema de ecuaciones puede ser resuelto con estos algoritmos, pero desafortunadamente el costo de la triangulacion en matrices esparcidas es extremadamente alto. Esto justifica el uso de metodos iterativos, los cuales sistematicamente refinan una solucion inicial. El mas simple de estos metodos es el de Jacobi, Gauss-Seidel, SOR (*successive over-relaxation*), metodo de Stone (Descomposicion LU incompleta), Gradiente Conjugado, Gradiente Biconjugado, CGSTAB, GMRES y Multigrid.

2.5.1. Uso de la librería UCSparceLib

UCSparceLib es una biblioteca portable para la resolucion serial y paralela de sistemas lineares densos y esparcidos. Los sistemas lineales considerados son de la forma

$$Ax = b$$

donde A es una matrix $n \times n$ densa o esparcida de gran tamaño suministrada por el usuario, y puede ser regular o irregular en su estructura. b es vector del lado derecho. Las investigaciones sobre técnicas de matrices esparcidas han llegado a incrementar su complejidad, y promete continuar acentuandose debido a la necesidad creciente de diseñar algoritmos eficientes para modernas supercomputadoras. Aun cuando existe gran numero de paquetes y herramientas para resolver computacionalmente matrices densas de pequeño tamaño, hay tambien la necesidad de herramientas similares o librerias de proposito general

para el trabajo con matrices esparcidas. Una colección de unos cuantos programas básicos para realizar tareas comunes y elementales, puede ser muy útil y reducir el tiempo dedicado a implementar y probar algoritmos para matrices esparcidas. Los paquetes Linpack y Eispack desarrollados en los años 70, han sido de gran ayuda en varias áreas de la computación científica, ahorrando grandes cantidades de esfuerzo y tiempo. Por otro lado, es muy frecuente que los investigadores de la computación de matrices esparcidas, codifiquen sus propias subrutinas para los modos de almacenamiento de la matriz o de su reordenamiento acorde con ciertas permutaciones; una de las razones pudiera ser la ausencia de estándares. Por ello, para la misma estructura de datos básica, puede estar en uso un gran número de variaciones.

UCSparceLib es una biblioteca de propósito general debido a que puede manipular matrices generales no simétricas y con patrones de estructura irregulares. Los patrones de las posiciones diferentes de cero no necesariamente deben ser simétricos. Desarrollada en ANSI C, UCSparceLib resuelve sistemas lineales esparcidos y densos. Entre las funcionalidades incluidas, y que han sido de utilidad en el desarrollo de este trabajo, están:

- Rutinas para leer y escribir matrices utilizando un formato simple. También es posible transformar una matriz a un formato postscript.
- Operaciones básicas aplicadas a vectores, multiplicación de matrices, *solvers* triangulares y reordenamiento de matrices.
- *Solvers* Iterativos como *Gauss-Seidel*, *Jacobi*, Gradiente Conjugado, GMRES(m), BiCGstab y BiCG.

- Rutinas de utilidades adicionales como *timers*, tiempos de CPU y manejo de memoria.

La siguiente figura muestra la organización general de la biblioteca:

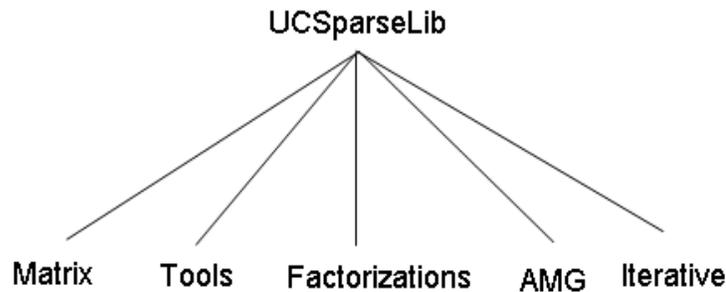


Figure 2.2: Organización General de UCSparseLib

Estructura de datos para matrices esparcidas o densas

Una de las mayores dificultades encontradas en el cálculo de matrices densas y esparcidas, es la variedad de tipos de matrices resultantes en aplicaciones prácticas. El propósito de un esquema de almacenamiento debe ser entonces, ganar eficiencia en términos de utilización de memoria y en operaciones aritméticas. Como resultado, se han propuesto muchas maneras diferentes de almacenar matrices esparcidas y densas, tomando ventajas de la estructura de las matrices o del problema específico del cual surgen. Uno de los esquemas más comunes de almacenamiento es el *Compressed Sparse Row* (CSR). En este esquema, todas las entradas no cero son almacenadas fila por fila en un arreglo de reales unidimensional, junto a un arreglo entero que contenga los índices de sus columnas y arreglos de apunadores a cada una de tales filas. El orden de los elementos dentro

de cada fila no es importante, lo cual simplifica el esquema de almacenamiento. En UCSparceLib una matriz es un objeto llamado `TDMatrix`, el cual es creado utilizando la siguiente rutina:

```
TDMatrixCreate( TDMatrix *M, TMatFormat format, int rows, int cols
);
```

donde `M` es una matriz, y debe ser declarada como `TDMatrix M`. El formato `format` es el modo de almacenamiento, el cual puede ser `MATRIX_DENSE_R`, `MATRIX_DENSE_C`, `MATRIX_SDENSE_R`, `MATRIX_SDENSE_C`, `MATRIX_DIAG`, `MATRIX_CSR`, `MATRIX_CSC`, `MATRIX_SCS_R`, `MATRIX_SCS_C`. `rows` y `cols` son las dimensiones de la matriz.

El objeto `TDMatrix` es definido como:

```
typedef struct DMatrix_t *TDMatrix;

struct DMatrix_t
{
    TMatFormat format; /* DENSE, CSR, CSC, DIAG */

    int    nn; /* IA dimension */
    int    mm; /* SDENSE_R, DENSE_R, SCSR or CSR -> nbcolls;*/
            /* SDENSE_C, DENSE_C, SCSC or CSC -> nbrows */
    size_t nnz; /* AN/JA dimension */
    TDSparseVec *ia; /* Point to the begin of each row/col */
    int    *ja; /* Col/Row of each AN element */
    double *an; /* No null entries of the matrix */
    Tcomplex *zan; /* No null entries of the matrix complex case */
    double *invd; /* Inverse of the diagonal */
    Tcomplex *zinvd; /* Inverse of the diagonal complex case */
    TLinkSparseVec *transpose; /* Linked list to access the transpose matrix */
    int    nblocks; /* For internal debugging */
};
```

```

    int    share_ja;    /* TRUE if ja is shared (allocated by the user) */
    int    share_an;    /* TRUE if an is shared (allocated by the user) */
};

```

Un componente importante del objeto TDMatrix es la estructura TDSparseVec.

Esta estructura representa un vector esparcido y es:

```

typedef struct
{
    int        nz;    /* No null elements */
    int        diag; /* Index of the diagonal element */
    int        *id;   /* col/row values */
    double     *val;  /* row/col elements */
} TDSparseVec;

```

Para acceder la fila *i* de la matriz TDMatrix *M*, se utiliza la siguiente macro:

```

For_TDMatrix_Row( M, i, row, mode ){
    .
    .
    .
}

```

donde, *row* es un TDSparceVec y *mode* puede ser: ACCESS_READ, ACCESS_WRITE
Y ACCESS_RW.

Existen tambien macros para acceder columnas o fila/columnas si el usuario no conoce el formato de la matriz. Las macros son: For_TDMatrix_Col y For_TDMatrix_RC, respectivamente.

2.6. Confinamiento de la Vorticidad

Fluidos tales como, el humo, las nubes, entre otros, contienen flujos rotacionales en una gran variedad de escalas. En Dinamica de Fluidos Computacional(CFD), la simulación de este tipo de flujos, tiende a ser deficiente, debido a la disipación numérica, como consecuencia de la utilización de mallados gruesos; ocasionando la perdida de información importante del fluido que se está simulando. Esta disipación se puede evitar, mediante la utilización de mallados finos; pero, a expensas de un considerable aumento del costo computacional, en el proceso de la simulación.

El método de Confinamiento de la Vorticidad, ha sido propuesto para reducir la propiedad difusiva de las simulaciones de los flujos vorticales. En este método, el término adherido a las ecuaciones de *Navier-Stokes*, trabaja como si devolviera el error resultante de la discretización, devuelta al centro del vórtice y así de esta manera confinarlo. Las ecuaciones en función de los operadores continuos, gradiente, divergencia y laplaciano, quedarían de la siguiente forma:

$$\frac{Dv}{\partial t} = -\frac{1}{\rho}\nabla p + \nu\nabla^2 v - \epsilon\vec{s} \quad \text{en } \Omega \quad (2.18)$$

En donde :

$$\vec{s} = \hat{n} \times \vec{\omega}; \quad \vec{\omega} = \nabla \times \vec{v}; \quad \eta = |\vec{\omega}|; \quad \hat{n} = \frac{\nabla\eta}{|\nabla\eta|}; \quad (2.19)$$

ϵ : es un parámetro que controla la fuerza del confinamiento

Ahora se desarrollan, las ecuaciones en 2.19:

$$\vec{\omega} = \begin{pmatrix} \hat{i} & \hat{j} & \hat{k} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \\ V_x & V_y & V_z \end{pmatrix} = \begin{pmatrix} -\frac{\partial V_y}{\partial z} \\ \frac{\partial V_x}{\partial z} \\ \frac{\partial V_y}{\partial x} - \frac{\partial V_x}{\partial y} \end{pmatrix}$$

Después de aplicar las operaciones convenientes, con $V_z = 0$, nos queda:

$$\begin{pmatrix} 0 \\ 0 \\ \frac{\partial V_y}{\partial x} - \frac{\partial V_x}{\partial y} \end{pmatrix}$$

Ahora desarrollamos η :

$$\eta = |\vec{\omega}| = \sqrt{0^2 + 0^2 + \left(\frac{\partial V_y}{\partial x} - \frac{\partial V_x}{\partial y}\right)^2} = \frac{\partial V_y}{\partial x} - \frac{\partial V_x}{\partial y}$$

Finalmente:

$$\nabla \eta = \begin{pmatrix} \frac{\partial^2 V_y}{\partial x^2} - \frac{\partial^2 V_x}{\partial x \partial y} \\ \frac{\partial^2 V_y}{\partial y \partial x} - \frac{\partial^2 V_x}{\partial y^2} \end{pmatrix}$$

Por lo tanto la norma de $\nabla \eta$ es:

$$|\nabla \eta| = \sqrt{\left(\frac{\partial^2 V_y}{\partial x^2} - \frac{\partial^2 V_x}{\partial x \partial y}\right)^2 + \left(\frac{\partial^2 V_y}{\partial y \partial x} - \frac{\partial^2 V_x}{\partial y^2}\right)^2}$$

En el capítulo 3, se desarrollarán las aproximaciones numéricas, de cada una de las expresiones analíticas, escritas en las líneas anteriores.

2.7. Métodos de Compresibilidad Artificial

Los flujos compresibles es un área de gran importancia en la mecánica de fluidos por sus aplicaciones en aerodinámica y diseño de turbinas, lo que ha llevado al desarrollo de métodos para su solución numérica. Surge entonces la posibilidad de adaptar tales métodos a la solución de flujos incompresibles. La mayor diferencia entre las ecuaciones de flujo compresible y los que no lo son, es netamente de carácter matemático. La versión compresible contiene la derivada en el tiempo de la densidad. En el caso de los flujos incompresibles la densidad es constante, lo cual elimina la opción de colocar tal derivada. Esto plantea que la derivada en el tiempo de la presión es una clara elección para estos casos, lo cual significa que realmente no se resuelven las ecuaciones de incompresibilidad. Como resultado, se ha cuestionado mucho la idea de utilizar métodos de compresibilidad artificial en problemas de flujo incompresible; sin embargo, ha sido ampliamente probado. Esta propuesta fue hecha por primera vez por Chorin en 1967, planteándola entre otras versiones posibles basadas en los métodos compresibles. En resumen, la propuesta esencial es agregar la derivada en el tiempo de la presión a la ecuación de continuidad:

$$\frac{1}{\beta} \frac{\partial p}{\partial t} + \frac{\partial V_x}{\partial x} + \frac{\partial V_y}{\partial y} = 0, \quad (2.20)$$

donde β es un parámetro de compresibilidad artificial cuyo valor es clave en el desempeño de este método. Obviamente, mientras mayor sea el valor de β , mayor será la “incompresibilidad” en las ecuaciones considerando el caso de densidad constante.

El factor crucial en la convergencia de este método, es la escogencia del parámetro

β , debido a que el valor óptimo es dependiente del problema. Aún así, algunos autores han sugerido procedimientos automáticos para tal selección, ya que valores demasiado grandes requieren de un correcto campo de velocidad para poder satisfacer la ecuación de continuidad incompresible. También es posible determinar el menor valor aceptable para β a través de la propagación de velocidad de las ondas de presión. Obviamente, $1/(\beta\Delta t)$ debe ser pequeño comparado con los coeficientes de los otros términos de la ecuación, lo cual es necesario si se desea una rápida convergencia.

2.8. OpenGL como API para la Visualización

OpenGL es una interfaz de software para hardware gráfico. Esta interfaz consiste en aproximadamente 150 comandos diferentes para especificar los objetos y operaciones necesarias para producir aplicaciones graficas interactivas.

OpenGL esta diseñado como una linea de procesos, y es una interfaz independiente de hardware que puede ser implementada en muchas plataformas diferentes.

Con el fin de satisfacer estas cualidades, no posee comandos para la creacion de ventanas u obtener entradas del usuario, por lo que se debe hacer uso de algun controlador del sistema de ventanas propio del hardware en particular. De igual manera, OpenGL no provee comandos de alto nivel para la creacion de objetos complejos; en su lugar, solo define un pequeño conjunto de primitivas geometricas tales como puntos, lineas y poligonos con los cuales es posible generar cualquier modelo deseado. Es por ello que se hace necesaria la construccion de todos los

elementos que conforman los graficos deseados para este trabajo, como una capa de abstraccion de software que tiene a OpenGL y otras librerias auxiliares como base.

Debido a que es posible reallizar muchas cosas con el sistema grafico de OpenGL, un programa puede llegar a ser complicado. Sin embargo, la estructura basica puede ser realmente sencilla: inicializar ciertas variable de estado que controlan la forma en la cual OpenGL dibuja y los objetos a ser dibujados. Entre la terminologia de la computacion grafica, la palabra Render implica el proceso en el cual un computador crea imagenes a partir de modelos. Estos modelos u objetos son construidos con primitivas geometricas (puntos, lineas y poligonos) que son especificadas por sus vertices.

El render final de la imgen consiste en pixels dibujados en la pantalla; un pixel es el elemento visible mas pequeño que el hardware grafico puede colocar en la pantalla. La informacion sobre los pixeles (como por ejemplo, el color que deben tener), es organizada en la memoria como mapas de bits. Un mapa de bit es un area de memoria que almacena un bit de informacion para cada pixel en la pantalla; un bit podria indicar cuan rojo se supone que debe ser un pixel en particular, por ejemplo.

Como ya se ha mencionado, OpenGL provee un conjunto de comandos primitivos pero poderoso, y todos tipo de dibujado de mas alto nivel debe ser hecho en terminos de tales comandos. Ademas, todos los programas hechos en OpenGL deben hacer uso de un mecanismo base de algun sistema especifico de manejo de ventanas. Existe un numero de bibliotecas que permiten simplificar algunas de

tales tareas de programación, como:

- OpenGL Utility Library (GLU), la cual contiene varias rutinas que hacen uso de los comandos de bajo nivel de OpenGL para llevar a cabo tareas como configuración de las matrices que especifican la orientación y proyección de las vistas, y dibujo de superficies. Esta biblioteca es distribuida como parte de todas las implementaciones de OpenGL.
- OpenGL Utility Toolkit (GLUT) es un conjunto de herramientas de ventanas independiente del sistema, que simplifica con pocas rutinas la necesidad de crear el entorno de dibujo requerido por OpenGL. La mayoría de las variables de estado se encuentran inicializadas por defecto en valores que facilitan el desarrollo en programas simples. Las rutinas igualmente requieren pocos parámetros. Por estas razones de simplicidad, GLUT no hace uso de manejadores, apuntadores o estructuras de datos nativas a algún sistema en particular; incluso provee de su propio conjunto (limitado) de fuentes.

2.9. Metodología de Ciclo de Vida del desarrollo de software

El desarrollo de software va unido a un ciclo de vida compuesto por una serie de etapas que comprenden todas las actividades, desde el momento en que surge la idea de crear un nuevo producto software, hasta aquel en que el producto deja definitivamente de ser utilizado por el último de sus usuarios.

Entre las diversas prácticas inherentes al Ciclo de Vida del Desarrollo de Software,

y que son consideradas en cada uno de los ciclos de desarrollo de este proyecto, se tiene:

2.9.1. Análisis

Es necesario determinar que elementos intervienen en el sistema a desarrollar, así como su estructura, relaciones, evolución en el tiempo, detalle de sus funcionalidades, que van a dar una descripción clara de qué sistema vamos a construir, qué funcionalidades va a aportar y qué comportamiento va a tener.

2.9.2. Diseño

Tras la etapa anterior ya se tiene claro que debe hacer el sistema, ahora tenemos que determinar como va a hacerlo; aquí se definirán en detalle entidades y relaciones de las bases de datos, se pasará de casos de uso esenciales a su definición como casos expandidos reales, se seleccionará el lenguaje más adecuado, el Sistema Gestor de Bases de Datos a utilizar en su caso, librerías, configuraciones hardware, redes, etc.).

2.9.3. Implementación

Llegado este punto se empieza a codificar algoritmos y estructuras de datos, definidos en las etapas anteriores, en el correspondiente lenguaje de programación y/o para un determinado sistema gestor de bases de datos.

2.9.4. Pruebas

El objetivo de estas pruebas es garantizar que el sistema ha sido desarrollado correctamente, sin errores de diseño y/o programación. Es conveniente que sean planteadas al menos tanto a nivel de cada módulo (aislado del resto), como de integración del sistema (según sea la naturaleza del proyecto en cuestión se podrán tener en cuenta pruebas adicionales, p.ej. de rendimiento).

Las pruebas se convierten en una herramienta de desarrollo, no en un paso de verificación que pueda despreciarse si se cree que el código está bien. Las unidades de prueba serán incluidas junto con el código que verifican dentro del repositorio. El código no se considerará completo si el mismo no consta de su unidad de prueba correspondiente. El código será implantado cuando supere sus correspondientes unidades de prueba.

Capítulo 3

DESARROLLO

En el capítulo anterior se expusieron diferentes métodos para resolver una ecuación diferencial parcial de segundo orden, transformando la ecuaciones diferenciales en un sistema de ecuaciones lineales. En este capítulo se utilizará el método Castillo-Grone 2-2-2, en un sistema de ecuaciones diferenciales de primer orden. Además, se mostrarán detalles de la librería UCSparcelib, utilizada como la base principal para el desarrollo computacional del método en la ecuación de estudio de este trabajo. Finalmente, se mostrará el desarrollo de procedimientos que permiten representar gráficamente el resultado del cálculo numérico de las ecuaciones implicadas en la simulación de flujo de fluido.

3.1. Solución Numérica de las Ecuaciones de Navier-Stokes y Compresibilidad Artificial en 2D, utilizando la técnica de Castillo-Grone 2-2-2 y Confinamiento de la Vorticidad

En la siguiente sección se desarrollaran las aproximaciones de las Ecuaciones de Navier-Stokes, tanto en los nodos internos como en el obstáculo y la frontera del mallado.

3.1.1. Aproximación de las Ecuaciones de Navier-Stokes, con el Término de Confinamiento de la Vorticidad y Compresibilidad Artificial, en los nodos internos del mallado

La ecuación de *Navier - Stokes* (2D) puede ser escrita como:

$$\frac{\partial V_x}{\partial t} + V_x \frac{\partial V_x}{\partial x} + V_y \frac{\partial V_x}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \left(\frac{\partial^2 V_x}{\partial x^2} + \frac{\partial^2 V_x}{\partial y^2} \right) - \epsilon S_x \quad (3.1)$$

$$\frac{\partial V_y}{\partial t} + V_x \frac{\partial V_y}{\partial x} + V_y \frac{\partial V_y}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \nu \left(\frac{\partial^2 V_y}{\partial x^2} + \frac{\partial^2 V_y}{\partial y^2} \right) - \epsilon S_y \quad (3.2)$$

La aproximación del Operador Divergencia está definida como:

$$(Df)_{i+1/2} \approx \frac{f_{i+1} - f_i}{h}$$

La aproximación del Operador Gradiente lo podemos definir como:

$$(Gf)_i \approx \frac{f_{i+1/2} - f_{i-1/2}}{h}$$

y para las fronteras:

$$(Gf)_0 \approx \frac{-4/3f_0 + 3/2f_{1/2} - 1/6f_{3/2}}{h/2}$$

$$(Gf)_N \approx \frac{-4/3f_N + 3/2f_{N-1/2} - 1/6f_{N-3/2}}{h/2}$$

Siendo hx y hy igual al espaciado uniforme del mallado en X y en Y ; para nuestro caso $hx = \Delta x$ o $hx = \Delta y$, con $\Delta x = x_{i+1} - x_i$ y $\Delta y = y_{i+1} - y_i$ respectivamente.

Se Evalua en los puntos medios y en el tiempo actual:

$$\begin{aligned} \partial_t V x_{i+1/2, j+1/2}^n + V x_{i+1/2, j+1/2}^n \partial_x V x_{i+1/2, j+1/2}^n + V y_{i+1/2, j+1/2}^n \partial_y V x_{i+1/2, j+1/2}^n = \\ -\frac{1}{\rho} \partial_x p_{i+1/2, j+1/2}^n + \nu (\partial_{xx}^2 V x_{i+1/2, j+1/2}^n + \partial_{yy}^2 V x_{i+1/2, j+1/2}^n) \end{aligned}$$

Aproximación de la derivada con respecto al tiempo:

$$\partial_t f \approx \frac{f_i^{n+1} - f_i^n}{\Delta t}$$

Las aproximaciones en los puntos medios se puede calcular como el promedio de los nodos enteros (en X) del mallado, estos promedios se escriben a continuación:

$$\text{Donde } f_{i+1/2, j+1/2} = \frac{f_{i+1, j+1/2} + f_{i, j+1/2}}{2} + O(h^2)$$

$$\text{Donde } f_{i-1/2, j+1/2} = \frac{f_{i, j+1/2} + f_{i-1, j+1/2}}{2} + O(h^2)$$

$$\text{Donde } f_{i+3/2, j+1/2} = \frac{f_{i+2, j+1/2} + f_{i+1, j+1/2}}{2} + O(h^2)$$

Aplicamos Crank-Nicholson:

$$\begin{aligned} & \partial_t V x_{i+1/2,j+1/2}^n + V x_{i+1/2,j+1/2}^n \left(\frac{\partial_x V x_{i+1/2,j+1/2}^{n+1} + \partial_x V x_{i+1/2,j+1/2}^n}{2} \right) + \\ & V y_{i+1/2,j+1/2}^n \left(\frac{\partial_y V x_{i+1/2,j+1/2}^{n+1} + \partial_y V x_{i+1/2,j+1/2}^n}{2} \right) = -\frac{1}{\rho} \left(\frac{\partial_x p_{i+1/2,j+1/2}^{n+1} + \partial_x p_{i+1/2,j+1/2}^n}{2} \right) + \\ & \nu \left(\frac{\partial_{xx}^2 V x_{i+1/2,j+1/2}^{n+1} + \partial_{xx}^2 V x_{i+1/2,j+1/2}^n}{2} + \frac{\partial_{yy}^2 V x_{i+1/2,j+1/2}^{n+1} + \partial_{yy}^2 V x_{i+1/2,j+1/2}^n}{2} \right) \end{aligned}$$

Se sustituye por los promedios:

$$\begin{aligned} & \partial_t V x_{i+1/2,j+1/2}^n + V x_{i+1/2,j+1/2}^n \left(\frac{\frac{\partial_x V x_{i+1,j+1/2}^{n+1} + \partial_x V x_{i,j+1/2}^{n+1}}{2} + \frac{\partial_x V x_{i+1,j+1/2}^n + \partial_x V x_{i,j+1/2}^n}{2}}{2} \right) \\ & + V y_{i+1/2,j+1/2}^n \left(\frac{\frac{\partial_y V x_{i+1/2,j+1}^{n+1} + \partial_y V x_{i+1/2,j}^{n+1}}{2} + \frac{\partial_y V x_{i+1/2,j+1}^n + \partial_y V x_{i+1/2,j}^n}{2}}{2} \right) = \\ & -\frac{1}{\rho} \left(\frac{\frac{\partial_x p_{i+1,j+1/2}^{n+1} + \partial_x p_{i,j+1/2}^{n+1}}{2} + \frac{\partial_x p_{i+1,j+1/2}^n + \partial_x p_{i,j+1/2}^n}{2}}{2} \right) \\ & + \nu \left(\frac{\frac{\partial_{xx}^2 V x_{i+1/2,j+1/2}^{n+1} + \partial_{xx}^2 V x_{i+1/2,j+1/2}^n}{2} + \frac{\partial_{yy}^2 V x_{i+1/2,j+1/2}^{n+1} + \partial_{yy}^2 V x_{i+1/2,j+1/2}^n}{2}}{2} \right) \end{aligned}$$

Se Despeja:

$$\begin{aligned} & \partial_t V x_{i+1/2,j+1/2}^n + \frac{1}{\rho} \left(\frac{\partial_x p_{i+1,j+1/2}^{n+1} + \partial_x p_{i,j+1/2}^{n+1}}{4} \right) - \nu \left(\frac{\partial_{xx}^2 V x_{i+1/2,j+1/2}^{n+1} + \partial_{yy}^2 V x_{i+1/2,j+1/2}^{n+1}}{2} \right) \\ & + V x_{i+1/2,j+1/2}^n \left(\frac{\partial_x V x_{i+1,j+1/2}^{n+1} + \partial_x V x_{i,j+1/2}^{n+1}}{4} \right) + V y_{i+1/2,j+1/2}^n \left(\frac{\partial_y V x_{i+1/2,j+1}^{n+1} + \partial_y V x_{i+1/2,j}^{n+1}}{4} \right) \\ & = -\frac{1}{\rho} \left(\frac{\partial_x p_{i+1,j+1/2}^n + \partial_x p_{i,j+1/2}^n}{4} \right) + \nu \left(\frac{\partial_{xx}^2 V x_{i+1/2,j+1/2}^n + \partial_{yy}^2 V x_{i+1/2,j+1/2}^n}{2} \right) \\ & - V x_{i+1/2,j+1/2}^n \left(\frac{\partial_x V x_{i+1,j+1/2}^n + \partial_x V x_{i,j+1/2}^n}{4} \right) - V y_{i+1/2,j+1/2}^n \left(\frac{\partial_y V x_{i+1/2,j+1}^n + \partial_y V x_{i+1/2,j}^n}{4} \right) \end{aligned}$$

Se aplica las aproximaciones:

$$\begin{aligned}
& \frac{Vx_{i+1/2,j+1/2}^{n+1} - Vx_{i+1/2,j+1/2}^n}{\Delta t} + \frac{1}{\rho} \left(\frac{(p_{i+3/2,j+1/2}^{n+1} - p_{i+1/2,j+1/2}^{n+1}) + (p_{i+1/2,j+1/2}^{n+1} - p_{i-1/2,j+1/2}^{n+1})}{4\Delta x} \right) \\
& - \nu \left(\frac{(Vx_{i+3/2,j+1/2}^{n+1} - Vx_{i+1/2,j+1/2}^{n+1}) - (Vx_{i+1/2,j+1/2}^{n+1} - Vx_{i-1/2,j+1/2}^{n+1})}{2(\Delta x)^2} \right) \\
& + \frac{(Vx_{i+1/2,j+3/2}^{n+1} - Vx_{i+1/2,j+1/2}^{n+1}) - (Vx_{i+1/2,j+1/2}^{n+1} - Vx_{i+1/2,j-1/2}^{n+1})}{2(\Delta y)^2} \\
& + Vx_{i+1/2,j+1/2}^n \left(\frac{(Vx_{i+3/2,j+1/2}^{n+1} - Vx_{i+1/2,j+1/2}^{n+1}) + (Vx_{i+1/2,j+1/2}^{n+1} - Vx_{i-1/2,j+1/2}^{n+1})}{4\Delta x} \right) \\
& + Vy_{i+1/2,j+1/2}^n \left(\frac{(Vx_{i+1/2,j+3/2}^{n+1} - Vx_{i+1/2,j+1/2}^{n+1}) + (Vx_{i+1/2,j+1/2}^{n+1} - Vx_{i+1/2,j-1/2}^{n+1})}{4\Delta y} \right) \\
& = -\frac{1}{\rho} \left(\frac{(p_{i+3/2,j+1/2}^n - p_{i+1/2,j+1/2}^n) + (p_{i+1/2,j+1/2}^n - p_{i-1/2,j+1/2}^n)}{4\Delta x} \right) \\
& + \nu \left(\frac{(Vx_{i+3/2,j+1/2}^n - Vx_{i+1/2,j+1/2}^n) - (Vx_{i+1/2,j+1/2}^n - Vx_{i-1/2,j+1/2}^n)}{2(\Delta x)^2} \right) \\
& + \frac{(Vx_{i+1/2,j+3/2}^n - Vx_{i+1/2,j+1/2}^n) - (Vx_{i+1/2,j+1/2}^n - Vx_{i+1/2,j-1/2}^n)}{2(\Delta y)^2} \\
& - Vx_{i+1/2,j+1/2}^n \left(\frac{(Vx_{i+3/2,j+1/2}^n - Vx_{i+1/2,j+1/2}^n) + (Vx_{i+1/2,j+1/2}^n - Vx_{i-1/2,j+1/2}^n)}{4\Delta x} \right) \\
& - Vy_{i+1/2,j+1/2}^n \left(\frac{(Vx_{i+1/2,j+3/2}^n - Vx_{i+1/2,j+1/2}^n) + (Vx_{i+1/2,j+1/2}^n - Vx_{i+1/2,j-1/2}^n)}{4\Delta y} \right)
\end{aligned}$$

Al simplificar:

$$\begin{aligned}
& \frac{Vx_{i+1/2,j+1/2}^{n+1}}{\Delta t} + \frac{1}{\rho} \left(\frac{p_{i+3/2,j+1/2}^{n+1} - p_{i-1/2,j+1/2}^{n+1}}{4\Delta x} \right) \\
& - \nu \left(\frac{Vx_{i+3/2,j+1/2}^{n+1} - 2Vx_{i+1/2,j+1/2}^{n+1} + Vx_{i-1/2,j+1/2}^{n+1}}{2(\Delta x)^2} \right. \\
& \quad \left. + \frac{Vx_{i+1/2,j+3/2}^{n+1} - 2Vx_{i+1/2,j+1/2}^{n+1} + Vx_{i+1/2,j-1/2}^{n+1}}{2(\Delta y)^2} \right) \\
& + Vx_{i+1/2,j+1/2}^n \left(\frac{Vx_{i+3/2,j+1/2}^{n+1} + Vx_{i-1/2,j+1/2}^{n+1}}{4\Delta x} \right) + Vy_{i+1/2,j+1/2}^n \left(\frac{Vx_{i+1/2,j+3/2}^{n+1} + Vx_{i+1/2,j-1/2}^{n+1}}{4\Delta y} \right) \\
& = \frac{Vx_{i+1/2,j+1/2}^n}{\Delta t} - \frac{1}{\rho} \left(\frac{p_{i+3/2,j+1/2}^n - p_{i-1/2,j+1/2}^n}{4\Delta x} \right) \\
& + \nu \left(\frac{Vx_{i+3/2,j+1/2}^n - 2Vx_{i+1/2,j+1/2}^n + Vx_{i-1/2,j+1/2}^n}{2(\Delta x)^2} \right. \\
& \quad \left. + \frac{Vx_{i+1/2,j+3/2}^n - 2Vx_{i+1/2,j+1/2}^n + Vx_{i+1/2,j-1/2}^n}{2(\Delta y)^2} \right) \\
& - Vx_{i+1/2,j+1/2}^n \left(\frac{Vx_{i+3/2,j+1/2}^n - Vx_{i-1/2,j+1/2}^n}{4\Delta x} \right) - Vy_{i+1/2,j+1/2}^n \left(\frac{Vx_{i+1/2,j+3/2}^n - Vx_{i+1/2,j-1/2}^n}{4\Delta y} \right)
\end{aligned}$$

Agrupamos términos semejantes:

$$\begin{aligned}
& \left(\frac{1}{\Delta t} + \frac{\nu}{(\Delta x)^2} + \frac{\nu}{(\Delta y)^2} \right) Vx_{i+1/2,j+1/2}^{n+1} + \frac{1}{4\rho\Delta x} p_{i+3/2,j+1/2}^{n+1} - \frac{1}{4\rho\Delta x} p_{i-1/2,j+1/2}^{n+1} \\
& + \left(\frac{Vx_{i+1/2,j+1/2}^n}{4\Delta x} - \frac{\nu}{2(\Delta x)^2} \right) Vx_{i+3/2,j+1/2}^{n+1} + \left(\frac{Vx_{i+1/2,j+1/2}^n}{4\Delta x} - \frac{\nu}{2(\Delta x)^2} \right) Vx_{i-1/2,j+1/2}^{n+1} \\
& + \left(\frac{Vy_{i+1/2,j+1/2}^n}{4\Delta y} - \frac{\nu}{2(\Delta y)^2} \right) Vx_{i+1/2,j+3/2}^{n+1} + \left(\frac{Vy_{i+1/2,j+1/2}^n}{4\Delta y} - \frac{\nu}{2(\Delta y)^2} \right) Vx_{i+1/2,j-1/2}^{n+1} \\
& = -\frac{1}{\rho} \left(\frac{p_{i+3/2,j+1/2}^n - p_{i-1/2,j+1/2}^n}{4\Delta x} \right) + \nu \frac{Vx_{i+3/2,j+1/2}^n - 2Vx_{i+1/2,j+1/2}^n + Vx_{i-1/2,j+1/2}^n}{2(\Delta x)^2} \\
& \quad + \nu \frac{Vx_{i+1/2,j+3/2}^n - 2Vx_{i+1/2,j+1/2}^n + Vx_{i+1/2,j-1/2}^n}{2(\Delta y)^2} + \frac{Vx_{i+1/2,j+1/2}^n}{\Delta t} \\
& - Vx_{i+1/2,j+1/2}^n \left(\frac{Vx_{i+3/2,j+1/2}^n - Vx_{i-1/2,j+1/2}^n}{4\Delta x} \right) - Vy_{i+1/2,j+1/2}^n \left(\frac{Vx_{i+1/2,j+3/2}^n - Vx_{i+1/2,j-1/2}^n}{4\Delta y} \right)
\end{aligned}$$

Desarrollamos la ecuación 3.2 de *Navier - Stokes*:

$$-\frac{1}{\rho} \frac{\partial p}{\partial y} = -\frac{1}{\rho} \left(\frac{\partial_y p_{i+1/2, j+1/2}^{n+1} + \partial_y p_{i+1/2, j+1/2}^n}{2} \right)$$

y

$$\begin{aligned} \partial_y p_{i+1/2, j+1/2}^{n+1} &= \frac{\partial_y p_{i+1/2, j+1}^{n+1} + \partial_y p_{i+1/2, j}^{n+1}}{2} \\ \partial_y p_{i+1/2, j+1/2}^n &= \frac{\partial_y p_{i+1/2, j+1}^n + \partial_y p_{i+1/2, j}^n}{2} \end{aligned}$$

Se Aplica la Aproximacion con el Operador Gradiente:

$$\begin{aligned} \partial_y p_{i+1/2, j+1/2}^{n+1} &= \frac{p_{i+1/2, j+3/2}^{n+1} - p_{i+1/2, j-1/2}^{n+1}}{4\Delta y} \\ \partial_y p_{i+1/2, j+1/2}^n &= \frac{p_{i+1/2, j+3/2}^n - p_{i+1/2, j-1/2}^n}{4\Delta y} \end{aligned}$$

Sustituyendo:

$$-\frac{1}{\rho} \frac{\partial p}{\partial y} = -\frac{1}{\rho} \left(\frac{p_{i+1/2, j+3/2}^{n+1} - p_{i+1/2, j-1/2}^{n+1}}{4\Delta y} + \frac{p_{i+1/2, j+3/2}^n - p_{i+1/2, j-1/2}^n}{4\Delta y} \right)$$

Podemos decir que la ecuación 3.2 de *Navier - Stokes* se puede escribir como:

$$\begin{aligned} & \left(\frac{1}{\Delta t} + \frac{\nu}{(\Delta x)^2} + \frac{\nu}{(\Delta y)^2} \right) V y_{i+1/2, j+1/2}^{n+1} + \frac{1}{4\rho\Delta y} p_{i+1/2, j+3/2}^{n+1} - \frac{1}{4\rho\Delta y} p_{i+1/2, j-1/2}^{n+1} \\ & + \left(\frac{V x_{i+1/2, j+1/2}^n}{4\Delta x} - \frac{\nu}{2(\Delta x)^2} \right) V y_{i+3/2, j+1/2}^{n+1} + \left(\frac{V x_{i+1/2, j+1/2}^n}{4\Delta x} - \frac{\nu}{2(\Delta x)^2} \right) V y_{i-1/2, j+1/2}^{n+1} \\ & + \left(\frac{V y_{i+1/2, j+1/2}^n}{4\Delta y} - \frac{\nu}{2(\Delta y)^2} \right) V y_{i+1/2, j+3/2}^{n+1} + \left(\frac{V y_{i+1/2, j+1/2}^n}{4\Delta y} - \frac{\nu}{2(\Delta y)^2} \right) V y_{i+1/2, j-1/2}^{n+1} \\ & = -\frac{1}{\rho} \left(\frac{p_{i+1/2, j+3/2}^n - p_{i+1/2, j-1/2}^n}{4\Delta y} \right) + \nu \frac{V y_{i+3/2, j+1/2}^n - 2V y_{i+1/2, j+1/2}^n + V y_{i-1/2, j+1/2}^n}{2(\Delta x)^2} \\ & \quad + \nu \frac{V y_{i+1/2, j+3/2}^n - 2V y_{i+1/2, j+1/2}^n + V y_{i+1/2, j-1/2}^n}{2(\Delta y)^2} + \frac{V y_{i+1/2, j+1/2}^n}{\Delta t} \\ & - V x_{i+1/2, j+1/2}^n \left(\frac{V y_{i+3/2, j+1/2}^n - V y_{i-1/2, j+1/2}^n}{4\Delta x} \right) - V y_{i+1/2, j+1/2}^n \left(\frac{V y_{i+1/2, j+3/2}^n - V y_{i+1/2, j-1/2}^n}{4\Delta y} \right) \end{aligned}$$

Ecuación de Compresibilidad Artificial 2D:

$$\frac{1}{\beta} \partial_t p_{i+1/2,j+1/2}^n + (\partial_x v x_{i+1/2,j+1/2}^n + \partial_y v y_{i+1/2,j+1/2}^n) = 0$$

Aplicamos Crank-Nicholson:

$$\begin{aligned} \frac{1}{\beta} \left(\frac{p_{i+1/2,j+1/2}^{n+1} - p_{i+1/2,j+1/2}^n}{\Delta t} \right) + \frac{\partial_x v x_{i+1/2,j+1/2}^{n+1} + \partial_x v x_{i+1/2,j+1/2}^n}{2} \\ + \frac{\partial_y v y_{i+1/2,j+1/2}^{n+1} + \partial_y v y_{i+1/2,j+1/2}^n}{2} = 0 \end{aligned}$$

Aproximaciones Discretas:

$$\begin{aligned} \frac{1}{\beta} \left(\frac{p_{i+1/2,j+1/2}^{n+1} - p_{i+1/2,j+1/2}^n}{\Delta t} \right) + \frac{v x_{i+1,j+1/2}^{n+1} - v x_{i,j+1/2}^{n+1}}{2\Delta x} + \frac{v x_{i+1,j+1/2}^n - v x_{i,j+1/2}^n}{2\Delta x} \\ + \frac{v y_{i+1/2,j+1}^{n+1} - v y_{i+1/2,j}^{n+1}}{2\Delta y} + \frac{v y_{i+1/2,j+1}^n - v y_{i+1/2,j}^n}{2\Delta y} = 0 \end{aligned}$$

Promedios:

$$\begin{aligned} v x_{i+1,j+1/2}^{n+1} &= \frac{v x_{i+3/2,j+1/2}^{n+1} + v x_{i+1/2,j+1/2}^{n+1}}{2} \\ v x_{i,j+1/2}^{n+1} &= \frac{v x_{i+1/2,j+1/2}^{n+1} + v x_{i-1/2,j+1/2}^{n+1}}{2} \\ v y_{i+1/2,j+1}^{n+1} &= \frac{v y_{i+1/2,j+3/2}^{n+1} + v y_{i+1/2,j+1/2}^{n+1}}{2} \\ v y_{i+1/2,j}^{n+1} &= \frac{v y_{i+1/2,j+1/2}^{n+1} + v y_{i+1/2,j-1/2}^{n+1}}{2} \end{aligned}$$

Los Promedios de los terminos del tiempo actual son identicos a las presentadas anteriormente, solo variando en el tiempo que son calculadas. La ecuación al sustituirlos por estos promedios queda como:

$$\begin{aligned} & \frac{1}{\beta\Delta t}p_{i+1/2,j+1/2}^{n+1} + \frac{1}{4\Delta x}vx_{i+3/2,j+1/2}^{n+1} - \frac{1}{4\Delta x}vx_{i-1/2,j+1/2}^{n+1} + \frac{1}{4\Delta y}vy_{i+1/2,j+3/2}^{n+1} - \frac{1}{4\Delta y}vy_{i+1/2,j-1/2}^{n+1} \\ = & \frac{1}{\beta\Delta t}p_{i+1/2,j+1/2}^n - \frac{1}{4\Delta x}vx_{i+3/2,j+1/2}^n + \frac{1}{4\Delta x}vx_{i-1/2,j+1/2}^n - \frac{1}{4\Delta y}vy_{i+1/2,j+3/2}^n + \frac{1}{4\Delta y}vy_{i+1/2,j-1/2}^n \end{aligned}$$

A continuación, se desarrollarán las ecuaciones numéricas, de las expresiones 2.19, que nos darán la aproximación del término de confinamiento($\epsilon\vec{s}$); el cuál se agregó a las ecuaciones de *Navier-Stokes*.

Se comenzará por la matriz $\nabla\eta$:

$$\nabla\eta = \begin{pmatrix} \frac{\partial^2 V_y}{\partial x^2} - \frac{\partial^2 V_x}{\partial x\partial y} \\ \frac{\partial^2 V_y}{\partial y\partial x} - \frac{\partial^2 V_x}{\partial y^2} \end{pmatrix}$$

Debido a razones de practicidad, se renombrarán los elementos, de la matriz $\nabla\eta$; de la siguiente manera:

$$A = \frac{\partial^2 V_y}{\partial x^2}; \quad B = \frac{\partial^2 V_x}{\partial x\partial y}; \quad C = \frac{\partial^2 V_y}{\partial y\partial x}; \quad D = \frac{\partial^2 V_x}{\partial y^2}$$

Con estos cambios, la matriz $\nabla\eta$, queda de la siguiente forma:

$$\nabla\eta = \begin{pmatrix} A - B \\ C - D \end{pmatrix}$$

Aplicando en A , las aproximaciones de segundo orden, de los operadores *Divergencia* y *Gradiente*, nos queda:

$$A = \frac{\partial_y V y_{i+1,j+1/2}^n - \partial_x V y_{i,j+1/2}^n}{hx} = \frac{\frac{V y_{i+3/2,j+1/2}^n - V y_{i+1/2,j+1/2}^n}{hx} - \frac{V y_{i+1/2,j+1/2}^n - V y_{i-1/2,j+1/2}^n}{hx}}{hx}$$

Después de realizar las operaciones matemáticas pertinentes, y agrupar los términos semejante, resulta:

$$A = \frac{Vy_{i+3/2,j+1/2}^n - 2Vy_{i+1/2,j+1/2}^n + Vy_{i-1/2,j+1/2}^n}{hx^2}$$

Efectuando, el mismo procedimiento para, B , C y D , resulta:

$$B = \frac{Vx_{i+3/2,j+3/2}^n - Vx_{i-1/2,j+3/2}^n - Vx_{i+3/2,j-1/2}^n + Vx_{i-1/2,j-1/2}^n}{4hxy}$$

$$C = \frac{Vy_{i+3/2,j+3/2}^n - Vy_{i-1/2,j+3/2}^n - Vy_{i+3/2,j-1/2}^n + Vy_{i-1/2,j-1/2}^n}{4hyhx}$$

$$D = \frac{Vx_{i+1/2,j+3/2}^n - 2Vx_{i+1/2,j+1/2}^n + Vx_{i+1/2,j-1/2}^n}{hy^2}$$

Ahora, se procederá a calcular la aproximación al vector $\vec{\omega}$:

$$\vec{\omega} = \begin{pmatrix} 0 \\ 0 \\ \frac{\partial V_y}{\partial x} - \frac{\partial V_x}{\partial y} \end{pmatrix}$$

Debido a razones de practicidad, se renombrarán los elementos, del vector $\vec{\omega}$; de la siguiente manera:

$$E = \frac{\partial V_y}{\partial x}; \quad F = \frac{\partial V_x}{\partial y};$$

Con estos cambios, el vector $\vec{\omega}$, queda de la siguiente forma:

$$\vec{\omega} = \begin{pmatrix} 0 \\ 0 \\ E - F \end{pmatrix}$$

Aplicando en E y F , las aproximaciones de segundo orden, de los operadores *Divergencia* y *Gradiente*, nos queda:

$$E = \frac{Vy_{i+1,j+1/2}^n - Vy_{i,j+1/2}^n}{hx}$$

$$F = \frac{Vx_{i+1/2,j+1}^n - Vx_{i+1/2,j}^n}{hy}$$

3.1.2. Aproximaciones de la Ecuaciones de Navier-Stokes, con el término de Confinamiento de la Vorticidad y Compresibilidad Artificial en el obstáculo y la frontera

Para la Ecuación 3.1 se tiene que:

$i =$ *Hacia adelante*:

$$\begin{aligned} & \left(\frac{1}{\Delta t} + \frac{Vx_{i+1/2,j+1/2}^n}{2\Delta x} + \frac{2\nu}{(\Delta x)^2} + \frac{\nu}{(\Delta y)^2} \right) Vx_{i+1/2,j+1/2}^{n+1} + \left(\frac{Vx_{i+1/2,j+1/2}^n}{6\Delta x} - \frac{2\nu}{3(\Delta x)^2} \right) Vx_{i+3/2,j+1/2}^{n+1} \\ & + \left(\frac{Vy_{i+1/2,j+1/2}^n}{4\Delta y} - \frac{\nu}{2(\Delta y)^2} \right) Vx_{i+1/2,j+3/2}^{n+1} - \left(\frac{Vy_{i+1/2,j+1/2}^n}{4\Delta y} - \frac{\nu}{2(\Delta y)^2} \right) Vx_{i+1/2,j-1/2}^{n+1} \\ & + \frac{p_{i+1/2,j+1/2}^{n+1}}{2\rho\Delta x} + \frac{p_{i+3/2,j+1/2}^{n+1}}{6\rho\Delta x} = -\frac{1}{\rho} \left(\frac{-(8/3)p_{i,j+1/2}^n + 2p_{i+1/2,j+1/2}^n + (2/3)p_{i+3/2,j+1/2}^n}{4\Delta x} \right) \\ & + \nu \left(\frac{(8/3)Vx_{i,j+1/2}^n - 4Vx_{i+1/2,j+1/2}^n + (4/3)Vx_{i+3/2,j+1/2}^n}{2(\Delta x)^2} \right) \\ & + \frac{Vx_{i+1/2,j+3/2}^n - 2Vx_{i+1/2,j+1/2}^n + Vx_{i+1/2,j-1/2}^n}{2(\Delta y)^2} + \left(\frac{4\nu}{3(\Delta x)^2} \right) Vx_{i+j+1/2}^{n+1} + \frac{2p_{i,j+1/2}^{n+1}}{3\rho\Delta x} \\ & - Vx_{i+1/2,j+1/2}^n \left(\frac{-(8/3)Vx_{i,j+1/2}^n + 2Vx_{i+1/2,j+1/2}^n + (2/3)Vx_{i+3/2,j+1/2}^n}{4\Delta x} \right) \\ & - Vy_{i+1/2,j+1/2}^n \left(\frac{Vx_{i+1/2,j+3/2}^n - Vx_{i+1/2,j-1/2}^n}{4\Delta y} \right) + \frac{Vx_{i+1/2,j+1/2}^n}{\Delta t} + Vx_{i+1/2,j+1/2}^n \left(\frac{2Vx_{i,j+1/2}^{n+1}}{3\Delta x} \right) \end{aligned}$$

$i = \text{Hacia atrás:}$

$$\begin{aligned}
& \left(\frac{1}{\Delta t} - \frac{Vx_{i-1/2,j+1/2}^n}{2\Delta x} + \frac{2\nu}{(\Delta x)^2} + \frac{\nu}{(\Delta y)^2} \right) Vx_{i-1/2,j+1/2}^{n+1} - \left(\frac{Vx_{i-1/2,j+1/2}^n}{6\Delta x} + \frac{2\nu}{3(\Delta x)^2} \right) Vx_{i-3/2,j+1/2}^{n+1} \\
& + \left(\frac{Vy_{i-1/2,j+1/2}^n}{4\Delta y} - \frac{\nu}{2(\Delta y)^2} \right) Vx_{i-1/2,j+3/2}^{n+1} - \left(\frac{Vy_{i-1/2,j+1/2}^n}{4\Delta y} + \frac{\nu}{2(\Delta y)^2} \right) Vx_{i-1/2,j-1/2}^{n+1} \\
& - \frac{p_{i-1/2,j+1/2}^{n+1}}{2\rho\Delta x} - \frac{p_{i-3/2,j+1/2}^{n+1}}{6\rho\Delta x} = -\frac{1}{\rho} \left(\frac{(8/3)p_{i,j+1/2}^n - 2p_{i-1/2,j+1/2}^n - (2/3)p_{i-3/2,j+1/2}^n}{4\Delta x} \right) \\
& + \frac{4\nu Vx_{i,j+1/2}^{n+1}}{3(\Delta x)^2} + \nu \left(\frac{(8/3)Vx_{i,j+1/2}^n - 4Vx_{i-1/2,j+1/2}^n + (4/3)Vx_{i-3/2,j+1/2}^n}{2(\Delta x)^2} \right. \\
& \quad \left. + \frac{Vx_{i-1/2,j+3/2}^n - 2Vx_{i-1/2,j+1/2}^n + Vx_{i-1/2,j-1/2}^n}{2(\Delta y)^2} \right) + \frac{Vx_{i-1/2,j+1/2}^n}{\Delta t} \\
& - Vx_{i-1/2,j+1/2}^n \left(\frac{(8/3)Vx_{i,j+1/2}^n - 2Vx_{i-1/2,j+1/2}^n - (2/3)Vx_{i-3/2,j+1/2}^n}{4\Delta x} \right) \\
& - Vy_{i-1/2,j+1/2}^n \left(\frac{Vx_{i-1/2,j+3/2}^n - Vx_{i-1/2,j-1/2}^n}{4\Delta y} \right) - \frac{2Vy_{i-1/2,j+1/2}^n}{3\Delta x} Vx_{i,j+1/2}^{n+1} - \frac{2p_{i,j+1/2}^{n+1}}{3\rho\Delta x}
\end{aligned}$$

$j = \text{Hacia adelante:}$

$$\begin{aligned}
& \left(\frac{1}{\Delta t} + \frac{Vy_{i+1/2,j+1/2}^n}{2\Delta y} + \frac{\nu}{(\Delta x)^2} + \frac{2\nu}{(\Delta y)^2} \right) Vx_{i+1/2,j+1/2}^{n+1} + \left(\frac{Vx_{i+1/2,j+1/2}^n}{4\Delta x} - \frac{\nu}{2(\Delta x)^2} \right) Vx_{i+3/2,j+1/2}^{n+1} \\
& - \left(\frac{Vx_{i+1/2,j+1/2}^n}{4\Delta x} + \frac{\nu}{2(\Delta x)^2} \right) Vx_{i-1/2,j+1/2}^{n+1} + \left(\frac{Vy_{i+1/2,j+1/2}^n}{6\Delta y} - \frac{2\nu}{3(\Delta y)^2} \right) Vx_{i+1/2,j+3/2}^{n+1} + \frac{p_{i+3/2,j+1/2}^{n+1}}{4\rho\Delta x} \\
& - \frac{p_{i-1/2,j+1/2}^{n+1}}{4\rho\Delta x} = -\frac{1}{\rho} \left(\frac{p_{i+3/2,j+1/2}^n - p_{i-1/2,j+1/2}^n}{4\Delta x} \right) + \nu \left(\frac{Vx_{i+3/2,j+1/2}^n - 2Vx_{i+1/2,j+1/2}^n + Vx_{i-1/2,j+1/2}^n}{2(\Delta x)^2} \right. \\
& \quad \left. + \frac{(8/3)Vx_{i+1/2,j}^n - 4Vx_{i+1/2,j+1/2}^n + (4/3)Vx_{i+1/2,j+3/2}^n}{2(\Delta y)^2} \right) + \frac{Vx_{i+1/2,j+1/2}^n}{\Delta t} \\
& - Vy_{i+1/2,j+1/2}^n \left(\frac{-(8/3)Vx_{i+1/2,j}^n + 2Vx_{i+1/2,j+1/2}^n + (2/3)Vx_{i+1/2,j+3/2}^n}{4\Delta y} \right) \\
& - Vx_{i+1/2,j+1/2}^n \left(\frac{Vx_{i+3/2,j+1/2}^n - Vx_{i-1/2,j+1/2}^n}{4\Delta x} \right) + \left(\frac{2Vy_{i+1/2,j+1/2}^n}{3\Delta y} \right) Vx_{i+1/2,j}^{n+1} + \left(\frac{4\nu}{3(\Delta y)^2} \right) Vx_{i+1/2,j}^{n+1}
\end{aligned}$$

$j = \text{Hacia atrás:}$

$$\begin{aligned}
& \left(\frac{1}{\Delta t} - \frac{Vy_{i+1/2,j-1/2}^n}{2\Delta y} + \frac{2\nu}{(\Delta y)^2} + \frac{\nu}{(\Delta x)^2} \right) Vx_{i+1/2,j-1/2}^{n+1} + \left(\frac{Vx_{i+1/2,j-1/2}^n}{4\Delta x} - \frac{\nu}{2(\Delta x)^2} \right) Vx_{i+3/2,j-1/2}^{n+1} \\
& - \left(\frac{Vx_{i+1/2,j-1/2}^n}{4\Delta x} + \frac{\nu}{2(\Delta x)^2} \right) Vx_{i-1/2,j-1/2}^{n+1} - \left(\frac{Vy_{i+1/2,j-1/2}^n}{6\Delta y} + \frac{2\nu}{3(\Delta y)^2} \right) Vx_{i+1/2,j-3/2}^{n+1} \\
& + \frac{p_{i+3/2,j-1/2}^{n+1}}{4\rho\Delta x} - \frac{p_{i-1/2,j-1/2}^{n+1}}{4\rho\Delta x} = -\frac{1}{\rho} \left(\frac{p_{i+3/2,j-1/2}^n - p_{i-1/2,j-1/2}^n}{4\Delta x} \right) \\
& + \nu \left(\frac{Vx_{i+3/2,j-1/2}^n - 2Vx_{i+1/2,j-1/2}^n + Vx_{i-1/2,j-1/2}^n}{2(\Delta x)^2} \right) \\
& + \frac{(8/3)Vx_{i+1/2,j}^n - 4Vx_{i+1/2,j-1/2}^n + (4/3)Vx_{i+1/2,j-3/2}^n}{2(\Delta y)^2} \\
& - \frac{2Vy_{i+1/2,j-1/2}^n}{3\Delta y} Vx_{i+1/2,j}^{n+1} - \frac{4\nu Vx_{i+1/2,j}^{n+1}}{3(\Delta y)^2} - Vx_{i+1/2,j-1/2}^n \left(\frac{Vx_{i+3/2,j-1/2}^n - Vx_{i-1/2,j-1/2}^n}{4\Delta x} \right) \\
& - Vy_{i+1/2,j-1/2}^n \left(\frac{(8/3)Vx_{i+1/2,j}^n - 2Vx_{i+1/2,j-1/2}^n - (2/3)Vx_{i+1/2,j-3/2}^n}{4\Delta y} \right) + \frac{Vx_{i+1/2,j-1/2}^n}{\Delta t}
\end{aligned}$$

$i, j = \text{Hacia adelante:}$

$$\begin{aligned}
& \left(\frac{1}{\Delta t} + \frac{Vx_{i+1/2,j+1/2}^n}{2\Delta x} + \frac{Vy_{i+1/2,j+1/2}^n}{2\Delta x} + \frac{2\nu}{(\Delta x)^2} + \frac{2\nu}{(\Delta y)^2} \right) Vx_{i+1/2,j+1/2}^{n+1} \\
& + \left(\frac{Vx_{i+1/2,j+1/2}^n}{6\Delta x} - \frac{2\nu}{3(\Delta x)^2} \right) Vx_{i+3/2,j+1/2}^{n+1} + \left(\frac{Vy_{i+1/2,j+1/2}^n}{6\Delta x} - \frac{2\nu}{3(\Delta y)^2} \right) Vx_{i+1/2,j+3/2}^{n+1} \\
& + \frac{p_{i+1/2,j+1/2}^{n+1}}{2\rho\Delta x} + \frac{p_{i+3/2,j+1/2}^{n+1}}{6\rho\Delta x} = -\frac{1}{\rho} \left(\frac{-(8/3)p_{i,j+1/2}^n + 2p_{i+1/2,j+1/2}^n + (2/3)p_{i+3/2,j+1/2}^n}{4\Delta x} \right) \\
& + \nu \left(\frac{(8/3)Vx_{i,j+1/2}^n - 4Vx_{i+1/2,j+1/2}^n + (4/3)Vx_{i+3/2,j+1/2}^n}{2(\Delta x)^2} \right) \\
& + \frac{(8/3)Vx_{i+1/2,j}^n - 4Vx_{i+1/2,j+1/2}^n + (4/3)Vx_{i+1/2,j+3/2}^n}{2(\Delta y)^2} - \\
& \frac{-2Vx_{i+1/2,j+1/2}^n}{3\Delta x} Vx_{i,j+1/2}^{n+1} - \frac{-2Vy_{i+1/2,j+1/2}^n}{3\Delta x} Vx_{i+1/2,j}^{n+1} \\
& - \frac{-2p_{i,j+1/2}^{n+1}}{3\rho\Delta x} + \frac{4\nu Vx_{i,j+1/2}^{n+1}}{3(\Delta x)^2} \\
& - Vx_{i+1/2,j+1/2}^n \left(\frac{-(8/3)Vx_{i,j+1/2}^n + 2Vx_{i+1/2,j+1/2}^n + (2/3)Vx_{i+3/2,j+1/2}^n}{4\Delta x} \right) + \frac{4\nu Vx_{i+1/2,j}^{n+1}}{3(\Delta y)^2} \\
& - Vy_{i+1/2,j+1/2}^n \left(\frac{-(8/3)Vx_{i+1/2,j}^n + 2Vx_{i+1/2,j+1/2}^n + (2/3)Vx_{i+1/2,j+3/2}^n}{4\Delta x} \right) + \frac{Vx_{i+1/2,j+1/2}^n}{\Delta t}
\end{aligned}$$

$i = \text{Hacia adelante}; j = \text{Hacia atrás}:$

$$\begin{aligned}
& \left(\frac{1}{\Delta t} + \frac{Vx_{i+1/2,j-1/2}^n}{2\Delta x} - \frac{Vy_{i+1/2,j-1/2}^n}{2\Delta y} + \frac{2\nu}{(\Delta x)^2} + \frac{2\nu}{(\Delta y)^2} \right) Vx_{i+1/2,j-1/2}^{n+1} \\
& + \left(\frac{Vx_{i+1/2,j-1/2}^n}{6\Delta x} - \frac{2\nu}{3(\Delta x)^2} \right) Vx_{i+3/2,j-1/2}^{n+1} - \left(\frac{Vy_{i+1/2,j-1/2}^n}{6\Delta y} + \frac{2\nu}{3(\Delta y)^2} \right) Vx_{i+1/2,j-3/2}^{n+1} + \frac{p_{i+1/2,j-1/2}^{n+1}}{2\rho\Delta x} \\
& + \frac{p_{i+3/2,j-1/2}^{n+1}}{6\rho\Delta x} = -\frac{1}{\rho} \left(\frac{-(8/3)p_{i,j-1/2}^n + 2p_{i+1/2,j-1/2}^n + (2/3)p_{i+3/2,j-1/2}^n}{4\Delta x} \right) \\
& + \nu \left(\frac{(8/3)Vx_{i,j-1/2}^n - 4Vx_{i+1/2,j-1/2}^n + (4/3)Vx_{i+3/2,j-1/2}^n}{2(\Delta x)^2} \right) \\
& + \left(\frac{(8/3)Vx_{i+1/2,j-M}^n - 4Vx_{i+1/2,j-1/2}^n + (4/3)Vx_{i+1/2,j-3/2}^n}{2(\Delta y)^2} \right) \\
& - Vx_{i+1/2,j-1/2}^n \left(\frac{-(8/3)Vx_{i,j-1/2}^n + 2Vx_{i+1/2,j-1/2}^n + (2/3)Vx_{i+3/2,j-1/2}^n}{4\Delta x} \right) \\
& - Vy_{i+1/2,j-1/2}^n \left(\frac{(8/3)Vx_{i+1/2,j}^n - 2Vx_{i+1/2,j-1/2}^n - (2/3)Vx_{i+1/2,j-3/2}^n}{4\Delta y} \right) \\
& + \frac{Vx_{i+1/2,j-1/2}^n}{\Delta t} + \frac{2Vx_{i+1/2,j-1/2}^n}{3\Delta x} Vx_{i,j-1/2}^{n+1} - \frac{2Vy_{i+1/2,j-1/2}^n}{3\Delta y} Vy_{i+1/2,j}^{n+1} + \frac{2}{3\rho\Delta x} p_{i,j-1/2}^{n+1} \\
& + \frac{4\nu Vx_{i,j-1/2}^{n+1}}{3(\Delta x)^2} + \frac{4\nu Vy_{i+1/2,j}^{n+1}}{3(\Delta y)^2}
\end{aligned}$$

$i = \text{Hacia atrás}; j = \text{Hacia atrás}:$

$$\begin{aligned}
& \left(\frac{1}{\Delta t} - \frac{Vx_{i-1/2,j-1/2}^n}{2\Delta x} - \frac{Vy_{i-1/2,j-1/2}^n}{2\Delta y} + \frac{2\nu}{(\Delta x)^2} + \frac{2\nu}{(\Delta y)^2} \right) Vx_{i-1/2,j-1/2}^{n+1} \\
& - \left(\frac{Vx_{i-1/2,j-1/2}^n}{6\Delta x} + \frac{2\nu}{3(\Delta x)^2} \right) Vx_{i-3/2,j-1/2}^{n+1} - \left(\frac{Vy_{i-1/2,j-1/2}^n}{6\Delta y} + \frac{2\nu}{3(\Delta y)^2} \right) Vx_{i-1/2,j-3/2}^{n+1} - \\
& \frac{p_{i-1/2,j-1/2}^{n+1}}{2\rho\Delta x} - \frac{p_{i-3/2,j-1/2}^{n+1}}{6\rho\Delta x} = -\frac{1}{\rho} \left(\frac{(8/3)p_{i,j-1/2}^n - 2p_{i-1/2,j-1/2}^n - (2/3)p_{i-3/2,j-1/2}^n}{4\Delta x} \right) \\
& + \nu \left(\frac{(8/3)Vx_{i,j-1/2}^n - 4Vx_{i-1/2,j-1/2}^n + (4/3)Vx_{i-3/2,j-1/2}^n}{2(\Delta x)^2} + \right. \\
& \left. \frac{(8/3)Vx_{i-1/2,j}^n - 4Vx_{i-1/2,j-1/2}^n + (4/3)Vx_{i-1/2,j-3/2}^n}{2(\Delta y)^2} \right) \\
& - Vx_{i-1/2,j-1/2}^n \left(\frac{(8/3)Vx_{i,j-1/2}^n - 2Vx_{i-1/2,j-1/2}^n - (2/3)Vx_{i-3/2,j-1/2}^n}{4\Delta x} \right) \\
& - Vy_{i-1/2,j-1/2}^n \left(\frac{(8/3)Vx_{i-1/2,j}^n - 2Vx_{i-1/2,j-1/2}^n - (2/3)Vx_{i-1/2,j-3/2}^n}{4\Delta y} \right) + \frac{Vx_{i-1/2,j-1/2}^n}{\Delta t} \\
& - \frac{2Vx_{i-1/2,j-1/2}^n}{3\Delta x} Vx_{i,j-1/2}^{n+1} - \frac{2Vy_{i-1/2,j-1/2}^n}{3\Delta y} Vx_{i-1/2,j}^{n+1} - \frac{2}{3\rho\Delta x} p_{i,j-1/2}^{n+1} \\
& + \frac{4\nu}{3(\Delta x)^2} Vx_{i,j-1/2}^{n+1} + \frac{4\nu}{3(\Delta y)^2} Vx_{i-1/2,j}^{n+1}
\end{aligned}$$

$i = \text{Hacia atrás}; j = \text{Hacia adelante}:$

$$\begin{aligned}
& \left(\frac{1}{\Delta t} - \frac{Vx_{i-1/2,j+1/2}^n}{2\Delta x} + \frac{Vy_{i-1/2,j+1/2}^n}{2\Delta y} + \frac{2\nu}{(\Delta x)^2} + \frac{2\nu}{(\Delta y)^2} \right) Vx_{i-1/2,j+1/2}^{n+1} \\
& - \left(\frac{Vx_{i-1/2,j+1/2}^n}{6\Delta x} + \frac{2\nu}{3(\Delta x)^2} \right) Vx_{i-3/2,j+1/2}^{n+1} + \left(\frac{Vy_{i-1/2,j+1/2}^n}{6\Delta y} - \frac{2\nu}{3(\Delta y)^2} \right) Vx_{i-1/2,j+3/2}^{n+1} \\
& - \frac{p_{i-1/2,j+1/2}^{n+1}}{2\rho\Delta x} - \frac{p_{i-3/2,j+1/2}^{n+1}}{6\rho\Delta x} = -\frac{1}{\rho} \left(\frac{(8/3)p_{i,j+1/2}^n - 2p_{i-1/2,j+1/2}^n - (2/3)p_{i-3/2,j+1/2}^n}{4\Delta x} \right) \\
& + \nu \left(\frac{(8/3)Vx_{i,j+1/2}^n - 4Vx_{i-1/2,j+1/2}^n + (4/3)Vx_{i-3/2,j+1/2}^n}{2(\Delta x)^2} \right) \\
& + \frac{(8/3)Vx_{i-1/2,j}^n - 4Vx_{i-1/2,j+1/2}^n + (4/3)Vx_{i-1/2,j+3/2}^n}{2(\Delta y)^2} \\
& - Vx_{i-1/2,j+1/2}^n \frac{((8/3)Vx_{i,j-1/2}^n - 2Vx_{i-1/2,j+1/2}^n - (2/3)Vx_{i-3/2,j+1/2}^n)}{4\Delta x} \\
& - Vy_{i-1/2,j+1/2}^n \frac{(-(8/3)Vx_{i-1/2,j}^n + 2Vx_{i-1/2,j+1/2}^n + (2/3)Vx_{i-1/2,j+3/2}^n)}{4\Delta y} \\
& + \frac{Vx_{i-1/2,j+1/2}^n}{\Delta t} - \frac{2Vx_{i-1/2,j+1/2}^n}{3\Delta x} Vx_{i,j+1/2}^{n+1} + \frac{2Vy_{i-1/2,j+1/2}^n}{3\Delta y} Vx_{i-1/2,j}^{n+1} + \frac{4\nu}{3(\Delta x)^2} Vx_{i,j+1/2}^{n+1} \\
& - \frac{2p_{i,j+1/2}^{n+1}}{3\rho\Delta x} + \frac{4\nu}{3(\Delta y)^2} Vx_{i-1/2,j}^{n+1}
\end{aligned}$$

Para la Ecuación 3.2 se tiene:

$i = \text{Hacia adelante:}$

$$\begin{aligned}
& \left(\frac{1}{\Delta t} + \frac{Vx_{i+1/2,j+1/2}^n}{2\Delta x} + \frac{2\nu}{(\Delta x)^2} + \frac{\nu}{(\Delta y)^2} \right) Vy_{i+1/2,j+1/2}^{n+1} \\
& \quad + \left(\frac{Vx_{i+1/2,j+1/2}^n}{6\Delta x} - \frac{2\nu}{3(\Delta x)^2} \right) Vy_{i+3/2,j+1/2}^{n+1} \\
& + \left(\frac{Vy_{i+1/2,j+1/2}^n}{4\Delta y} - \frac{\nu}{2(\Delta y)^2} \right) Vy_{i+1/2,j+3/2}^{n+1} - \left(\frac{Vy_{i+1/2,j+1/2}^n}{4\Delta y} - \frac{\nu}{2(\Delta y)^2} \right) Vy_{i+1/2,j-1/2}^{n+1} + \frac{p_{i+1/2,j+3/2}^{n+1}}{4\rho\Delta y} \\
& \quad - \frac{p_{i+1/2,j-1/2}^{n+1}}{4\rho\Delta y} = -\frac{1}{\rho} \left(\frac{p_{i+1/2,j+3/2}^n - p_{i+1/2,j-1/2}^n}{4\Delta y} \right) \\
& \quad + \nu \left(\frac{(8/3)Vy_{i,j+1/2}^n - 4Vy_{i+1/2,j+1/2}^n + (4/3)Vy_{i+3/2,j+1/2}^n}{2(\Delta x)^2} \right. \\
& \quad \left. + \frac{Vy_{i+1/2,j+3/2}^n - 2Vy_{i+1/2,j+1/2}^n + Vy_{i+1/2,j-1/2}^n}{2(\Delta y)^2} \right) + \left(\frac{4\nu}{3(\Delta x)^2} \right) Vy_{i,j+1/2}^{n+1} \\
& \quad - Vx_{i+1/2,j+1/2}^n \left(\frac{-(8/3)Vy_{i,j+1/2}^n + 2Vy_{i+1/2,j+1/2}^n + (2/3)Vy_{i+3/2,j+1/2}^n}{4\Delta x} \right) \\
& \quad - Vy_{i+1/2,j+1/2}^n \left(\frac{Vy_{i+1/2,j+3/2}^n - Vy_{i+1/2,j-1/2}^n}{4\Delta y} \right) + \frac{Vy_{i+1/2,j+1/2}^n}{\Delta t} + Vx_{i+1/2,j+1/2}^n \left(\frac{2Vy_{i,j+1/2}^{n+1}}{3\Delta x} \right)
\end{aligned}$$

$i = \text{Hacia atrás:}$

$$\begin{aligned}
& \left(\frac{1}{\Delta t} - \frac{Vx_{i-1/2,j+1/2}^n}{2\Delta x} + \frac{2\nu}{(\Delta x)^2} + \frac{\nu}{(\Delta y)^2} \right) Vy_{i-1/2,j+1/2}^{n+1} - \left(\frac{Vx_{i-1/2,j+1/2}^n}{6\Delta x} + \frac{2\nu}{3(\Delta x)^2} \right) Vy_{i-3/2,j+1/2}^{n+1} \\
& + \left(\frac{Vy_{i-1/2,j+1/2}^n}{4\Delta y} - \frac{\nu}{2(\Delta y)^2} \right) Vy_{i-1/2,j+3/2}^{n+1} - \left(\frac{Vy_{i-1/2,j+1/2}^n}{4\Delta y} + \frac{\nu}{2(\Delta y)^2} \right) Vy_{i-1/2,j-1/2}^{n+1} \\
& + \frac{p_{i-1/2,j+3/2}^{n+1}}{4\rho\Delta y} - \frac{p_{i-1/2,j-1/2}^{n+1}}{4\rho\Delta y} = -\frac{1}{\rho} \left(\frac{p_{i-1/2,j+3/2}^n - p_{i-1/2,j-1/2}^n}{4\Delta y} \right) \\
& + \nu \left(\frac{(8/3)Vy_{i,j+1/2}^n - 4Vy_{i-1/2,j+1/2}^n + (4/3)Vy_{i-3/2,j+1/2}^n}{2(\Delta x)^2} \right. \\
& \left. + \frac{Vy_{i-1/2,j+3/2}^n - 2Vy_{i-1/2,j+1/2}^n + Vy_{i-1/2,j-1/2}^n}{2(\Delta y)^2} \right) + \frac{Vy_{i-1/2,j+1/2}^n}{\Delta t} \\
& - Vx_{i-1/2,j+1/2}^n \left(\frac{(8/3)Vy_{i,j+1/2}^n - 2Vy_{i-1/2,j+1/2}^n - (2/3)Vy_{i-3/2,j+1/2}^n}{4\Delta x} \right) \\
& - Vy_{i-1/2,j+1/2}^n \left(\frac{Vy_{i-1/2,j+3/2}^n - Vy_{i-1/2,j-1/2}^n}{4\Delta y} \right) - \frac{2Vx_{i-1/2,j+1/2}^n Vy_{i,j+1/2}^{n+1}}{3\Delta x} - \frac{2p_{i,j+1/2}^{n+1}}{3\rho\Delta y} \\
& + \frac{4\nu Vy_{i,j+1/2}^{n+1}}{3(\Delta x)^2}
\end{aligned}$$

$j = \text{Hacia adelante:}$

$$\begin{aligned}
& \left(\frac{1}{\Delta t} + \frac{Vy_{i+1/2,j+1/2}^n}{2\Delta y} + \frac{\nu}{(\Delta x)^2} + \frac{2\nu}{(\Delta y)^2} \right) Vy_{i+1/2,j+1/2}^{n+1} + \left(\frac{Vx_{i+1/2,j+1/2}^n}{4\Delta x} - \frac{\nu}{2(\Delta x)^2} \right) Vy_{i+3/2,j+1/2}^{n+1} \\
& - \left(\frac{Vx_{i+1/2,j+1/2}^n}{4\Delta x} + \frac{\nu}{2(\Delta x)^2} \right) Vy_{i-1/2,j+1/2}^{n+1} + \left(\frac{Vy_{i+1/2,j+1/2}^n}{6\Delta y} - \frac{2\nu}{3(\Delta y)^2} \right) Vy_{i+1/2,j+3/2}^{n+1} + \frac{p_{i+1/2,j+1/2}^{n+1}}{2\rho\Delta y} \\
& + \frac{p_{i+1/2,j+3/2}^{n+1}}{6\rho\Delta y} = -\frac{1}{\rho} \left(\frac{-(8/3)p_{i+1/2,j}^n + 2p_{i+1/2,j+1/2}^n + (2/3)p_{i+1/2,j+3/2}^n}{4\Delta y} \right) \\
& + \nu \left(\frac{Vy_{i+3/2,j+1/2}^n - 2Vy_{i+1/2,j+1/2}^n + Vy_{i-1/2,j+1/2}^n}{2(\Delta x)^2} \right. \\
& \left. + \frac{(8/3)Vy_{i+1/2,j}^n - 4Vy_{i+1/2,j+1/2}^n + (4/3)Vy_{i+1/2,j+3/2}^n}{2(\Delta y)^2} \right) + \frac{Vy_{i+1/2,j+1/2}^n}{\Delta t} \\
& + \frac{2p_{i+1/2,j}^{n+1}}{3\rho\Delta y} - Vy_{i+1/2,j+1/2}^n \left(\frac{-(8/3)Vy_{i+1/2,j}^n + 2Vy_{i+1/2,j+1/2}^n + (2/3)Vy_{i+1/2,j+3/2}^n}{4\Delta y} \right) \\
& - Vx_{i+1/2,j+1/2}^n \left(\frac{Vy_{i+3/2,j+1/2}^n - Vy_{i-1/2,j+1/2}^n}{4\Delta x} \right) + \left(\frac{2Vy_{i+1/2,j+1/2}^n}{3\Delta y} \right) Vy_{i+1/2,j}^{n+1} + \left(\frac{4\nu}{3(\Delta y)^2} \right) Vy_{i+1/2,j}^{n+1}
\end{aligned}$$

$j = \text{Hacia atrás:}$

$$\begin{aligned}
& \left(\frac{1}{\Delta t} - \frac{Vy_{i+1/2,j-1/2}^n}{2\Delta y} + \frac{2\nu}{(\Delta y)^2} + \frac{\nu}{(\Delta x)^2} \right) Vy_{i+1/2,j-1/2}^{n+1} + \left(\frac{Vx_{i+1/2,j-1/2}^n}{4\Delta x} - \frac{\nu}{2(\Delta x)^2} \right) Vy_{i+3/2,j-1/2}^{n+1} \\
& - \left(\frac{Vx_{i+1/2,j-1/2}^n}{4\Delta x} + \frac{\nu}{2(\Delta x)^2} \right) Vy_{i-1/2,j-1/2}^{n+1} - \left(\frac{Vy_{i+1/2,j-1/2}^n}{6\Delta y} + \frac{2\nu}{3(\Delta y)^2} \right) Vy_{i+1/2,j-3/2}^{n+1} \\
& - \frac{p_{i+1/2,j-1/2}^{n+1}}{2\rho\Delta y} - \frac{p_{i+1/2,j-3/2}^{n+1}}{6\rho\Delta y} = -\frac{1}{\rho} \left(\frac{(8/3)p_{i+1/2,j}^n - 2p_{i+1/2,j-1/2}^n - (2/3)p_{i+1/2,j-3/2}^n}{4\Delta y} \right) \\
& - \frac{2Vy_{i+1/2,j-1/2}^n}{3\Delta y} Vy_{i+1/2,j}^{n+1} + \nu \left(\frac{Vy_{i+3/2,j-1/2}^n - 2Vy_{i+1/2,j-1/2}^n + Vy_{i-1/2,j-1/2}^n}{2(\Delta x)^2} \right. \\
& \quad \left. + \frac{(8/3)Vy_{i+1/2,j}^n - 4Vy_{i+1/2,j-1/2}^n + (4/3)Vy_{i+1/2,j-3/2}^n}{2(\Delta y)^2} \right) \\
& - \frac{4\nu Vy_{i+1/2,j}^{n+1}}{3(\Delta y)^2} - Vx_{i+1/2,j-1/2}^n \left(\frac{Vy_{i+3/2,j-1/2}^n - Vy_{i-1/2,j-1/2}^n}{4\Delta x} \right) \\
& - Vy_{i+1/2,j-1/2}^n \left(\frac{(8/3)Vy_{i+1/2,j}^n - 2Vy_{i+1/2,j-1/2}^n - (2/3)Vy_{i+1/2,j-3/2}^n}{4\Delta y} \right) + \frac{Vy_{i+1/2,j-1/2}^n}{\Delta t}
\end{aligned}$$

$i = \text{Hacia adelante; } j = \text{Hacia adelante:}$

$$\begin{aligned}
& \left(\frac{1}{\Delta t} + \frac{Vx_{i+1/2,j+1/2}^n}{2\Delta x} + \frac{Vy_{i+1/2,j+1/2}^n}{2\Delta x} + \frac{2\nu}{(\Delta x)^2} + \frac{2\nu}{(\Delta y)^2} \right) Vy_{i+1/2,j+1/2}^{n+1} \\
& + \left(\frac{Vx_{i+1/2,j+1/2}^n}{6\Delta x} - \frac{2\nu}{3(\Delta x)^2} \right) Vy_{i+3/2,j+1/2}^{n+1} + \left(\frac{Vy_{i+1/2,j+1/2}^n}{6\Delta x} - \frac{2\nu}{3(\Delta y)^2} \right) Vy_{i+1/2,j+3/2}^{n+1} \\
& + \frac{p_{i+1/2,j+1/2}^{n+1}}{2\rho\Delta y} + \frac{p_{i+1/2,j+3/2}^{n+1}}{6\rho\Delta y} = -\frac{1}{\rho} \left(\frac{-(8/3)p_{i+1/2,j}^n + 2p_{i+1/2,j+1/2}^n + (2/3)p_{i+1/2,j+3/2}^n}{4\Delta y} \right) \\
& + \nu \left(\frac{(8/3)Vy_{i,j+1/2}^n - 4Vy_{i+1/2,j+1/2}^n + (4/3)Vy_{i+3/2,j+1/2}^n}{2(\Delta x)^2} \right. \\
& \quad \left. + \frac{(8/3)Vy_{i+1/2,j}^n - 4Vy_{i+1/2,j+1/2}^n + (4/3)Vy_{i+1/2,j+3/2}^n}{2(\Delta y)^2} \right) \\
& + \frac{2Vx_{i+1/2,j+1/2}^n}{3\Delta x} Vy_{i,j+1/2}^{n+1} + \frac{2Vy_{i+1/2,j+1/2}^n}{3\Delta x} Vy_{i+1/2,j}^{n+1} + \frac{2p_{i+1/2,j}^{n+1}}{3\rho\Delta y} + \frac{4\nu Vy_{i,j+1/2}^{n+1}}{3(\Delta x)^2} \\
& - Vx_{i+1/2,j+1/2}^n \left(\frac{-(8/3)Vy_{i,j+1/2}^n + 2Vy_{i+1/2,j+1/2}^n + (2/3)Vy_{i+3/2,j+1/2}^n}{4\Delta x} \right) + \frac{4\nu Vy_{i+1/2,j}^{n+1}}{3(\Delta y)^2} \\
& - Vy_{i+1/2,j+1/2}^n \left(\frac{-(8/3)Vy_{i+1/2,j}^n + 2Vy_{i+1/2,j+1/2}^n + (2/3)Vy_{i+1/2,j+3/2}^n}{4\Delta x} \right) + \frac{Vy_{i+1/2,j+1/2}^n}{\Delta t}
\end{aligned}$$

$i = \text{Hacia adelante}; j = \text{Hacia atrás}:$

$$\begin{aligned}
& \left(\frac{1}{\Delta t} + \frac{Vx_{i+1/2,j-1/2}^n}{2\Delta x} - \frac{Vy_{i+1/2,j-1/2}^n}{2\Delta y} + \frac{2\nu}{(\Delta x)^2} + \frac{2\nu}{(\Delta y)^2} \right) Vy_{i+1/2,j-1/2}^{n+1} \\
& + \left(\frac{Vx_{i+1/2,j-1/2}^n}{6\Delta x} - \frac{2\nu}{3(\Delta x)^2} \right) Vy_{i+3/2,j-1/2}^{n+1} - \left(\frac{Vy_{i+1/2,j-1/2}^n}{6\Delta y} + \frac{2\nu}{3(\Delta y)^2} \right) Vy_{i+1/2,j-3/2}^{n+1} - \frac{p_{i+1/2,j-1/2}^{n+1}}{2\rho\Delta y} \\
& - \frac{p_{i+1/2,j-3/2}^{n+1}}{6\rho\Delta y} = -\frac{1}{\rho} \left(\frac{(8/3)p_{i+1/2,j}^n - 2p_{i+1/2,j-1/2}^n - (2/3)p_{i+1/2,j-3/2}^n}{4\Delta y} \right) \\
& + \nu \left(\frac{(8/3)Vy_{i,j-1/2}^n - 4Vy_{i+1/2,j-1/2}^n + (4/3)Vy_{i+3/2,j-1/2}^n}{2(\Delta x)^2} \right. \\
& \quad \left. + \frac{(8/3)Vy_{i+1/2,j}^n - 4Vy_{i+1/2,j-1/2}^n + (4/3)Vy_{i+1/2,j-3/2}^n}{2(\Delta y)^2} \right) \\
& - Vx_{i+1/2,j-1/2}^n \left(\frac{-(8/3)Vy_{i,j-1/2}^n + 2Vy_{i+1/2,j-1/2}^n + (2/3)Vy_{i+3/2,j-1/2}^n}{4\Delta x} \right) \\
& - Vy_{i+1/2,j-1/2}^n \left(\frac{(8/3)Vy_{i+1/2,j}^n - 2Vy_{i+1/2,j-1/2}^n - (2/3)Vy_{i+1/2,j-3/2}^n}{4\Delta y} \right) \\
& + \frac{Vy_{i+1/2,j-1/2}^n}{\Delta t} + \frac{2Vx_{i+1/2,j-1/2}^n}{3\Delta x} Vy_{i,j-1/2}^{n+1} - \frac{2Vy_{i+1/2,j-1/2}^n}{3\Delta y} Vy_{i+1/2,j}^{n+1} + \frac{2}{3\rho\Delta y} p_{i+1/2,j}^{n+1} \\
& \quad + \frac{4\nu Vy_{i,j-1/2}^{n+1}}{3(\Delta x)^2} + \frac{4\nu Vy_{i+1/2,j}^{n+1}}{3(\Delta y)^2}
\end{aligned}$$

$i = \text{Hacia atrás}; j = \text{Hacia atrás}:$

$$\begin{aligned}
& \left(\frac{1}{\Delta t} - \frac{Vx_{i-1/2,j-1/2}^n}{2\Delta x} - \frac{Vy_{i-1/2,j-1/2}^n}{2\Delta y} + \frac{2\nu}{(\Delta x)^2} + \frac{2\nu}{(\Delta y)^2} \right) Vy_{i-1/2,j-1/2}^{n+1} \\
& - \left(\frac{Vx_{i-1/2,j-1/2}^n}{6\Delta x} + \frac{2\nu}{3(\Delta x)^2} \right) Vy_{i-3/2,j-1/2}^{n+1} - \left(\frac{Vy_{i-1/2,j-1/2}^n}{6\Delta y} + \frac{2\nu}{3(\Delta y)^2} \right) Vy_{i-1/2,j-3/2}^{n+1} - \\
& \frac{p_{i-1/2,j-1/2}^{n+1}}{2\rho\Delta y} - \frac{p_{i-1/2,j-3/2}^{n+1}}{6\rho\Delta y} = -\frac{1}{\rho} \left(\frac{(8/3)p_{i-1/2,j}^n - 2p_{i-1/2,j-1/2}^n - (2/3)p_{i-1/2,j-3/2}^n}{4\Delta y} \right) \\
& + \nu \left(\frac{(8/3)Vy_{i,j-1/2}^n - 4Vy_{i-1/2,j-1/2}^n + (4/3)Vy_{i-3/2,j-1/2}^n}{2(\Delta x)^2} + \right. \\
& \quad \left. \frac{(8/3)Vy_{i-1/2,j}^n - 4Vy_{i-1/2,j-1/2}^n + (4/3)Vy_{i-1/2,j-3/2}^n}{2(\Delta y)^2} \right) \\
& - Vx_{i-1/2,j-1/2}^n \left(\frac{(8/3)Vy_{i,j-1/2}^n - 2Vy_{i-1/2,j-1/2}^n - (2/3)Vy_{i-3/2,j-1/2}^n}{4\Delta x} \right) \\
& - Vy_{i-1/2,j-1/2}^n \left(\frac{(8/3)Vy_{i-1/2,j}^n - 2Vy_{i-1/2,j-1/2}^n - (2/3)Vy_{i-1/2,j-3/2}^n}{4\Delta y} \right) + \frac{Vy_{i-1/2,j-1/2}^n}{\Delta t} \\
& - \frac{2Vx_{i-1/2,j-1/2}^n}{3\Delta x} Vy_{i,j-1/2}^{n+1} - \frac{2Vy_{i-1/2,j-1/2}^n}{3\Delta y} Vy_{i-1/2,j}^{n+1} - \frac{2}{3\rho\Delta y} p_{i-1/2,j}^{n+1} \\
& \quad + \frac{4\nu}{3(\Delta x)^2} Vy_{i,j-1/2}^{n+1} + \frac{4\nu}{3(\Delta y)^2} Vy_{i-1/2,j}^{n+1}
\end{aligned}$$

$i = \text{Hacia atrás}; j = \text{Hacia adelante}:$

$$\begin{aligned}
& \left(\frac{1}{\Delta t} - \frac{Vx_{i-1/2,j+1/2}^n}{2\Delta x} + \frac{Vy_{i-1/2,j+1/2}^n}{2\Delta y} + \frac{2\nu}{(\Delta x)^2} + \frac{2\nu}{(\Delta y)^2} \right) Vy_{i-1/2,j+1/2}^{n+1} \\
& - \left(\frac{Vx_{i-1/2,j+1/2}^n}{6\Delta x} + \frac{2\nu}{3(\Delta x)^2} \right) Vy_{i-3/2,j+1/2}^{n+1} + \left(\frac{Vy_{i-1/2,j+1/2}^n}{6\Delta y} - \frac{2\nu}{3(\Delta y)^2} \right) Vy_{i-1/2,j+3/2}^{n+1} \\
& + \frac{p_{i-1/2,j+1/2}^{n+1}}{2\rho\Delta y} + \frac{p_{i-1/2,j+3/2}^{n+1}}{6\rho\Delta y} = -\frac{1}{\rho} \left(\frac{-(8/3)p_{i-1/2,j}^n + 2p_{i-1/2,j+1/2}^n + (2/3)p_{i-1/2,j+3/2}^n}{4\Delta y} \right) \\
& + \nu \left(\frac{(8/3)Vy_{i,j+1/2}^n - 4Vy_{i-1/2,j+1/2}^n + (4/3)Vy_{i-3/2,j+1/2}^n}{2(\Delta x)^2} \right) \\
& + \frac{(8/3)Vy_{i-1/2,j}^n - 4Vy_{i-1/2,j+1/2}^n + (4/3)Vy_{i-1/2,j+3/2}^n}{2(\Delta y)^2} \\
& - Vx_{i-1/2,j+1/2}^n \frac{((8/3)Vy_{i,j+1/2}^n - 2Vy_{i-1/2,j+1/2}^n - (2/3)Vy_{i-3/2,j+1/2}^n)}{4\Delta x} \\
& - Vy_{i-1/2,j+1/2}^n \frac{(-(8/3)Vy_{i-1/2,j}^n + 2Vy_{i-1/2,j+1/2}^n + (2/3)Vy_{i-1/2,j+3/2}^n)}{4\Delta y} \\
& + \frac{Vy_{i-1/2,j+1/2}^n}{\Delta t} - \frac{2Vx_{i-1/2,j+1/2}^n}{3\Delta x} Vy_{i,j+1/2}^{n+1} + \frac{2Vy_{i-1/2,j+1/2}^n}{3\Delta y} Vy_{i-1/2,j}^{n+1} + \frac{4\nu}{3(\Delta x)^2} Vy_{i,j+1/2}^{n+1} \\
& - \frac{2p_{i,j+1/2}^{n+1}}{3\rho\Delta y} + \frac{4\nu}{3(\Delta y)^2} Vy_{i-1/2,j}^{n+1}
\end{aligned}$$

Aproximaciones en la Frontera, del término de Confinamiento. Aquí, se utilizarán los operadores de segundo orden, en la frontera:

$i = \text{Hacia adelante}:$

$$\begin{aligned}
A &= \frac{\partial_x Vy_{1,j+1/2}^n - \partial_x Vy_{0,j+1/2}^n}{hx} \\
&= \frac{\frac{Vy_{i+3/2,j+1/2}^n - Vy_{i+1/2,j+1/2}^n}{hx} - \frac{-4/3Vy_{i,j+1/2}^n + 3/2Vy_{i+1/2,j+1/2}^n - 1/6Vy_{i+3/2,j+1/2}^n}{hx/2}}{hx}
\end{aligned}$$

Realizando, operaciones elementales y agrupación de términos semejantes, nos queda:

$$A = \frac{4/3Vy_{i+3/2,j+1/2}^n - 4Vy_{i+1/2,j+1/2}^n + 8/3Vy_{i,j+1/2}^n}{hx^2}$$

Procediendo, de la misma manera para B , C y D , resulta:

$$B = \frac{Vx_{i+3/2,j+3/2}^n + Vx_{i+1/2,j+3/2}^n - Vx_{i+3/2,j-1/2}^n - Vx_{i+1/2,j-1/2}^n - 2Vx_{i,j+1/2}^n + 2Vx_{i,j-1/2}^n}{4hxy}$$

$$C = \frac{Vy_{i+3/2,j+3/2}^n + Vy_{i+1/2,j+3/2}^n - Vy_{i+3/2,j-1/2}^n - Vy_{i+1/2,j-1/2}^n - 2Vy_{i,j+1/2}^n + 2Vy_{i,j-1/2}^n}{4hxy}$$

$$D = \frac{Vx_{i+1/2,j+3/2}^n - 2Vx_{i+1/2,j+1/2}^n + Vx_{i+1/2,j-1/2}^n}{hy^2}$$

Para E y F , resulta:

$$E = \frac{Vy_{i+3/2,j+1/2}^n + Vy_{i+1/2,j+1/2}^n - 2Vy_{i,j+1/2}^n}{2hx}$$

$$F = \frac{Vx_{i+1/2,j+3/2}^n - Vx_{i+1/2,j-1/2}^n}{2hy}$$

$j = \text{Hacia adelante:}$

$$A = \frac{Vy_{i+3/2,j+1/2}^n - 2Vy_{i+1/2,j+1/2}^n + Vy_{i-1/2,j+1/2}^n}{hx^2}$$

$$B = \frac{Vx_{i+3/2,j+3/2}^n + Vx_{i+3/2,j+1/2}^n - 2Vx_{i+3/2,j}^n - Vx_{i-1/2,j+3/2}^n - Vx_{i-1/2,j+1/2}^n + 2Vx_{i-1/2,j}^n}{hxy}$$

$$C = \frac{Vy_{i+3/2,j+3/2}^n + Vy_{i+3/2,j+1/2}^n - 2Vy_{i+3/2,j}^n - Vy_{i-1/2,j+3/2}^n - Vy_{i-1/2,j+1/2}^n + 2Vy_{i-1/2,j}^n}{hxy}$$

$$D = \frac{4/3Vx_{i+1/2,j+3/2}^n - 4Vx_{i+1/2,j+1/2}^n + 8/3Vx_{i+1/2,j}^n}{hy^2}$$

$$E = \frac{Vy_{i+3/2,j+1/2}^n - Vy_{i-1/2,j+1/2}^n}{2hx}$$

$$F = \frac{Vx_{i+1/2,j+3/2}^n + Vx_{i+1/2,j+1/2}^n - 2Vx_{i+1/2,j}^n}{2hy}$$

$i = \text{Hacia atrás:}$

$$A = \frac{8/3Vy_{i,j+1/2}^n - 4Vy_{i-1/2,j+1/2}^n + 4/3Vy_{i-3/2,j+1/2}^n}{hx^2}$$

$$B = \frac{2(Vx_{i,j+3/2}^n - Vx_{i,j-1/2}^n) - Vx_{i-1/2,j+3/2}^n - Vx_{i-3/2,j+3/2}^n + Vx_{i-3/2,j-1/2}^n + Vx_{i-3/2,j-1/2}^n}{4hxy}$$

$$C = \frac{2(Vy_{i,j+3/2}^n - Vy_{i,j-1/2}^n) - Vy_{i-1/2,j+3/2}^n - Vy_{i-3/2,j+3/2}^n + Vy_{i-3/2,j-1/2}^n + Vy_{i-3/2,j-1/2}^n}{4hxy}$$

$$D = \frac{Vx_{i-1/2,j+3/2}^n - 2Vx_{i-1/2,j+1/2}^n + Vx_{i-1/2,j-1/2}^n}{hy^2}$$

$$E = \frac{2Vy_{i,j+1/2}^n - Vy_{i-1/2,j+1/2}^n - Vy_{i-1/2,j+1/2}^n}{2hx}$$

$$F = \frac{Vx_{i,j+3/2}^n - Vx_{i,j-1/2}^n}{hy}$$

$j = \text{Hacia atrás:}$

$$A = \frac{Vy_{i+3/2,j-1/2}^n - 2Vy_{i+1/2,j-1/2}^n + Vy_{i-1/2,j-1/2}^n}{hx^2}$$

$$B = \frac{2(Vx_{i+3/2,j}^n - Vx_{i-1/2,j}^n) - Vx_{i+3/2,j-1/2}^n - Vx_{i+3/2,j-3/2}^n + Vx_{i-1/2,j-3/2}^n + Vx_{i-1/2,j-3/2}^n}{4hxy}$$

$$C = \frac{2(Vy_{i+3/2,j}^n - Vy_{i-1/2,j}^n) - Vy_{i+3/2,j-1/2}^n - Vy_{i+3/2,j-3/2}^n + Vy_{i-1/2,j-3/2}^n + Vy_{i-1/2,j-3/2}^n}{4hxy}$$

$$D = \frac{8/3Vx_{i+1/2,j}^n - 4Vx_{i+1/2,j-1/2}^n + 4/3Vx_{i+1/2,j-3/2}^n}{hy^2}$$

$$E = \frac{Vy_{i+3/2,j}^n - Vy_{i-1/2,j}^n}{2hx}$$

$$F = \frac{2Vx_{i+1/2,j}^n - Vx_{i+1/2,j}^n - Vx_{i+1/2,j-3/2}^n}{2hy}$$

$i, j = \text{Hacia adelante:}$

$$\begin{aligned}
 A &= \frac{4/3Vy_{i+3/2,j+1/2}^n - 4Vy_{i+1/2,j+1/2}^n + 8/3Vy_{i,j+1/2}^n}{hx^2} \\
 B &= \frac{Vx_{i+3/2,j+3/2}^n + Vx_{i+1/2,j+3/2}^n + Vx_{i+1/2,j+1/2}^n + Vx_{i+3/2,j+1/2}^n - 4(Vx_{i+1,j}^n + Vx_{i,j+1}^n - Vx_{i,j}^n)}{4hxy} \\
 C &= \frac{Vy_{i+3/2,i+3/2}^n + Vy_{i+1/2,i+3/2}^n + Vy_{i+1/2,i+1/2}^n + Vy_{i+3/2,i+1/2}^n - 4(Vy_{i+1,j}^n + Vy_{i,j+1}^n - Vy_{i,j}^n)}{4hxy} \\
 D &= \frac{4/3Vx_{i+1/2,j+3/2}^n - 4Vx_{i+1/2,j+1/2}^n + 8/3Vx_{i+1/2,j}^n}{hy^2}
 \end{aligned}$$

$$\begin{aligned}
 E &= \frac{2Vy_{i+3/2,j+1/2}^n + Vy_{i+1/2,j+1/2}^n - 2Vy_{i,j+1/2}^n}{2hx} \\
 F &= \frac{2Vx_{i+1/2,j+3/2}^n - Vx_{i+1/2,j+1/2}^n - 2Vx_{i+1/2,j}^n}{2hy}
 \end{aligned}$$

$i, j = \text{Hacia atrás:}$

$$\begin{aligned}
 A &= \frac{8/3Vy_{i,j-1/2}^n - 4Vy_{i-1/2,j-1/2}^n + 4/3Vy_{i-3/2,j-1/2}^n}{hx^2} \\
 B &= \frac{4(Vx_{i,j}^n - Vx_{i,j-1}^n - Vx_{i-1,j}^n) + Vx_{i-1/2,j-1/2}^n + Vx_{i-3/2,j-1/2}^n + Vx_{i-3/2,j-3/2}^n + Vx_{i-1/2,i-3/2}^n}{4hxy} \\
 C &= \frac{4(Vy_{i,j}^n - Vy_{i,j-1}^n - Vy_{i-1,j}^n) + Vy_{i-1/2,j-1/2}^n + Vy_{i-3/2,j-1/2}^n + Vy_{i-3/2,j-3/2}^n + Vy_{i-1/2,j-3/2}^n}{4hxy} \\
 D &= \frac{8/3Vx_{i-1/2,j}^n - 4Vx_{i-1/2,j-1/2}^n + 4/3Vx_{i-1/2,j-3/2}^n}{hy^2}
 \end{aligned}$$

$$\begin{aligned}
 E &= \frac{2Vy_{i,j}^n - Vy_{i-1/2,j}^n - Vy_{i-3/2,j}^n}{2hx} \\
 F &= \frac{2Vx_{i,j}^n - Vx_{i,j-1/2}^n - 2Vx_{i,j-3/2}^n}{2hy}
 \end{aligned}$$

$i = \text{Hacia adelante}, j = \text{Hacia atrás}:$

$$A = \frac{4/3Vy_{i+3/2,j-1/2}^n - 4Vy_{i+1/2,j-1/2}^n + 8/3Vy_{i,j-1/2}^n}{hx^2}$$

$$B = \frac{4(Vx_{i+1,j}^n - Vx_{i,j}^n + Vx_{i,j-1}^n) - Vx_{i+3/2,j-1/2}^n - Vx_{i+1/2,j-3/2}^n - Vx_{i+3/2,j-3/2}^n}{4hxy}$$

$$C = \frac{4(Vy_{i+1,j}^n - Vy_{i,j}^n + Vy_{i,j-1}^n) - Vy_{i+3/2,j-1/2}^n - Vy_{i+1/2,j-3/2}^n - Vy_{i+3/2,j-3/2}^n}{4hxy}$$

$$D = \frac{8/3Vx_{i+1/2,j}^n - 4Vx_{i+1/2,j-1/2}^n + 4/3Vx_{i+1/2,j-3/2}^n}{hy^2}$$

$$E = \frac{Vy_{i+3/2,j}^n - Vy_{i+1/2,j}^n - 2Vy_{i,j}^n}{2hx}$$

$$F = \frac{2Vx_{i+1/2,j}^n - Vx_{i+1/2,j-1/2}^n - Vx_{i+1/2,j-3/2}^n}{2hy}$$

$i = \text{Hacia atrás}, j = \text{Hacia adelante}:$

$$A = \frac{8/3Vy_{i,j+1/2}^n - 4Vy_{i-1/2,j+1/2}^n + 4/3Vy_{i-3/2,j+1/2}^n}{hx^2}$$

$$B = \frac{4(Vx_{i,j+1}^n - Vx_{i,0}^n + Vx_{i-1,j}^n) - Vx_{i-1/2,j+3/2}^n - Vx_{i-3/2,j+3/2}^n - Vx_{i-3/2,j+1/2}^n - Vx_{i-1/2,j+1/2}^n}{4hxy}$$

$$C = \frac{4(Vx_{i,j+1}^n - Vx_{i,j}^n + Vx_{i-1,j}^n) - Vx_{i-1/2,j+3/2}^n - Vx_{i-3/2,j+3/2}^n - Vx_{i-3/2,j+1/2}^n - Vx_{i-1/2,j+1/2}^n}{4hxy}$$

$$D = \frac{4/3Vx_{i-1/2,j+3/2}^n - 4Vx_{i-1/2,j+1/2}^n + 8/3Vx_{i-1/2,j}^n}{hy^2}$$

$$E = \frac{2Vy_{i,j+1/2}^n - Vy_{i-1/2,j+1/2}^n - 2Vy_{i-3/2,j+1/2}^n}{2hx}$$

$$F = \frac{Vx_{i,j+3/2}^n + Vx_{i,j+1/2}^n - 2Vx_{i,j}^n}{2hy}$$

Ecuación de Compresibilidad Artificial:

Aproximaciones en la Frontera:

$i = \text{Hacia adelante:}$

$$\begin{aligned} & \frac{p_{i+1/2,j+1/2}^{n+1}}{\beta\Delta t} + \frac{vx_{i+3/2,j+1/2}^{n+1}}{4\Delta x} + \frac{vx_{i+1/2,j+1/2}^{n+1}}{4\Delta x} + \frac{vy_{i+1/2,j+3/2}^{n+1}}{4\Delta y} + \frac{vy_{i+1/2,j+1/2}^{n+1}}{4\Delta y} \\ & - \frac{vy_{i+1/2,j+1/2}^{n+1}}{4\Delta y} - \frac{vy_{i+1/2,j-1/2}^{n+1}}{4\Delta y} = \frac{p_{i+1/2,j+1/2}^n}{\beta\Delta t} + \frac{vx_{i,j+1/2}^{n+1}}{2\Delta x} - \frac{vx_{i+3/2,j+1/2}^n}{4\Delta x} - \frac{vx_{i+1/2,j+1/2}^n}{4\Delta x} \\ & + \frac{vx_{i,j+1/2}^n}{2\Delta x} - \frac{vy_{i+1/2,j+3/2}^n}{4\Delta y} - \frac{vy_{i+1/2,j+1/2}^n}{4\Delta y} + \frac{vy_{i+1/2,j+1/2}^n}{4\Delta y} + \frac{vy_{i+1/2,j-1/2}^n}{4\Delta y} \end{aligned}$$

$i = \text{Hacia atrás:}$

$$\begin{aligned} & \frac{p_{i-1/2,j+1/2}^{n+1}}{\beta\Delta t} - \frac{vx_{i-3/2,j+1/2}^{n+1}}{4\Delta x} - \frac{vx_{i-1/2,j+1/2}^{n+1}}{4\Delta x} + \frac{vy_{i-1/2,j+3/2}^{n+1}}{4\Delta y} + \frac{vy_{i-1/2,j+1/2}^{n+1}}{4\Delta y} - \frac{vy_{i-1/2,j+1/2}^{n+1}}{4\Delta y} \\ & - \frac{vy_{i-1/2,j-1/2}^{n+1}}{4\Delta y} = \frac{p_{i-1/2,j+1/2}^n}{\beta\Delta t} - \frac{vx_{i,j+1/2}^{n+1}}{2\Delta x} - \frac{vx_{i,j+1/2}^n}{2\Delta x} + \frac{vx_{i-3/2,j+1/2}^n}{4\Delta x} + \frac{vx_{i-1/2,j+1/2}^n}{4\Delta x} \\ & - \frac{vy_{i-1/2,j+3/2}^n}{4\Delta y} - \frac{vy_{i-1/2,j+1/2}^n}{4\Delta y} + \frac{vy_{i-1/2,j+1/2}^n}{4\Delta y} + \frac{vy_{i-1/2,j-1/2}^n}{4\Delta y} \end{aligned}$$

$j = \text{Hacia adelante:}$

$$\begin{aligned} & \frac{p_{i+1/2,j+1/2}^{n+1}}{\beta\Delta t} + \frac{vx_{i+3/2,j+1/2}^{n+1}}{4\Delta x} + \frac{vx_{i+1/2,j+1/2}^{n+1}}{4\Delta x} - \frac{vx_{i+1/2,j+1/2}^{n+1}}{4\Delta x} - \frac{vx_{i-1/2,j+1/2}^{n+1}}{4\Delta x} \\ & + \frac{vy_{i+1/2,j+3/2}^{n+1}}{4\Delta y} + \frac{vy_{i+1/2,j+1/2}^{n+1}}{4\Delta y} = \frac{p_{i+1/2,j+1/2}^n}{\beta\Delta t} + \frac{vy_{i+1/2,j}^{n+1}}{2\Delta y} + \frac{vy_{i+1/2,j}^n}{2\Delta y} - \frac{vy_{i+1/2,j+3/2}^n}{4\Delta y} \\ & - \frac{vy_{i+1/2,j+1/2}^n}{4\Delta y} - \frac{vx_{i+3/2,j+1/2}^n}{4\Delta x} - \frac{vx_{i+1/2,j+1/2}^n}{4\Delta x} + \frac{vx_{i+1/2,j+1/2}^n}{4\Delta x} + \frac{vx_{i-1/2,j+1/2}^n}{4\Delta x} \end{aligned}$$

$j = \text{Hacia atrás:}$

$$\begin{aligned} & \frac{p_{i+1/2,j-1/2}^{n+1}}{\beta\Delta t} + \frac{vx_{i+3/2,j-1/2}^{n+1}}{4\Delta x} + \frac{vx_{i+1/2,j-1/2}^{n+1}}{4\Delta x} - \frac{vx_{i+1/2,j-1/2}^{n+1}}{4\Delta x} - \frac{vx_{i-1/2,j-1/2}^{n+1}}{4\Delta x} \\ & - \frac{vy_{i+1/2,j-3/2}^{n+1}}{4\Delta y} - \frac{vy_{i+1/2,j-1/2}^{n+1}}{4\Delta y} = \frac{p_{i+1/2,j-1/2}^n}{\beta\Delta t} - \frac{vx_{i+3/2,j-1/2}^n}{4\Delta x} - \frac{vx_{i+1/2,j-1/2}^n}{4\Delta x} \\ & + \frac{vx_{i+1/2,j-1/2}^n}{4\Delta x} + \frac{vx_{i-1/2,j-1/2}^n}{4\Delta x} - \frac{vy_{i+1/2,j}^{n+1}}{2\Delta y} - \frac{vy_{i+1/2,j}^n}{2\Delta y} + \frac{vy_{i+1/2,j-3/2}^n}{4\Delta y} + \frac{vy_{i+1/2,j-1/2}^n}{4\Delta y} \end{aligned}$$

$i = \text{Hacia adelante}; j = \text{Hacia adelante}:$

$$\begin{aligned} & \frac{p_{i+1/2,j+1/2}^{n+1}}{\beta\Delta t} + \frac{vx_{i+3/2,j+1/2}^{n+1}}{4\Delta x} + \frac{vx_{i+1/2,j+1/2}^{n+1}}{4\Delta x} + \frac{vy_{i+1/2,j+3/2}^{n+1}}{4\Delta y} + \frac{vy_{i+1/2,j+1/2}^{n+1}}{4\Delta y} = \frac{p_{i+1/2,j+1/2}^n}{\beta\Delta t} \\ & - \frac{vy_{i+1/2,j+1/2}^n}{4\Delta y} - \frac{vy_{i+1/2,j+3/2}^n}{4\Delta y} + \frac{vy_{i+1/2,j}^n}{2\Delta y} - \frac{vx_{i+1/2,j+1/2}^n}{4\Delta x} - \frac{vx_{i+3/2,j+1/2}^n}{4\Delta x} + \frac{vx_{i,j+1/2}^n}{2\Delta x} \\ & + \frac{vx_{i,j+1/2}^{n+1}}{2\Delta x} + \frac{vy_{i+1/2,j}^{n+1}}{2\Delta y} \end{aligned}$$

$i = \text{Hacia adelante}; j = \text{Hacia atrás}:$

$$\begin{aligned} & \frac{p_{i+1/2,j-1/2}^{n+1}}{\beta\Delta t} + \frac{vx_{i+3/2,j-1/2}^{n+1}}{4\Delta x} + \frac{vx_{i+1/2,j-1/2}^{n+1}}{4\Delta x} - \frac{vy_{i+1/2,j-3/2}^{n+1}}{4\Delta y} - \frac{vy_{i+1/2,j-1/2}^{n+1}}{4\Delta y} = \frac{p_{i+1/2,j-1/2}^n}{\beta\Delta t} \\ & - \frac{vx_{i+3/2,j-1/2}^n}{4\Delta x} - \frac{vx_{i+1/2,j-1/2}^n}{4\Delta x} + \frac{vx_{i,j-1/2}^n}{2\Delta x} - \frac{vy_{i+1/2,j}^n}{2\Delta y} + \frac{vy_{i+1/2,j-1/2}^n}{4\Delta y} + \frac{vy_{i+1/2,j-3/2}^n}{4\Delta y} \\ & + \frac{vx_{i,j-1/2}^{n+1}}{2\Delta x} - \frac{vy_{i+1/2,j}^{n+1}}{2\Delta y} \end{aligned}$$

$i = \text{Hacia atrás}; j = \text{Hacia adelante}:$

$$\begin{aligned} & \frac{p_{i-1/2,j+1/2}^{n+1}}{\beta\Delta t} - \frac{vx_{i-1/2,j+1/2}^{n+1}}{4\Delta x} - \frac{vx_{i-3/2,j+1/2}^{n+1}}{4\Delta x} + \frac{vy_{i-1/2,j+1/2}^{n+1}}{4\Delta y} + \frac{vy_{i-1/2,j+3/2}^{n+1}}{4\Delta y} = \frac{p_{i-1/2,j+1/2}^n}{\beta\Delta t} \\ & - \frac{vx_{i,j+1/2}^n}{2\Delta x} - \frac{vx_{i-1/2,j+1/2}^n}{4\Delta x} - \frac{vx_{i-3/2,j+1/2}^n}{4\Delta x} - \frac{vy_{i-1/2,j+3/2}^n}{4\Delta y} - \frac{vy_{i-1/2,j+1/2}^n}{4\Delta y} - \frac{vy_{i-1/2,j}^n}{2\Delta y} - \frac{vx_{i,j+1/2}^{n+1}}{2\Delta x} \\ & + \frac{vy_{i-1/2,j}^{n+1}}{2\Delta y} \end{aligned}$$

$i = \text{Hacia atrás}; j = \text{Hacia atrás}:$

$$\begin{aligned} & \frac{p_{i-1/2,j-1/2}^{n+1}}{\beta\Delta t} - \frac{vx_{i-1/2,j-1/2}^{n+1}}{4\Delta x} - \frac{vx_{i-3/2,j-1/2}^{n+1}}{4\Delta x} - \frac{vy_{i-1/2,j-1/2}^{n+1}}{4\Delta y} - \frac{vy_{i-1/2,j-3/2}^{n+1}}{4\Delta y} = \\ & \frac{p_{i-1/2,j-1/2}^n}{\beta\Delta t} - \frac{vy_{i-1/2,j}^n}{2\Delta y} + \frac{vy_{i-1/2,j-1/2}^n}{4\Delta y} + \frac{vy_{i-1/2,j-3/2}^n}{4\Delta y} - \frac{vx_{i,j-1/2}^n}{2\Delta x} + \frac{vx_{i-3/2,j-1/2}^n}{4\Delta x} \\ & + \frac{vx_{i-1/2,j-1/2}^n}{4\Delta x} - \frac{vx_{i,j-1/2}^{n+1}}{2\Delta x} - \frac{vy_{i-1/2,j}^{n+1}}{2\Delta y} \end{aligned}$$

Finalmente, una vez que hemos obtenidos los valores de A, B, C y D , para $\nabla\eta$; y E y F , para $\vec{\omega}$; tanto en los nodos internos, como en la frontera y el obstáculo del mallado. Se procederá a contruir el equivalente numérico de $\epsilon\vec{s}$.

Con los valores A, B, C y D obtenidos, $|\nabla\eta|$, es igual a:

$$|\nabla\eta| = \sqrt{(A - B)^2 + (C - D)^2}$$

Una vez obtenido $\nabla\eta$, al igual que su norma; se puede obtener \hat{n} :

$$\hat{n} = \frac{\nabla\eta}{|\nabla\eta|} = \begin{pmatrix} \frac{A-B}{|\nabla\eta|} \\ \frac{C-D}{|\nabla\eta|} \end{pmatrix} = \begin{pmatrix} \hat{n}_x \\ \hat{n}_y \end{pmatrix}$$

Ahora se desarrolla el vector \vec{s} :

$$\vec{s} = \hat{n} \times \vec{\omega} = \begin{pmatrix} \hat{i} & \hat{j} & \hat{k} \\ \hat{n}_x & \hat{n}_y & \hat{n}_z \\ 0 & 0 & E - F \end{pmatrix} = (E - F)(\hat{n}_y\hat{i} - \hat{n}_x\hat{j}) + 0 = (E - F) \begin{pmatrix} \hat{n}_y \\ -\hat{n}_x \end{pmatrix}$$

Finalmente, el término de Confinamiento $\epsilon\vec{s}$, se puede escribir como:

$$\epsilon\vec{s} = \begin{pmatrix} \epsilon s_x \\ \epsilon s_y \end{pmatrix}$$

en donde

$$s_x = \hat{n}_y(E - F)$$

$$s_y = -\hat{n}_x(E - F)$$

Capítulo 4

RESULTADOS Y CONCLUSIONES

4.1. Resultados Numéricos

Las pruebas para las simulaciones fueron evaluadas en base al realismo que debían presentar los resultados de acuerdo a las condiciones proporcionadas y el comportamiento computacional que presenta la simulación. Con respecto al tiempo computacional para el cálculo, depende no solo del tipo de computador, sino de las condiciones, complejidad y magnitud de la situación deseada. Debido a que los resultados fueron obtenidos independientemente de la visualización, no existe una relación entre el tiempo de computo y la fluidez en que es representada la animación. Es importante destacar, que las pruebas mostradas en este trabajo, son solo una pequeña muestra de la gran variedad y cantidad que pueden ser apreciados en la bibliografía o en reportes de experimentos de laboratorio. El objetivo principal de estas pruebas es verificar el comportamiento de las ecuaciones

en función de los métodos numéricos utilizados y no las capacidades de las herramientas que han sido creadas para tal fin.

En cuanto al hardware; en todas las pruebas se utilizó un procesador Intel Pentium IV 1,6 GHZ con 256 Mb de Ram.

El mallado utilizado en todas las pruebas fue de dimensión 20×30 , con $\Delta X = 1,0/30$ y $\Delta Y = 1,0/20$ y un refinamiento de $0,5 * \Delta X$ y $0,5 * \Delta Y$ cercano al obstáculo; como consecuencia dicho refinamiento aumentaría la cantidad de nodos. En cuanto a los parámetros de entrada; en todas las pruebas se utilizó el parámetro ν con un valor igual a $8,04E-1$, una inicialización de tipo parabólico $y = y*(1-x)$ de la presión en todo el dominio y el método utilizado para resolver el Sistema de Ecuaciones fue GMRES.

Prueba 1

Se establece una velocidad constante en X y Y igual a 0.0 , con un término de confinamiento igual a cero, $\beta = 1000$ y $\rho = 09,957E2$. Se puede observar la existencia de vórtices alrededor del obstáculo.(Figuras ??,??,?? y ??).

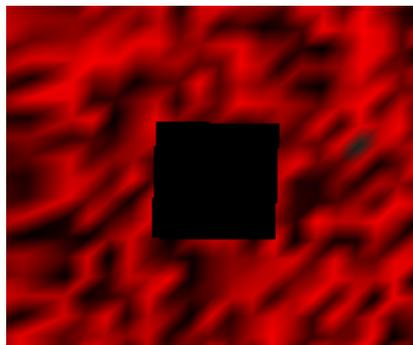


Figura 4.1: Prueba 1: Fluido

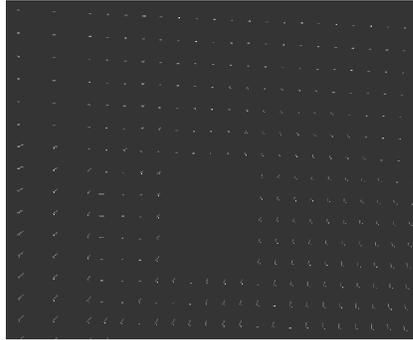


Figura 4.2: Prueba 1: Velocidad

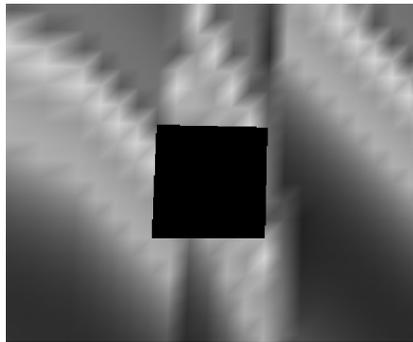


Figura 4.3: Prueba 1: Presión

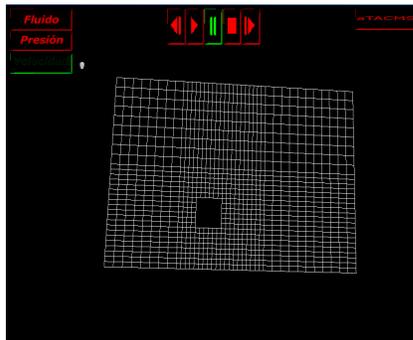


Figura 4.4: Prueba 1: Mallado

Prueba 2

Se establece una velocidad constante en X y Y igual a 0.0, con $\beta = 10000$, $\rho = 10,957E0$, un término de confinamiento distinto de cero y un $\epsilon = 0,25$,

disminuyendo así la vorticidad cercana al obstáculo.(Figuras ??, ??,?? y ??)

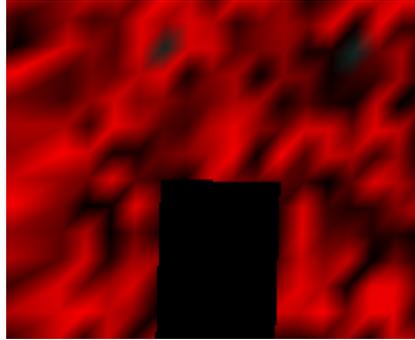


Figura 4.5: Prueba 2: Fluido

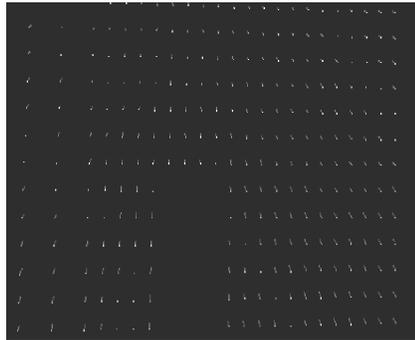


Figura 4.6: Prueba 2: Velocidad



Figura 4.7: Prueba 2: Presión

Prueba 3

Se establece una velocidad constante en X a 1.5 y 0.0 en Y, con $\beta = 10000$,

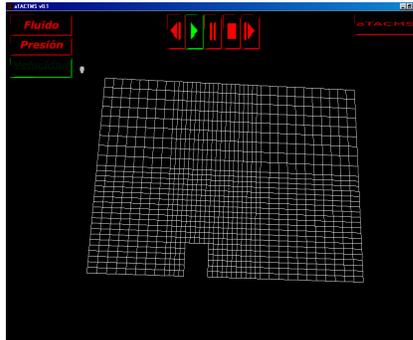


Figura 4.8: Prueba 2: Mallado

$\rho = 20,957E2$ y un término de confinamiento distinto de cero con $\epsilon = 0,5$, disminuyendo aún más la vorticidad cercana al obstáculo. Figuras ??,??,?? y ??

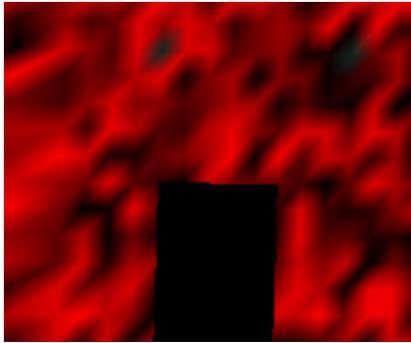


Figura 4.9: Prueba 3: Fluido

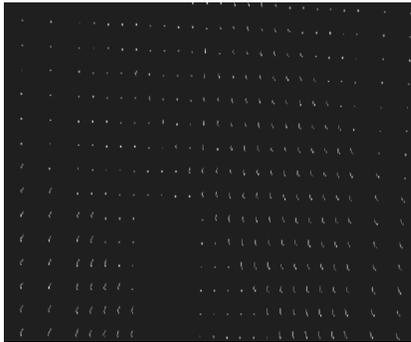


Figura 4.10: Prueba 3: Velocidad

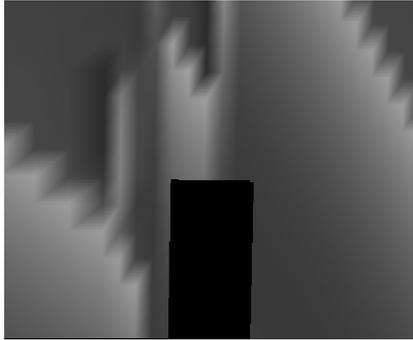


Figura 4.11: Prueba 3: Presión

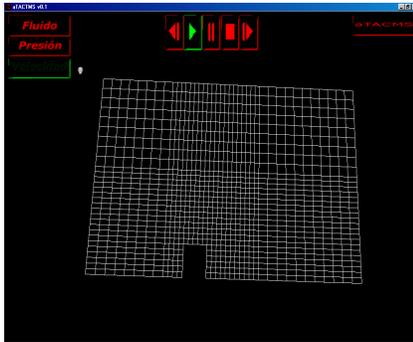


Figura 4.12: Prueba 3: Mallado

Prueba 4

Se establece una velocidad constante en X a 1.0 y 0.0 en Y, un término de confinamiento distinto de cero con $\epsilon = 5,0$; el parámetro β es llevado al valor de 5000 y $\rho = 09,957E2$, ocasionando una deformación del comportamiento de la velocidad a partir de la cuarta o quinta iteración; se puede notar también una ausencia total de vórtices cerca del obstáculo.(Figuras ??,??,?? y ??)

Prueba 5

Se establece una velocidad constante en X a 1.0 y 0.0 en Y, un término de confinamiento distinto de cero con $\epsilon = 15,0$; el parámetro $\beta = 10000$ y

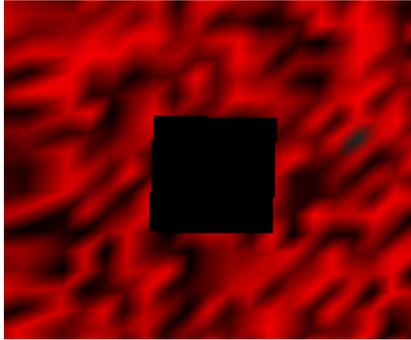


Figura 4.13: Prueba 4: Fluido

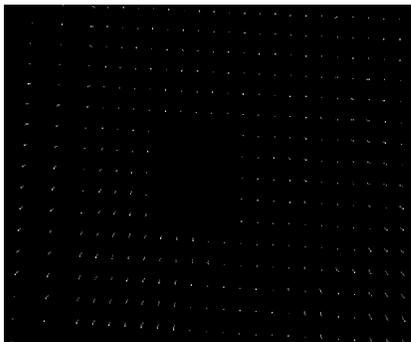


Figura 4.14: Prueba 4: Velocidad

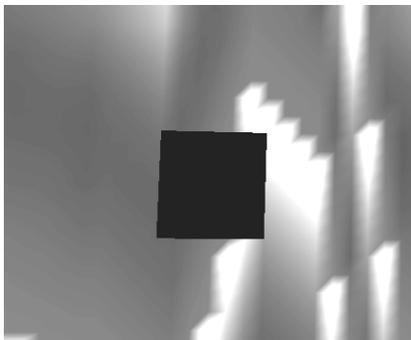


Figura 4.15: Prueba 4: Presión

$\rho = 09,957E2$; convergiendo la velocidad después de sucesivas iteraciones al comportamiento deseado; se puede notar también una ausencia total de vórtices cerca del obstáculo.(Figuras ??,??,?? y ??)

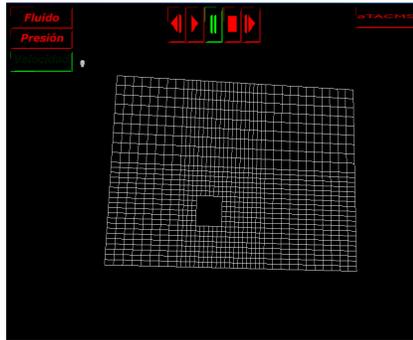


Figura 4.16: Prueba 4: Mallado

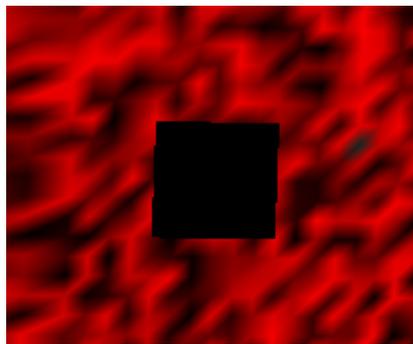


Figura 4.17: Prueba 5: Fluido

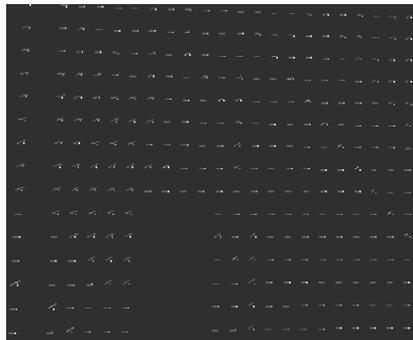


Figura 4.18: Prueba 5: Velocidad

Prueba 6

Se establece una velocidad constante en X a 1.0 y 0.0 en Y, un termino de confinamiento distinto de cero con $\epsilon = 25,0$; el valor de los parámetros β y ρ

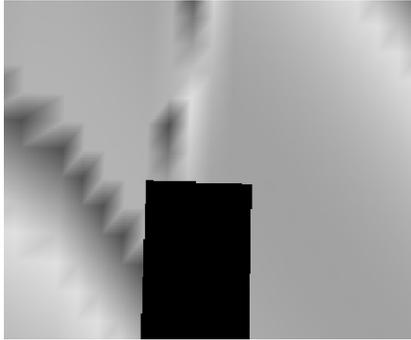


Figura 4.19: Prueba 5: Presión

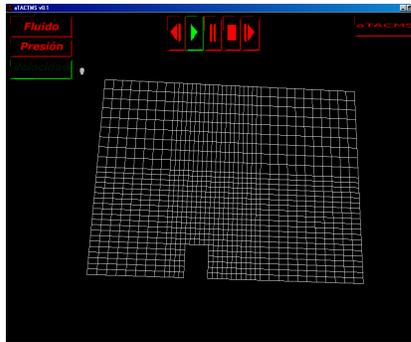


Figura 4.20: Prueba 5: Mallado

son mantenidos en 10000 y $09,957E2$ respectivamente, convergiendo la simulación después de sucesivas iteraciones al comportamiento deseado; se puede notar también una ausencia total de vórtices cerca del obstáculo.(Figuras ??, ??,?? y ??)

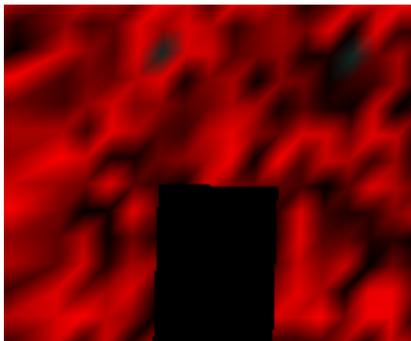


Figura 4.21: Prueba 6: Fluido

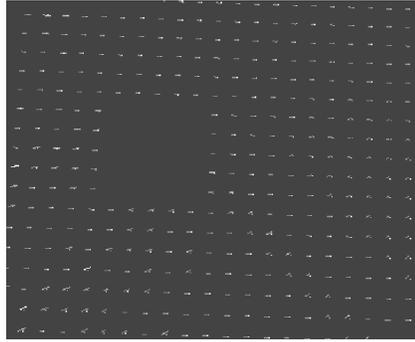


Figura 4.22: Prueba 6: Velocidad

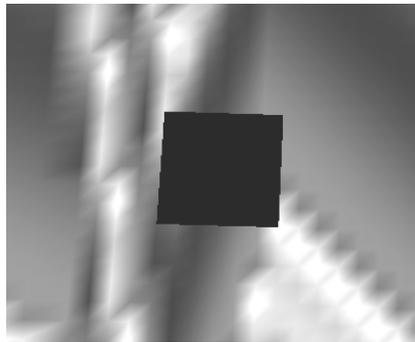


Figura 4.23: Prueba 6: Presión

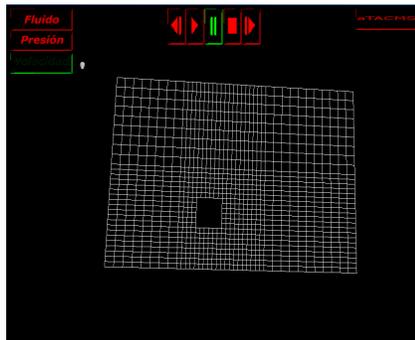


Figura 4.24: Prueba 6: Mallado

4.2. Conclusiones y Recomendaciones

- El método implementado(GMRES) convergen en todos los casos de prueba mostrados anteriormente. En los casos que diverge, fueron en aquellos en

que se cambiaba el método o los parámetros de entrada.

- Los resultados obtenidos en las simulaciones, en base a los casos de prueba, presentan una respuesta semejante a la que prodría ocurrir en la realidad, de concordancia con la teoría existente; más en concreto en lo concerniente al termino de confinamiento, es decir, en dichos experimentos se concluye, que mientras mayor sea el valor del dicho término, más ajustado al cuerpo poligonal estará el flujo y por ende prevendrá la disipación de las estructuras vorticales.
- En todos los casos la tolerancia del error en el cálculo de la solución del sistema de ecuaciones por el método GMRES fue de $1.0E-12$, exactitud generalmente alcanzada en las primeras iteraciones y por lo que no se considera necesario cambiar el método.

4.3. Trabajos Futuros

- Realizar una mayor cantidad de pruebas, con un refinamiento mayor del dominio, con la finalidad de experimentar con otros cuerpos poligonales más complejos.
- Desarrollar los cálculos y la visualización para resolver los problemas en tres dimensiones.
- Reprogramar el código obtenido, utilizando el enfoque de operadores.
- Profundizar en el estudio y modelos matemáticos para escenarios con diferentes flujos interactuando entre sí, con estructuras elásticas y/o medios porosos.

Bibliografía

- [1] M. Shaskov and S. Steinberg (1996). *Solving Diffusion Equations with Rough Coefficients in Rough Grids*, Journal of Computational Physics, **129**, 383–405.
- [2] J. M. Hyman y M. Shaskov (1998), *Mimetic Discretizations for Maxwell's Equations and the Equations of Magnetic Diffusion*, LA-UR-98-1032.
- [3] J. E. Castillo, J. M. Hyman, M. Shashkov y S. Steinberg (2001), *Fourth and Six Order Conservative Finite-Difference approximations of the Divergence and Gradient*, Applied Numerical Mathematics **37**, 171–187.
- [4] J. Castillo y D. Grone, (2003). *A Matrix Analysis Approach to Higher Order Approximations for Divergence and Gradients Satisfying a Global Conservation Law*, SIAM J. Matrix Anal. Appl., **25**:1, 128–142.
- [5] B. Jiang (1998). *The Least-Square Finite Element Method. Theory and Applications in Computational Fluid Dynamics and Electromagnetic*, Edit. Springer, New York, NY 10010, USA.
- [6] J. Stam (1999). *Stable Fluids*, SIGGRAPH 99 Conference Proceedings, annual Conference Series, 121–128.

- [7] J. Stam (2001). *A Simple Fluid Solver based on the FFT*, Journal of Graphics Tools, **6**:2, 43–52.
- [8] M. Fagúndez y J. Medina (2003). *Simulación Numérica de Flujo de Fluido Viscoso Incompresible en 2D*, Trabajo Especial de Grado, Facultad de Ciencia y Tecnología, Departamento de Computación, Universidad de Carabobo.
- [9] G. Larrazabal (2002). *UCSparceLib: Una Librería Numérica para la resolución de sistemas Lineales Dispersos*, Facultad de Ciencia y Tecnología, Departamento de Computación, Universidad de Carabobo.
- [10] Y. Wenren, M. Fan, W. Dietz, G. Hu, C. Braun, J. Steinhoff, B. Grossman (2001). *Efficient Eulerian Computation of Realistic Rotorcraft Flows Using Vorticity Confinement*, 39th AIAA Aerospace Sciences Meeting and Exhibit 8–11 Jan 2001, Reno NV.
- [11] W. Dietz, M. Fan, J. Steinhoff, Y. Wenren (2001). *Application of Vorticity Confinement to the Prediction of the Flow Over Complex Bodies*, AIAA CFD Conference 11–14 June 2001, Anaheim, CA.
- [12] M. J. Harris (2003). *Real-Time Cloud Simulation and Rendering*, Tesis Doctoral, University of North Carolina at Chapel Hill.
- [13] I. Sommerville (1996). *Software Engineering*, Fifth Edition, Edit. Addison-Wesley, Boston, MA, USA.