

Capítulo 1

Introducción

Comienza describiendo el avance de la tecnología computacional para abordar la solución de problemas físicos.

A comienzos del siglo pasado las computadoras, tal como se les conoce actualmente, estaban únicamente en la imaginación de unos cuantos, sin embargo, había quienes ya pensaban en la resolución numérica de problemas, como es el caso de Lewis Fray Richardson, quien presentó un modelo de predicción climática el cuál consistía en cálculos de “computadoras humanas” alrededor del mundo que se pasaran mensajes entre sí buscando predecir el clima con la precisión que se hace hoy en día [7]. El avance en la tecnología a mediados de ese siglo le permitió a Richardson presenciar la primer predicción climática hecha por ENIAC con base en sus modelos planteados, y desde ese momento en adelante los modelos computacionales y la resolución de problemas físicos no volverían a desprenderse el uno del otro, pasando por toda la computación involucrada en vuelos espaciales y estructuras complejas [9], el desafortunado proyecto Manhattan en la segunda guerra mundial, los inicios de la simulación de fluidos [3], su aplicación en industrias aéreas tales como Boeing o General Electrics y hasta su uso en industrias no aéreas como General Motors o Ford.

En la actualidad es de interés en la comunidad científica el estudio de *fenómenos de transporte*, término utilizado para describir procesos que involucran el movimiento de varias entidades tales como la masa, el momento, el momento angular o energía a través de un fluido o de un sólido con la característica de no tener condiciones uniformes dentro del medio mencionado, como puede ser la dinámica de fluidos, el transporte de calor o el fenómeno de difusión que a su vez incluyen pero no están limitados a: predicción de fenómenos meteorológicos, comportamiento de pandemias, estudios de salud, ingeniería aeronáutica, comportamiento de multitudes, entre otros. explica un poco lo que son los fenómenos de transporte y que problemas aborda.

Siguiendo con la idea anterior, en cuanto a la computación se refiere, una herramienta de estudio de los fenómenos de transporte es a lo que se le conoce como Dinámica de Fluidos Computacional (CFD por sus siglas en inglés). Como bien sabemos, los aspectos físicos de cualquier fluido están gobernados por tres principios fundamentales: La conservación de la masa, la segunda ley de Newton y la conservación de la energía; principios que pueden ser expresados entérminos de ecuaciones matemáticas, que en su forma más general suelen ser ecuaciones diferenciales parciales. CFD es, en cierto modo, el arte de reemplazar estas ecuaciones diferenciales parciales del flujo de un fluido por números y propagar estos números en el espacio o en el tiempo para así obtener una descripción numérica final de todo el fluido de interés [11]. continua describiendo qué es y cómo la CFD nos ayuda a resolver fenómenos de transporte

Menciona la efectividad de CFD en ciertos problemas de distintas disciplinas y busca más software para hacer CFD por ejemplo como **Fluent** <https://www.ansys.com/products/fluids/ansys-fluent> Continuando con la idea anterior, CFD es algo que ya se ha estudiando e implementado de muchas maneras y con distintos escenarios, de modo que existen *softwares* que lo hacen de manera sencilla y automatizado, tales como COMSOL, SimScale Ansys-Fluent, Autodesk CFD, entre otros. Estos sistemas usan los clásicos métodos de mallado o cuadrículado del espacio tales como *diferencias finitas*, *volumen finito*, *elemento finito*, *redes de Boltzman*, entre otros. Lo que hacen estos métodos, en general, es discretizar el espacio de estudio y analizar localmente cada elemento o su vecindad para así obtener entes discretos para analizar, tales como matrices o tensores. Mediante el uso de CFD se han podido abordar distintas problemáticas de áreas de conocimiento ajenas entre sí, como lo son la aviación, la aeronáutica, automovilismo, electrónica, salud (flujo sanguíneo, diseño de fármacos, flujo respiratorio, etc), dispersión de epidemias, comportamiento climático, entre otros.

Bien la justificación de desarrollar el software, sin embargo argumenta que es largo el desarrollo de una aplicación eficiente y efectiva.

Estos softwares realmente lo hacen muy bien, con interfaces muy amigables y agradables a la vista, sin embargo las licencias para usarlos son excesivamente caras y se tienen que renovar cada año si se quieren publicar artículos en los que se utilicen, cosa que no es viable de pagar para una institución mexicana pública actualmente, de modo que una alternativa sería comenzar con el desarrollo de software de código abierto de los algoritmos utilizados en estos programas y así borrar esa brecha que podría existir entre instituciones públicas con menos recursos que una privada, de modo que se pueda ofrecer un servicio de calidad y haya competencia interinstitucional. El desarrollo de este tipo de *softwares* es una tarea larga para lograr una eficiencia y control que esté a la altura de su uso para publicaciones, investigación y para llevar a la industria, esto debido a que si se quiere tener un modelo muy fino y cercano a la realidad se tiene que lidiar con sistemas de ecuaciones diferenciales parciales que lleva a resolver sistemas de ecuaciones lineales algebraicos enormes y dispersos de manera iterativa, de modo que si no se optimiza bien, se obtendrán resultados lentos y sin una fácil interpretación. Cierra esta parte e introduce la explicación de la presencia de ecuaciones diferenciales parciales en la dinámica de fluidos y como la solución numérica implica la solución de varios sistemas de ecuaciones lineales algebraicos SELA

Menciona problemas específicos para la solución numérica y ve “tejiendo” la necesidad de HPC Como ya se mencionó, para estos modelos se obtienen sistemas de ecuaciones lineales algebraicos que pueden llegar a ser muy grandes, además de que suelen ser dispersos, es decir, que la mayoría de sus entradas es cero, lo que lleva a la necesidad de utilizar metodologías específicas para su resolución, que por su naturaleza pueden ser tardados, y que para un modelo no tan básico, se tendrán que resolver no uno sino varios de estos sistemas tan rápido como se pueda para poder tener simulaciones casi en tiempo real, por lo que es necesario usar todos los recursos posibles para acelerar estos procesos. Actualmente se cuentan con tecnologías de código abierto para poder realizar estos programas sin la necesidad de haber estudiado por años un lenguaje de programación, pero además de necesitar esto, se espera que estas herramientas tengan la suficiente rapidez para lidiar con este tipo de problemas como lo hacían los lenguajes de programación antiguos como C, Fortran, etc. Estas nuevas tecnologías de programación se deben anexar al desarrollo de este tipo de softwares para tener un fácil acceso y aportaciones para cualquiera que pueda entender lo básico de programación. Siguiendo con la idea del modelado, la resolución de estos sistemas conlleva una carga computacional intensa para un procesador, ya que se hace uso de operaciones matriciales con matrices grandes y dispersas, por lo que el uso de tecnologías como GPU's podría ser de gran apoyo para las

necesidades de resolución de estos problemas. Cierra este párrafo vendiendo la idea de Julia y GPU's como dos herramientas que constituyen una excelente “navaja suiza” para obtener HPC

La implementación de dichos algoritmos no es cosa fácil, empezando por el entendimiento del modelo matemático y siguiendo con la implementación computacional para resolver dicho modelo como se explicó anteriormente, esto último contiene por un lado el dilema de hacer algoritmos de resolución con una ejecución rápida, y por otro lado la cuestión de que esta ejecución rápida no implique haber cursado toda una carrera en computación o algo afín para poder hacer un buen algoritmo.

Las dos últimas problemáticas son las que se buscan resolver con este trabajo mediante el cómputo de alto rendimiento y las herramientas que se tiene actualmente para tratar con éste, tales como el cómputo paralelo con Unidades de Procesamiento Gráfico (GPU's) y su manejo con lenguajes de programación de alto nivel que estén diseñados para el cómputo de alto rendimiento, tal como lo hace el *envoltorio* CUDA.jl del lenguaje de programación *Julia*, que nos ofrece todo el manejo que se tiene CUDA de las tarjetas de video para cómputo paralelo que originalmente se manipula mediante el lenguaje de programación C, pero que con el envoltorio se pueden ejecutar las mismas órdenes sin la necesidad de hacer la gestión de recursos y declaración de variables como se tenía que hacer en el pasado.

Termina mencionando como una aportación la utilización del envoltorio CUDANative.jl para programar GPU's.

Así, el presente trabajo de tesis pretende ofrecer una herramienta a la comunidad científica de fácil entendimiento e iniciar con la implementación de los alcances que tienen los softwares que se ofrecen en la actualidad, pero sin costo y con la finalidad de contribución de la comunidad para mejoras o nuevos alcances. Con lo que se tendría una aportación de una metodología completa, eficiente, rápida y comprensible para manejar problemas físicos mediante la CFD. Cierra esta sección mencionando que la aportación de tu trabajo consisten en una metodología completa, eficiente y rápida para resolver problemas físicos por medio de CFD

Motivación

Puedes abordar el avance en el desarrollo de lenguajes de programación de alto rendimiento y de alto nivel como herramienta para la CFD

La física y las ciencias de la computación convergen en varias ramas, como pueden ser el modelado numérico de distintos fenómenos, el análisis de imágenes, cálculos numéricos complejos, etc. Una rama muy importante dentro de este espectro de opciones es la Dinámica de Fluidos Computacional (CFD por sus siglas en inglés), que aborda fenómenos de transporte desde cómo se calienta una barra de acero unidimensional hasta la dispersión de partículas en el aire. Esta rama de estudio tiene una gran cantidad de aportes desde hace décadas que en la actualidad siguen creciendo y se sigue expandiendo su área de estudio y de aplicación. Un problema que aborda la dinámica de fluidos computacionales es la simulación numérica de flujo sanguíneo a través de diferentes zonas del cuerpo bajo distintas condiciones, un caso que es de sumo interés para la comunidad científica y médica es el de la simulación numérica del flujo sanguíneo a través de aneurismas y el vaso sanguíneo en que se encuentran. La simulación numérica de este tipo de problemas contiene una fuerte colaboración de parte de un físico, ya que se requieren conocimientos de fenómenos de transporte, física computacional, cálculos numéricos masivos, entre otros.

Debido a las complicaciones que puede surgir por la ubicación, forma y tamaño de un aneurisma, no es recomendable hacer análisis invasivos de éste, por lo que técnicas provenientes de imágenes médicas o simulaciones son el camino que se debe de tomar. Al modelar numéricamente un aneurisma se puede saber cuál es la situación médica del paciente y si bajo ciertas situaciones podría existir algún riesgo, por lo que se podría dar un diagnóstico temprano y recomendaciones para tratar su salud y evitar el peor escenario.

Dentro de este marco, se tiene que en la actualidad hay varios sistemas que pueden simular situaciones como las ya mencionadas, tales como ANSYS FLUENTE, COMSOL, etc. Sin embargo la principal problemática que presentan estos softwares es que se venden sus licencias a universidades e institutos en precios muy elevados y además se tiene que renovar cada año la licencia, por lo que implica un costo que se podría eliminar de universidades y hospitales públicos con softwares de código abierto que tengan un entendimiento que no necesite años de academia para poder utilizarlo, como podrían ser los lenguajes de alto nivel y de alto rendimiento, como lo que nos ofrece Julia. Además, este tipo de problemas se pueden refinar para obtener resultados más cercanos a la realidad, cosa que es muy importante en el área médica, por lo que el uso de tarjetas de video o GPUs puede ser de ayuda para la simulación rápida y más exacta de este tipo de situaciones. Otra ventaja que presenta Julia es que también puede lidiar con este tipo de hardware de manera más sencilla a como se hacía en el pasado con otros lenguajes de programación, por lo que se está acortando la curva de aprendizaje tanto de la implementación del hardware como la del software.

Objetivo

El objetivo general de este trabajo consiste en:

Definir una metodología físico-computacional para resolver problemas físicos que tengan alta demanda computacional en el modelado, resolución y simulación de fluidos en diferentes bajo ciertas condiciones mediante el cómputo paralelo y lenguajes de alto nivel.

Los objetivos particulares consisten en:

- *Desarrollar el modelo matemático y computacional de las ecuaciones que describen el comportamiento de un fluido.*
- *Utilizar una estrategia para la resolución numérica proveniente del modelado matemático y computacional.*
- *Experimentar con herramientas de cómputo de alto rendimiento como el cómputo paralelo para la resolución numérica del problema.*

Metodología

Básicamente el recuento de todo lo que vas a hacer, subraya `julia+cuda+CUDataNative.jl` Un recuento no se hace hasta que se hizo algo?

La introducción y profundización en el problema y lo que conlleva fue parte importante del comienzo del trabajo, desde saber qué tipo de problemas resolvía la física computacional que necesiten cómputo de alto rendimiento, qué es un aneurisma, cómo se ha simulado el flujo sanguíneo dentro y fuera de un aneurisma, cuáles son las estrategias que se siguen para controlar lo que pueda llegar a provocar un aneurisma y por qué se hacen simulaciones numéricas para tratar este problema.

En relación a la problemática expuesta, lo que siguió fue investigar sobre cómo se ha modelado matemáticamente este problema de modo que se pudiera llevar a un modelado

computacional. Se llegó a que este tipo de problemas puede ser modelado por las ecuaciones de Navier-stokes bajo ciertas condiciones que impone el fenómeno en cuestión, como la densidad de la sangre, las paredes de los vasos sanguíneos y de un aneurisma, entre otras cosas.

En función de lo planteado, se comenzó con la investigación de los métodos numéricos que existen para la simulación de flujo sanguíneo, con esto se llegó a que los más utilizados incluyen el método de diferencias finitas, el método de volumen finito y el método del elemento finito. De modo que se plantearon las ventajas y desventajas que tenían cada uno de estos, con lo que se llegó a la conclusión de utilizar el método del volumen finito como modelado computacional del problema.

Se adopta entonces un modelado previamente hecho por un compañero de investigación. Con este modelado como base, se modificó para que tuviera una estructura que pudiera ser mejor interpretable físicamente de acuerdo al modelado matemático y físico, comenzando con pruebas de modelos sencillos en una dimensión y generalizando a 3 dimensiones, corroborando que el programa desarrollado tuviera resultados congruentes al comparar con resultados de distintos libros.

Hipótesis de investigación

Incluimos como hipótesis algo relativo a la eficacia de utilizar HPC julia+cuda+CUANative en la paralelización de la solución de un problema de CFD en términos de speedup y escalamiento, velocidad de desarrollo, etc.

Metas

Se establecieron como metas:

1. Mostrar la efectividad de usar un lenguaje de programación de alto nivel que a su vez sea un lenguaje de programación especializado en el alto rendimiento como lo es Julia aplicado a técnicas de CFD.
2. Mostrar la efectividad del uso de herramientas de cómputo de alto rendimiento como lo es CUDA y su fácil implementación que puede ser utilizando el *wrapper* CUDA.jl de Julia.
3. Probar que con las herramientas de software indicadas (las ya mencionadas) se pueden realizar simulaciones numéricas en una infraestructura básica de cómputo que podrían ser costosas computacionalmente.
4. Incluir y probar la efectividad del método Monte Carlo - Cadenas de Markov como preconditionador para la solución de un sistema de ecuaciones lineales algebraico.

Contribución

La contribución de este trabajo es:

Un método físico-computacional adecuado a las tecnologías actuales, de software libre y con lenguajes de alto nivel que pueda resolver problemas que se abordan mediante CFD.

Logros

A lo que lleguemos, todavía no hemos llegado a nada, esto reafirma lo que había dicho arriba de que esto se debería plantear hasta el final.

Organización de la Tesis

Desarrolla más esta parte cuando ya se tengan los demás capítulos Este trabajo se desarrolla en 6 capítulos, incluyendo esta **introducción**, y una sección adicional de apéndices que complementan la investigación.

En el Capítulo 2, se presentan los **antecedentes**. Donde se plantean más a fondo los modelos matemáticos utilizados para la discretización del espacio y así describir cómo se hace el análisis de fluidos inmersos en condiciones específicas, así como la resolución numérica de los sistemas de ecuaciones lineales algebraicos dados por los modelos mencionados anteriormente y algunos *solver* que se utilizan actualmente para resolver esta tarea. También se abordan aquí trabajos relacionados a esta tesis con el fin de poder comparar resultados. Además se presentan las herramientas computacionales a utilizar y el por qué de su utilización en este trabajo.

En el Capítulo 3, **implementación computacional**, se abordan los modelos de paralelización utilizados para la resolución numérica de los sistemas de ecuaciones lineales algebraicos para los distintos solvers utilizados.

En el Capítulo 4, se aborda la **experimentación(?)** con los distintos modelos, distintas implementaciones para el lenguaje de programación utilizado, el uso de distintos *solvers* con y sin preconditionadores.

En el Capítulo 5, se aborda el **análisis de resultados** provenientes de la experimentación realizada en el Capítulo 4, esto incluye pruebas de rendimiento, complejidad, escalabilidad, etc.

Se concluye el trabajo con una sección de **Conclusiones y Trabajo Futuro**, Capítulo 6, en donde se exponen las problemáticas encontradas al realizar cada implementación computacional, los beneficios que ofrece cada método de modelado y de resolución, así como las ventajas y desventajas de utilizar cada uno de éstos. Posteriormente se incluyen los comentarios finales y se sugiere el trabajo a relizar en próximas investigaciones.

Se anexan en la sección **Apéndices** los apartados: ...

Capítulo 2

Antecedentes

Esta parte podría tener esta estructura:

- Investigar trabajos que utilizan CFD para resolver problemas físicos y exponer su metodología
- Investigar trabajos que hacen lo anterior pero con HPC: lenguajes de alto nivel, paralelismo, CUDA+GPU's, o todo lo anterior.
- Investigar trabajos que hacen lo planteado en los puntos anteriores + Julia

Los trabajos de Iturriarán y las referencias en ellos nos van ayudar para esto

El presente trabajo está basado en un modelo matemático sobre el comportamiento físico de los fluidos. En este capítulo se presentan y discuten las ideas principales de dicho modelo, así como su discretización y el método numérico que se utiliza para simular la discretización; posteriormente se presentan los *solvers* que se utilizan para resolver los sistemas de ecuaciones lineales algebraicos que nos arroja el método numérico mencionado, finalmente se presentan los conceptos y paradigmas computacionales utilizados, tanto las herramientas del manejo de cómputo paralelo como el lenguaje de programación utilizado para el modelado y la resolución de los problemas. Todo esto con el objetivo de plantear el contexto teórico de esta tesis.

2.1. Problema Físico

Justificar el problema físico de interés y desarrollar el planteamiento

2.2. Modelado matemático

El estudio y la simulación de los fluidos se abordan desde el cálculo infinitesimal, por lo que dado un fluido, se toma un elemento finito de éste, del cual se toma una partícula infinitesimal de masa dm y volumen inicial $dx dy dz$ tal como se muestra en la Figura 2.1.

Se tiene además que la velocidad de esta partícula infinitesimal no viene dada por una ecuación para la partícula como tal sino como un campo vectorial dado por $\mathbf{u}(x, y, z)$. Con la velocidad expresada de esta manera lo que se necesita ahora es obtener la aceleración para poder utilizar la segunda ley de Newton, sin embargo aquí no podemos simplemente tomar la derivada parcial respecto al tiempo de la velocidad justamente porque no estamos hablando de una partícula solamente, esto se puede observar en la figura 2.2, que muestra que la velocidad

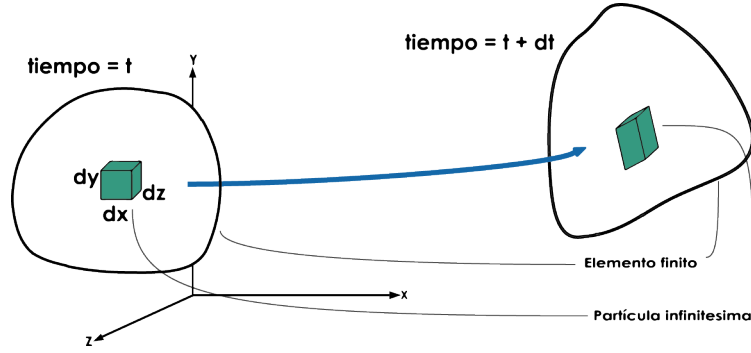


Figura 2.1: Elemento finito de un fluido y partícula infinitesimal en los tiempos t y $t + dt$.

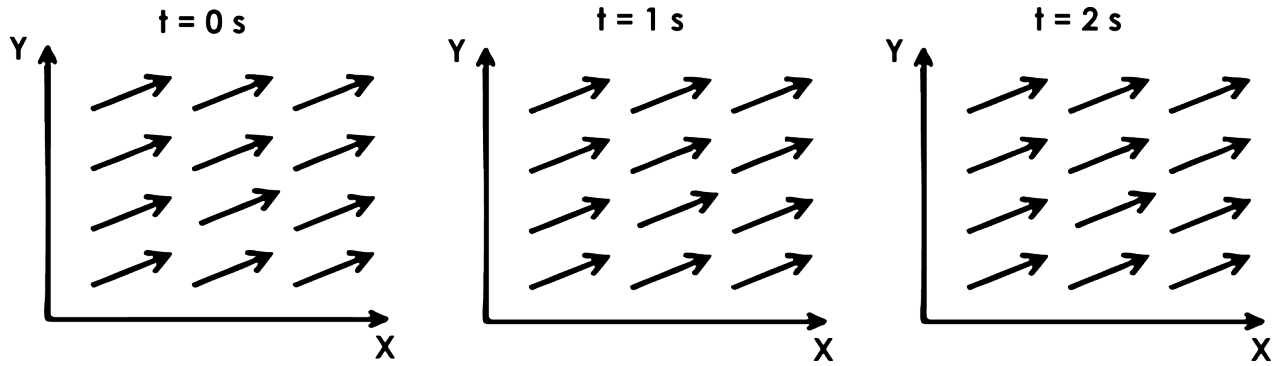


Figura 2.2: Diagrama de un campo de velocidades con variaciones respecto a las dimensiones espaciales pero no respecto al tiempo.

sí cambia respecto a la posición, pero no respecto al tiempo como vemos en la Figura 2.3, por lo que es necesario derivar respecto a las dimensiones espaciales y además respecto al tiempo.

Por lo que el cambio de la velocidad de la partícula $d\mathbf{u}_p$ se puede expresar mediante la regla de la cadena de la forma

$$d\mathbf{u}_p = \frac{\partial \mathbf{u}}{\partial x} dx_p + \frac{\partial \mathbf{u}}{\partial y} dy_p + \frac{\partial \mathbf{u}}{\partial z} dz_p + \frac{\partial \mathbf{u}}{\partial t} dt,$$

con lo que la aceleración total de la partícula queda de la siguiente forma:

$$\vec{a}_p = \frac{d\mathbf{u}_p}{dt} = \frac{\partial \mathbf{u}}{\partial x} \frac{dx_p}{dt} + \frac{\partial \mathbf{u}}{\partial y} \frac{dy_p}{dt} + \frac{\partial \mathbf{u}}{\partial z} \frac{dz_p}{dt} + \frac{\partial \mathbf{u}}{\partial t},$$

con lo que si definimos a $u = \frac{dx_p}{dt}$, $v = \frac{dy_p}{dt}$ y $w = \frac{dz_p}{dt}$ tenemos como resultado a lo que se le conoce como la *derivada material*:

$$\frac{D\mathbf{u}_p}{Dt} \equiv \vec{a}_p = u \frac{\partial \mathbf{u}}{\partial x} + v \frac{\partial \mathbf{u}}{\partial y} + w \frac{\partial \mathbf{u}}{\partial z} + \frac{\partial \mathbf{u}}{\partial t} = \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u}, \quad (2.1)$$

donde el término $(\mathbf{u} \cdot \nabla) \mathbf{u}$ es conocido como la **aceleración convectiva** o el término convectivo en este tipo de ecuaciones para fluidos. El término $\frac{\partial \mathbf{u}}{\partial t}$ es conocido como la aceleración local, y físicamente expresa que el fluido está en un estado no estacionario, por lo que su campo de

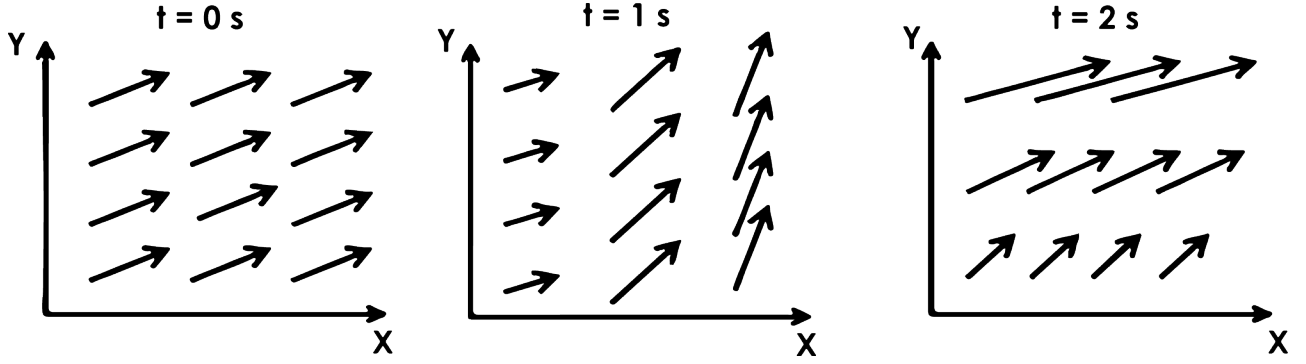


Figura 2.3: Diagrama de un campo de velocidades con variaciones respecto a las dimensiones espaciales y respecto al tiempo.

velocidades estará cambiando con el paso del tiempo; este término es importante para este modelo ya que el flujo sanguíneo no se considera un flujo estacionario.

Ya que se tiene cómo será la aceleración de una partícula infinitesimal en un fluido, es posible mostrar la segunda ley de Newton para este tipo de sistemas. Para la partícula de la que hablamos con masa dm , usando la ecuación 2.1, se tiene que

$$d\mathbf{F} = dm \left[\frac{d\mathbf{u}}{dt} \right]_{\text{sistema}} = dm \frac{D\mathbf{u}}{Dt} = dm \left[\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right] \quad (2.2)$$

Teniendo esta ecuación de momentum, es posible comparar con las fuerzas que actúan sobre la partícula categorizadas en dos grupos: fuerzas de cuerpo y fuerzas de superficie. Normalmente encontramos como una fuerza de cuerpo a la debida por la aceleración de la gravedad, sin embargo para el caso de los vasos sanguíneos no se tomará en cuenta debido a su tamaño, por lo que las únicas fuerzas actuando sobre la partícula serán las de superficie. En este tipo de fuerzas se encuentra dos, las normales y las tangenciales. Si de momento se consideran sólo las fuerzas normales tendremos que la única aportación viene dada por la fuerza debida a la presión aplicada sobre un área, esto para todas las caras, por lo tanto la ecuación 2.2 puede verse de la siguiente manera:

$$-\nabla p(dx dy dz) = dm \left[\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} \right],$$

y simplificando se tiene la ecuación de movimiento de Euler:

$$-\frac{1}{\rho} \nabla p = \frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u}, \quad (2.3)$$

si se estuviera tratando con otra clase de problema, la interacción con la gravedad sí podría afectar como una fuerza de cuerpo, por lo que se agregaría un término (como $\rho \mathbf{g}$) al lado izquierdo de la ecuación 2.3.

Para deducir la ecuación de movimiento de Euler no tomamos en cuenta fuerzas de superficie tangenciales, sin embargo para el modelo del flujo sanguíneo se tienen que considerar debido a la viscosidad de la sangre y a la interacción de esta con las paredes de los vasos sanguíneos. Para deducir este término se toma en cuenta el tensor de esfuerzos, por lo que cada entrada afectará de manera distinta a la fuerza superficial tangencial aplicada sobre la

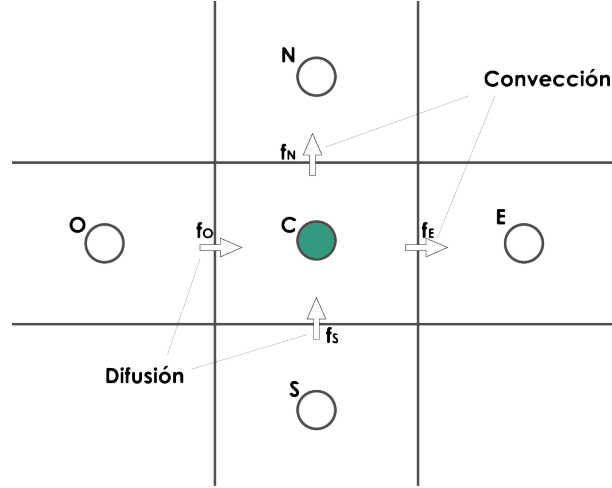


Figura 2.4: Representación gráfica de la partícula infinitesimal y los cambios que sufre debido a su movimiento y las fuerzas aplicadas sobre ésta.

partícula de estudio. De aquí se deduce un término difusivo para la ecuación de conservación de momentum. Si la viscosidad del fluido no es constante, esta ecuación toma una forma complicada, pero para el modelo del flujo sanguíneo sí se considera como constante, por lo que el término difusivo que se agrega será más inofensivo, de modo que el modelo final queda de la forma:

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} = -\frac{1}{\rho} \nabla p + \frac{\mu}{\rho} \nabla^2 \mathbf{u}, \quad (2.4)$$

que es una forma de las ecuaciones de Navier-Stokes.

Al pensar el modelo como muchas partículas pegadas entre sí, puede verse como se muestra en la Figura 2.4, con el término convectivo pensando en que representan a una cantidad física (masa) que es transportada (hacia la partícula de estudio) por el mero movimiento del flujo y el término difusivo, que representa el transporte de una cantidad física (masa) por el movimiento aleatorio de las partículas del flujo. Además también se toma en cuenta el término de flujo no estacionario, pensando en que a cada cambio infinitesimal de tiempo se tienen cambios en el campo de velocidades que afectan a estos dos términos antes mencionados.

Se tiene también la ecuación de continuidad en torno a una partícula infinitesimal con densidad ρ y volumen $dx dy dz$, que nos dice que todo lo que sale de un sistema está entrando en otro:

$$\nabla \cdot \rho \mathbf{u} + \frac{\partial \rho}{\partial t} = 0, \quad (2.5)$$

y como para la sangre tenemos que es constante en el tiempo y en el espacio (incompresibilidad) la ecuación 2.5 toma la forma

$$\nabla \cdot \mathbf{u} = 0 \quad (2.6)$$

Como se vio, la ecuación de movimiento de Euler es una simplificación de la ecuación 2.10. Un modelo más sencillo que éste sería teniendo un flujo estacionario en la ecuación 2.3, lo que daría como resultado

$$-\frac{1}{\rho} \nabla p = (\mathbf{u} \cdot \nabla) \mathbf{u}, \quad (2.7)$$

que junto con la ecuación 2.6 se podría tener el modelo mas sencillo para simular flujo

sanguíneo.

Como se está teniendo en consideración un flujo a través de un vaso sanguíneo se puede tomar el espacio como un cilindro, lo que implicaría que hay una simetría axial, de modo que las soluciones dependerán únicamente del radio del vaso sanguíneo y de la longitud, no del ángulo. Esto nos reduce una dimensión, tal como se muestra en la Figura 2.5. Teniendo esto en consideración se tiene que el vaso sanguíneo se verá ahora como un rectángulo con mallado rectangular, problema que también se puede resolver mediante las ecuaciones cartesianas, sin tener que recurrir a las ecuaciones de Navier - Stokes en coordenadas cilíndricas.

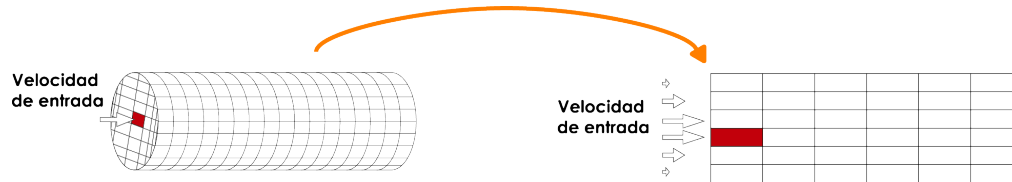


Figura 2.5: Reducción de una dimensión debido a la simetría axial.

Las ecuaciones que gobiernan estos problemas son las ecuaciones de Navier-Stokes y la ecuación de continuidad para la mecánica de fluidos:

$$\rho \frac{Du_i}{Dt} = \rho F_i - \frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_j} [2\mu(e_{ij} - \Delta\delta_{ij}/3)] \quad (2.8)$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0 \quad (2.9)$$

donde ρ es la densidad, t el tiempo, $\vec{u} = u_x \hat{i} + u_y \hat{j} + u_z \hat{k}$ la velocidad del fluido, P la presión, F_i el campo de aceleraciones creado por las fuerzas aplicadas sobre el cuerpo, como la gravedad, μ la viscosidad dinámica,

$$\frac{D}{Dt}(\cdot) \equiv \frac{\partial(\cdot)}{\partial t} + (\vec{u} \cdot \nabla)(\cdot),$$

la derivada total, δ_{ij} la delta de Kronecker y por último

$$e_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right),$$

con $\Delta = e_{ii}$. Al desarrollar la ecuación 2.8 tenemos:

$$\rho \left(\frac{\partial}{\partial t} + u_j \frac{\partial}{\partial x_j} \right) u_i = \rho F_i - \frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_j} \left[\mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \frac{\partial u_i}{\partial x_i} \delta_{ij} \right) \right],$$

donde al tener una viscosidad uniforme (segundo término del último elemento del lado derecho de la ecuación anterior), sin tomar en cuenta la gravedad (primer término del lado derecho de la igualdad anterior) y suponiendo un fluido incompresible

$$\rho \left(\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} \right) = - \frac{\partial P}{\partial x_i} + \mu \frac{\partial^2 u_i}{\partial x_j \partial x_j},$$

y la ecuación 2.9 queda de la forma

$$\frac{\partial u_i}{\partial x_i} = 0,$$

Y en sus formas vectoriales tenemos a la ecuación de Navier-Stokes:

$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u} = - \frac{1}{\rho} \nabla P + \frac{\mu}{\rho} \nabla^2 \mathbf{u} \quad (2.10)$$

y la ecuación de continuidad (incompresibilidad):

$$\nabla \cdot \mathbf{u} = 0 \quad (2.11)$$

Los términos del lado izquierdo de la ecuación 2.10 se refieren a lo que se le conoce en la física como la *aceleración total de la partícula en un campo de velocidades*, que se divide en la aceleración local (derivada parcial respecto al tiempo) y la aceleración convectiva, esto porque nos encontramos en un campo de velocidades que depende de las coordenadas x , y , z y t , no solamente de t . Del lado derecho de la ecuación se tienen dos términos que son más complicados de obtener, pero que básicamente vienen de las ecuaciones de la tensión cizallante

de pared (WSS):

$$\tau_{w,ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad i, j = x, y, z \quad (2.12)$$

que relacionan a las tensiones mencionadas con el cambio de velocidades tangenciales a la partícula. Esto nos lleva a que el cambio en el momentum de una partícula de fluido se debe a la fuerza neta de la presión (dirección de las partículas en cierto punto que se mueven a una zona con menor presión), dada por el primer elemento del lado derecho de la ecuación 2.10, y la fuerza de fricción, dada por el segundo elemento del lado derecho de dicha ecuación.

En la tabla ?? se hace un análisis más profundo de cada término de las ecuaciones 2.10 y 2.11.

Término de la ecuación	Significado físico
$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla) \mathbf{u}$	Aceleración total de la partícula.
$\frac{\partial \mathbf{u}}{\partial t}$	Fluido no estacionario
$(\mathbf{u} \cdot \nabla) \mathbf{u}$	Término convectivo
$\frac{\mu}{\rho} \nabla^2 \mathbf{u}$	Término difusivo
$-\frac{1}{\rho} \nabla P$	Cambio en la presión. Gradiente de presiones.
$\nabla \cdot \mathbf{u} = 0$	Incompresibilidad

Tabla 2.1: Explicación de los términos de las ecuaciones 2.10 y 2.11

Sin embargo, al parecer esto se suele hacer adimensional [4][5] de modo que tenemos que la ecuación 2.10

$$\frac{1}{\pi} \frac{\alpha^2}{Re_M} \frac{\partial \vec{U}}{\partial t} + (\vec{U} \cdot \nabla) \vec{U} = -\nabla P + \frac{2}{Re_M} \nabla^2 \vec{U}, \quad (2.13)$$

donde α es el número de Womersley y Re_M el número de Reynolds máximo. Cada expresión viene de:

$$\alpha = \frac{D}{2\sqrt{\rho\omega/\mu}} ; \quad Re_M = \frac{\rho U_{max} D}{\mu}$$

de donde $\omega = 2\pi/T$, con $T = 0.8s$ como lo vimos en la anterior sección, D es el diámetro de la vena hospedadora del aneurisma y U_{max} es la velocidad máxima de dicha vena. Las cantidades que se mencionan en el artículo [4] son:

- $T = 0.8s$
- $\rho = 1050kg/m^3$
- $\mu = 3.5cP$
- $D = 5mm$

En la referencia [1] se puede ver un procedimiento detallado con sus respectivas referencias. Sin embargo lo realizado ahí es para aneurismas de perros, por lo que las variables consideradas no son necesariamente ciertas para los humanos.

Para aneurismas con simetría axial se tiene un modelo muy específico [2][10][8] que en realidad parte de lo visto anteriormente pero con un cambio de coordenadas, se tiene que la ecuación de continuidad viene dada de la forma

$$\frac{\partial}{\partial x}(ru) + \frac{\partial}{\partial r}(rv) = 0, \quad (2.14)$$

donde u es la velocidad en la dirección longitudinal, v es la velocidad en la dirección radial. Y las ecuaciones qde Navier-Stokes en la forma de simetría axial son

$$\rho \left(\frac{\partial u}{\partial t} + v \frac{\partial u}{\partial r} + u \frac{\partial u}{\partial x} \right) = -\frac{\partial p}{\partial x} + \mu \left[\frac{1}{r} \frac{\partial}{\partial r} \left(r \frac{\partial u}{\partial r} \right) + \frac{\partial^2 u}{\partial x^2} \right] \quad (2.15)$$

$$\rho \left(\frac{\partial v}{\partial t} + v \frac{\partial v}{\partial r} + u \frac{\partial v}{\partial x} \right) = -\frac{\partial p}{\partial r} + \mu \left\{ \frac{\partial}{\partial r} \left[\frac{1}{r} \frac{\partial}{\partial r} (rv) \right] + \frac{\partial^2 v}{\partial x^2} \right\}, \quad (2.16)$$

donde p es la presión, ρ es la densidad del fluido y μ es la viscosidad cinemática, tal como se vio en las ecuaciones 2.8.

Este tipo de modelo se usa para geometrías con simetría axial, cuya estructura no tiene tanto sentido para el proble a del aneurisma ya que sabemos que aunque el vaso sanguíneo en el que se encuentra sí tiene dicha geometría, el mismo aneurisma no la tiene en la vida real, ya que sólo sale de un lado del vaso. Quizás la asunción que se podría hacer es considerar esta simetría y partir en dos en resultado, teniendo así como un efecto espejo para facilidades del mismo problema.

Justificar el método numérico que vas a utilizar, ventajas y desventajas sobre otros métodos, además de incluir la exposición breve y concisa del método. Existen varios métodos numéricos que se utilizan para modelar fenómenos físicos tales como la transferencia de calor, el flujo de algún fluido, electromagnetismo, etc. Entre éstos se encuentran los métodos de diferencias finitas, de elemento finito, de volumen finito, esprectales, de mallado de Boltzmann, basados en vorticidad, entre otros. Los más utilizados son los primeros tres y cada uno tiene sus ventajas y desventajas, tal como se muestra a continuación:

Método de diferencias finitas (FDM por sus siglas en inglés): Es el más antiguo de los tres y permaneció como el favorito durante muchos años, es fácil de implementar pero está restringido a un cuadriculado simple del espacio (rectangular o similar) y no tiene las propiedades de conservación de la energía, momento o masa en cuadrículas gruesas. Es un método basado en esquema diferencial, es decir, se puede pensar como una aproximación en serie de Taylor.

Método de Elemento Finito (FEM por sus siglas en inglés): Este es un método basado en el esquema integral, lo que significa que tomará para su análisis un volumen de análisis (elemento finito). Es el más difícil de implementar de los tres pero es el más general y tiene más aplicaciones para la física computacional. Los métodos numéricos con enfoque integral suelen ser más lentos pero ganan en cuanto a minimizar errores. No tiene dificultades con un mallado irregular ya que considera cada elemento de volumen y su contorno.

Método de Volumen Finito (FVM por sus siglas en inglés): Al igual que el FEM este es un método basado en el esquema integral. Por su naturaleza encaja bien con el modelado de problemas físicos ya que les son inertes las leyes de conservación de masa, momento y energía. Tampoco se ve afectado por la forma de los volúmenes de control que se estudian. Aunque no tiene tantas aplicaciones como el FEM, es el que presenta mejores resultados de los tres en cuanto a CFD. Entre estos principales, el FVM es el más nuevo y no es tan difícil de implementar o entender como el FEM.

Así, tomando en consideración que se busca implementar un método numérico para CFD con condiciones de conservación de masa y una geometría no necesariamente sencilla se optó por usar el método de volumen finito, el cuál se explica a continuación.

El método de volumen finito consiste en dividir el espacio de análisis en pequeños volúmenes de tamaño finito V_p , tal como se muestra en la figura 2.6, teniendo además el punto \mathbf{x}_p en el centro del volumen de control, y además cada uno de estos posee un número finito de caras cuyo centro llamaremos $\mathbf{x}_i p$ y se les conoce como *punto de integración*, con lo que se puede

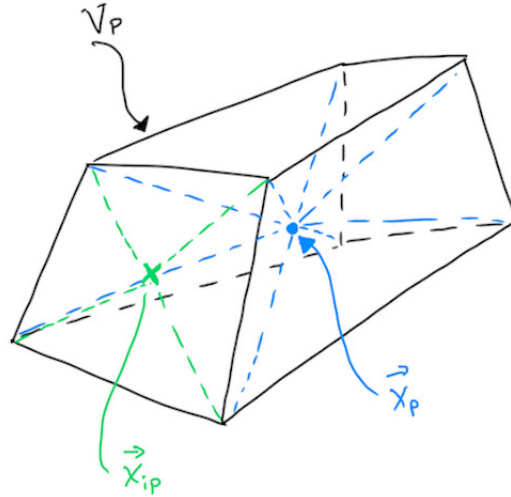


Figura 2.6: Representación de un volumen/celda genérica utilizada para el método de volumen finito.

analizar cada volumen del espacio teniendo en cuenta que a través de cada punto de integración se tiene un flujo hacia adentro o hacia afuera que puede ser calculado, tomando en cuenta los conceptos de conservación de masa, de momento y de energía. Con el concepto gráfico en mente se puede retomar que este método está basado en un sistema integral, lo que significa que dado un sistema de ecuaciones diferenciales parciales por resolver, se hará la resolución integrando dicho sistema sobre los volúmenes de control V_p , siendo muy utilizado para esto el teorema de la divergencia de Gauss, que permite pasar de integrales volumétricas a integrales de superficie, es decir se integrará sobre el contorno de dichos volúmenes, lo que se puede discretizar fácilmente pensando en que los volúmenes que se construyan tendrán un número finito de caras. En la referencia [?] se puede encontrar todo un desarrollo computacional de este método.

2.2.1. Algoritmo del Método del Volumen Finito

Es muy común que en la física un problema se vea representado mediante un sistema de ecuaciones diferenciales parciales que no se pueda resolver analíticamente, por lo que se recurre a métodos numéricos tales como el Método del Volumen Finito (MVF). A continuación se presentan una serie de pasos para resolver un problema de este tipo mediante el MVF, para más información consultar [?].

- **Discretización del espacio:** El primer paso del proceso de solución es la discretización del dominio geométrico, que para este método significa discretizar en elementos o volúmenes finitos (volúmenes de control) que no se traslapen entre sí. Dependiendo del dominio geométrico que se esté trabajando será la forma que tendrán dichos volúmenes y sus dimensiones. Como este método suele resolver problemas físicos reales se suelen usar volúmenes tridimensionales, pero no se limita a esta dimensionalidad. Aquí se suele tener un **mallado** del espacio en términos computacionales (véase Figura 2.7)
- **Etiquetado de los volúmenes:** Ya que el espacio se tiene dividido en volúmenes de control, se toma de cada uno de estos su centroide (centro geométrico), así como de cada cara de cada volumen, por ejemplo, para un cubo se tiene su centroide en el centro del

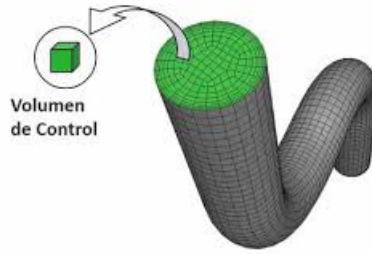


Figura 2.7: Ejemplo de un mallado computacional del espacio de estudio.

volumen, y cada cara tendrá a su vez un centroide asociado en su centro (veáse Figura 2.8).

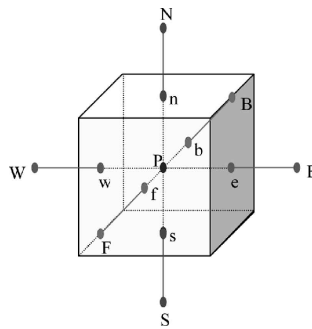


Figura 2.8: Centroides de un cubo y sus caras.

- **Integración sobre volúmenes:** Como se mencionó, este método sirve para resolver analíticamente sistemas de ecuaciones diferenciales parciales, tal como la que se muestra a continuación

$$\frac{\partial \phi}{\partial t} + \nabla \cdot (\mathbf{u}\phi) + \nabla \cdot \mathbf{J}_\phi = S_\phi \quad (2.17)$$

La ecuación 2.17 representa una ecuación de conservación genérica, tal como las que se tratan en el MVF, esto debido a que en este método siempre se considera conservación en el sentido en que algo que entra a un volumen de control es lo mismo que sale de otro volumen. Así, teniendo una ecuación de la forma 2.17 lo que se hace es integrar toda la ecuación sobre los volúmenes de control, lo que resulta en una ecuación integral de conservación. Al integrar se toman varias estrategias para poder discretizar como sumas, por ejemplo, es muy común utilizar para este método el teorema de la divergencia de Gauss que transforma una integral de volumen a una de superficie, y sumar sobre las caras del volumen de control, dependiendo de la dirección en que esté fluyendo algo a través de éstas. El resultado final de este proceso es una ecuación que exprese el equilibrio entre los términos de flujo superficial y los términos de fuente fuente de volumen. Este punto del algoritmo es el que lleva todo el entendimiento físico-matemático necesario para poder llevar de una manera exitosa el método. Las definiciones de conservación, flujo y teoremas matemáticos serán tomados a nuestro favor aquí.

- **Linearización:** Como un añadido al anterior punto se tiene que al discretizar el sistema de ecuaciones diferenciales parciales las soluciones suelen tener términos no lineales, lo que puede llevar a ciertas problemáticas, entonces lo que se hace es linealizar mediante el método de Newton, con lo que siempre se tendrán que obtener coeficientes de linealización. **Esto nos da un sistema de ecuaciones lineales algebraicas a resolver, que es la corrección de la solución.**
- **Condiciones de frontera:** Sabemos que un sistema de ecuaciones diferenciales tiene una infinidad de soluciones por su misma naturaleza, es por esto que se requiere que se fijen condiciones de frontera y/o condiciones iniciales. Se pueden tener tanto condiciones de frontera de la solución (Dirichlet) como de la derivada de la solución (Neumann).

2.3. Resumen

Capítulo 3

Implementación Computacional

No sé si aquí se pone lo que voy haciendo pero supongo que sí, sino le moveremos.

En principio se tomó una implementación hecha del método de volumen finito en python de un compañero. Vimos que servía con ejemplos de una dimensión del libro [6]. Para poder implementar nuestro modelo se necesitaba una actualización de esta implementación, ya que ésta no incluía una evolución temporal en el sistema. Para esto se tenía que la implementación ya no estaba del todo clara para problemas de dos dimensiones con los que fuera más fácil corroborar la evolución. Para esto se hicieron modificaciones al código ya hecho, que incluyen por un lado hacer las clases más sencillas y fáciles de entender, con menos dependencias de métodos incluidos en las clases y dándole un poco de estructura. Dado esto, se tuvo que cambiar todo el curso de la implementación, desde lo básico que representaba la difusión y advección. Los cambios que se están haciendo son respecto a los coeficientes de los nodos de cada volumen, hay un problema con las dimensiones y su arreglo, en una dimensión estaba bien, pero al intentar dejar todo más claro ya no se podía en una dimensión, el chiste es que todo esto quede generalizado hasta a tres dimensiones sin problemas. Para esto está el problema de que el volumen construido no contempla los puntos que rodean a la frontera, lo que se hace es agregarlos manualmente.

Esto de agregar los puntos manualmente funcionó muy bien, hasta dió la pauta para agregar el coeficiente S_P de manera muy sencilla. Con esto se pudieron probar ejemplos en una dimensión con datos uniformes y no uniformes dadas todas las modificaciones y graficación que se hizo.

El siguiente problema que surgió fue que no funcionaba en dos dimensiones, esto por el mal manejo de las deltas que van dividiendo en los coeficientes, se propone una redimensionalización de estas.

La redimensionalización no funcionará, o al menos no será generalizada, se propone hacer todo mediante rendijas o mallas y así conservar bien las dimensiones.

La estrategia de las rendijas funcionó bien pero se tuvieron que hacer varios cambios y ciertos métodos que no estaban contemplados. Falta ver si los coeficientes que se están obteniendo son correctos.

Al parecer el problema con los coeficientes en dos dimensiones es que cuando hay un flujo no se agrega directo como una fuente sino que se agrega al vector "b" del sistema, entonces hay que poner esta condición. Se comienza por separar cada coeficiente de las fuentes.

Fueron separados los términos de coeficientes y de fuentes, ahora hay que dar condiciones para cada caso, para cuando haya condición de frontera de dirichlet o de neumann.

Se realizó este proceso de especificar la fuente dada la condición de frontera, cuando se tiene una frontera aislada se hace cero, cuando se tiene una condición de dirichlet se hace

la cuenta, el siguiente paso será ver qué pasa cuando es un flujo, que ahí ya va directo a la solución de la matriz, junto con condición de dirichlet.

Se obtuvo también el valor de Su para toda la frontera tomando en cuenta cuando hay flujo, cuando hay temperatura y cuando está aislada la frontera, con esto es posible obtener el vector "b" de la ecuación $Ax = b$.

Se hace la matriz A y el vector b a partir de los coeficientes obtenidos con anterioridad, se puede obtener una solución del sistema a partir de esto.

El siguiente paso fue meter el término convectivo, esto haciendo una nueva clase `Convectionz` añadiendo esto a la clase que ya se tenía de coeficientes. Se prueba primero para una dimensión con los ejemplos dados por José del Malalasekera.

Ya que se tienen los términos de convectividad y difusividad se tiene el modelo más sencillo. El siguiente paso sería pasar a un modelo en que se toma en cuenta la presión (como un escalar) y la velocidad (vectorial), con lo que se ataca mediante el método SIMPLE y el mallado escalonado.

Se hace un mallado escalonado pensando en que se tiene que desplazar la malla que se ha creado en tantas direcciones como dimensiones tenga, puede ser hacia adelante o hacia atrás, aquí pensaremos en que sea hacia atrás.

Esta sección incluiría

- Características específicas de julia: alto nivel, tipos de datos, despacho múltiple.
- Paralelismo: modelos y paradigmas de programación paralela. Puedes tomar como referencia el trabajo: "Análisis comparativo del desempeño de patrones arquitectónicos para programación paralela" de Rodrigo Novelo Cervera. Tesis UNAM.
- Programación de GPU's y su lenguaje nativo CUDA: Kernel, bloques, `threadidx`, `host`, `device`.
- `CUDAnative.jl`, Resumen generoso del artículo de Tim Besard.
- Describir la estrategia de paralelismo: qué vas a paralelizar y por qué elegiste la GPU. Podrías incluir aquí los solvers, y explicar el método de MCCM.
- Como se va implementar en CUDA, usando `CUDAnative.jl`

3.1. Resumen

Capítulo 4

Experimentación(?)

¿Cuál es la duda?

En general aquí vas a exponer los resultados que obtuviste propios del problema físico y los resultados obtenidos variando el solver y parámetros del modelo matemático.

4.1. Planteamiento

Vas a explicar el experimento

4.2. ~~Implementación en Julia~~

4.3. ~~Implementación en Julia variando parámetros~~

4.4. Distintos solvers

La exposición de los solvers van en el capítulo 3, aquí explicarías la comparación que haces de estos solvers. La referencia de esto son los trabajos de MCMC de Alexandrov, Esquivel-Flores, Strassburg.

4.4.1. GMRES

4.4.2. BiC-STAB

4.5. Método de Monte Carlo - Cadenas de Markov

4.6. Resumen

Capítulo 5

Análisis de Resultados

Aquí va el análisis de los resultados:

- La congruencia de los resultados obtenidos respecto al problema físico
- La exactitud de los resultados cambiando parámetros y solver
- Speed up y escalabilidad
- Comentarios finales sobre la implementación Julia+CUAnative

5.1. Resumen

Capítulo 6

Conclusiones y Trabajo Futuro

Aquí mencionar:

- La vigencia e importancia de CFD
- La necesidad de HPC
- Las ventajas de Julia
- Las ventajas y desventajas de paralelismo con GPUs
- Los resultados obtenidos
- Comentarios sobre la metodología físico-computacional que propusiste en tu trabajo

Bibliografía

- [1] M. Aenis, A. P. Stancampiano, A. K. Wakhloo, and B. B. Lieber. Modeling of Flow in a Straight Stented and Nonstented Side Wall Aneurysm Model. Journal of Biomechanical Engineering, 119(2):206–212, 05 1997.
- [2] E. Finol and C. Amon. Blood flow in abdominal aortic aneurysms: Pulsatile flow hemodynamics. Journal of biomechanical engineering, 123:474–84, 10 2001.
- [3] F. Harlow. Fluid dynamics in group t-3 los alamos national laboratory (la-ur-03-3852). Journal of Computational Physics, 195:414–433, 04 2004.
- [4] T.-M. Liou and Y.-C. Li. Effects of stent porosity on hemodynamics in a sidewall aneurysm model. Journal of biomechanics, 41:1174–83, 02 2008.
- [5] T.-M. Liou, Y.-C. Li, and W.-C. Juan. Numerical and experimental studies on pulsatile flow in aneurysms arising laterally from a curved parent vessel at various angles. Journal of biomechanics, 40(6):1268—1275, 2007.
- [6] W. Malalasekera and H. Versteeg. An Introduction to Computational Fluid Dynamics. 2 edition, 2007.
- [7] L. F. Richardson and P. Lynch. Weather Prediction by Numerical Process. Cambridge Mathematical Library. Cambridge University Press, 2 edition, 2007.
- [8] W. Sven Oivind. Pulsatile pressure and flow in an arterial aneurysm simulated in a mathematical model. Journal of Biomedical Engineering, 3(2):153 – 158, 1981.
- [9] M. J. TURNER, R. W. CLOUGH, H. C. MARTIN, and L. J. TOPP. Stiffness and deflection analysis of complex structures. Journal of the Aeronautical Sciences, 23(9):805–823, 1956.
- [10] N. Viswanath, C. M. Rodkiewicz, and S. Zak’c. On the abdominal aortic aneurysms: pulsatile state considerations. Medical engineering & physics, 19 4:343–51, 1997.
- [11] J. Wendt. Computational Fluid Dynamics: An Introduction. A von Karman Institute book. Springer Berlin Heidelberg, 2008.

Índice alfabético

contribución, 5

hipotesis, 5

logros, 5

metas, 5

metodología, 4

motivación, 3

objetivo, 4

organización, 5