



IMPLEMENTAÇÃO DE UM SIMULADOR DE ESCOAMENTO DE FLUIDOS
EM JULIA USANDO MÉTODO DA SEPARAÇÃO DAS CURVAS
CARACTERÍSTICAS

Gabriel Felipe de Oliveira Antão

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Mecânica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Mecânica.

Orientador: Gustavo Rabello dos Anjos

Rio de Janeiro
Novembro de 2024

IMPLEMENTAÇÃO DE UM SIMULADOR DE ESCOAMENTO DE FLUIDOS
EM JULIA USANDO MÉTODO DA SEPARAÇÃO DAS CURVAS
CARACTERÍSTICAS

Gabriel Felipe de Oliveira Antão

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO
ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE
ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO
PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU
DE MESTRE EM CIÊNCIAS EM ENGENHARIA MECÂNICA.

Orientador: Gustavo Rabello dos Anjos

Aprovada por: Prof. Nome do Primeiro Examinador Sobrenome
Prof. Nome do Segundo Examinador Sobrenome
Prof. Nome do Terceiro Examinador Sobrenome
Prof. Nome do Quarto Examinador Sobrenome
Prof. Nome do Quinto Examinador Sobrenome

RIO DE JANEIRO, RJ – BRASIL
NOVEMBRO DE 2024

Antão, Gabriel Felipe de Oliveira

Implementação de um simulador de escoamento de fluidos em Julia usando método da separação das curvas características/Gabriel Felipe de Oliveira Antão. – Rio de Janeiro: UFRJ/COPPE, 2024.

XIII, 80 p.: il.; 29, 7cm.

Orientador: Gustavo Rabello dos Anjos

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia Mecânica, 2024.

Referências Bibliográficas: p. 75 – 80.

1. simulador. 2. Characteristic-Based Split. 3. CBS. 4. fluidos. 5. Julia. 6. WaveFE. I. dos Anjos, Gustavo Rabello. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Mecânica. III. Título.

Aos meus mestres

Agradecimentos

Agradeço aos meus pais Katia e Fernando e todos demais familiares por apoiarem minha caminhada acadêmica desde o princípio. Especial agradecimento à minha esposa Fúvia por ter dado todo suporte e compreensão durante o desenvolvimento deste trabalho.

Agradeço ao Programa de Engenharia Mecânica e principalmente ao meu orientador Gustavo Rabelo por terem me aceitado como aluno. O professor sempre enormemente compreensível e solícito, sempre dedicado a fazer seus alunos melhores e posso dizer que seus ensinamentos muito influenciaram na escolha da atividade profissional que exerço atualmente.

Quero aproveitar para de dedicar agradecimentos também aos colegas de trabalho na ESSS, em especial ao time de desenvolvedores científicos por todos os incentivos, dicas, e ensinamentos com os quais tenho sido agraciado nos últimos anos de convívio.

Por fim, devo agradecer ao CNPq por ter proporcionado condições financeiras para o desenvolvimento da pesquisa.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

IMPLEMENTAÇÃO DE UM SIMULADOR DE ESCOAMENTO DE FLUIDOS
EM JULIA USANDO MÉTODO DA SEPARAÇÃO DAS CURVAS
CARACTERÍSTICAS

Gabriel Felipe de Oliveira Antão

Novembro/2024

Orientador: Gustavo Rabello dos Anjos

Programa: Engenharia Mecânica

A simulação computacional é cada vez mais crucial na indústria porque permite aos engenheiros a experimentação de possíveis cenários para identificar as melhores formas de projetar seus produtos antes mesmo de que necessitem construir protótipos, o que reduz custos e tempo de desenvolvimento. Nesse contexto, diversas companhias desenvolveram soluções comerciais ao longo das últimas décadas para atender a essa necessidade da indústria. Muitos desses *softwares* desenvolvidos para simulação de problemas de fluidos fazem uso de técnicas de simulação baseados no Método dos Elementos Finitos (MEF) para modelar e solucionar numericamente problemas de interesse. Neste trabalho se apresenta o desenvolvimento do simulador WaveFE para tratar de problemas envolvendo escoamento de fluidos utilizando MEF, mais especificamente a técnica de *Characteristic-Based Split* (CBS) semi-implícito, para solução das equações governantes do fenômeno. Apesar de ser comum o uso do MEF nessas aplicações, há raríssimos exemplos de implementação de CBS disponíveis de código aberto para servir como um *framework* de base para novos desenvolvimentos, nesse sentido o WaveFE pretende ser uma implementação de base para desenvolvimentos futuros de modo que se possa facilmente experimentar novos modelos sem necessitar reescrever código base do CBS comum a todos métodos. Além da arquitetura do simulador, apresenta-se o seu uso em casos de simulação clássicos como do escoamento incompressível entre placas, caso da cavidade quadrada e caso do degrau que foram utilizados para validação e verificação dos resultados obtidos no simulador e apresentaram excelente concordância com resultados de referência, o que aumenta o nível de evidência de que o simulador possa ser utilizado futuramente para experimentos com modelos baseados em CBS.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

DEVELOPMENT OF FLUID FLOW SIMULATOR BASED ON
CHARACTERISTIC-BASED SPLIT METHOD

Gabriel Felipe de Oliveira Antão

November/2024

Advisor: Gustavo Rabello dos Anjos

Department: Mechanical Engineering

Computer simulation is increasingly crucial in the industry because it allows engineers to experiment with possible scenarios to identify the best ways to design their products even before they need to build prototypes, which reduces costs and development time. In this context, several companies have developed commercial solutions over the last few decades to meet this industry need. Many of these *software* developed for simulating fluid problems make use of simulation techniques based on the Finite Element Method (FEM) to model and numerically solve problems of interest. This work presents the development of the WaveFE simulator to deal with problems involving fluid flow using FEM, more specifically the semi-implicit Characteristic-Based Split (CBS) technique, for solving the governing equations of the phenomenon. Although the use of MEF in these applications is common, there are very few examples of CBS implementation available in open source to serve as a base framework for new developments, in this sense WaveFE intends to be a base implementation for future developments so that you can easily try new models without having to rewrite the CBS base code common to all methods. In addition to the architecture of the simulator, its use is presented in classic simulation cases such as incompressible flow between plates, the square cavity case and the step case, which were used to validate and verify the results obtained in the simulator and showed excellent agreement with reference results. , which increases the level of evidence that the simulator can be used in the future for experiments with CBS-based models.

Sumário

Lista de Figuras	x
Lista de Tabelas	xiii
1 Introdução	1
1.1 Apresentação do problema	1
1.2 Diretrizes do projeto	2
2 Revisão bibliográfica	4
2.1 Histórico do desenvolvimento de soluções de problemas de fluidos . .	4
2.2 Problemas modelados com uso de CBS	5
3 Formulação das equações	18
3.1 Modelagem do problema de convecção-difusão	18
3.2 Método das características	19
3.3 Separação das curvas características (CBS)	23
3.4 Formas adimensionais das equações	25
3.5 O método da separação das características generalizado	27
3.6 Discretização espacial das equações	30
3.7 Equações discretizadas para a separação B	36
3.8 Diagonalização da matriz de massa	36
3.9 Forma explícita e semi-implícita	37
3.10 Condições de contorno	38
4 Arquitetura do simulador	40
4.1 Ferramentas utilizadas para o projeto	40
4.2 Fluxo de execução no simulador	42
4.3 Estrutura do código	44
4.4 Arquivos de entrada	45
4.5 Arquivos de saída	47
4.6 Testes unitários e testes de caso	48

5	Resultados	49
5.1	Montagem dos experimentos	49
5.2	Caso do escoamento entre placas	50
5.3	Caso da cavidade quadrada	52
5.4	Caso do degrau	58
5.5	Caso do obstáculo quadrado	66
5.6	Caso do banco de tubos	67
6	Considerações finais	70
6.1	Conclusões	70
6.2	Trabalhos futuros	71
6.2.1	Implementação de novos tipos de malha	71
6.2.2	Implementação de novos modelos	72
6.2.3	Variantes do método CBS	73
6.2.4	Melhorias de performance do simulador	73
	Referências Bibliográficas	75

Lista de Figuras

2.1	Linhas de fluxo para o caso do obstáculo quadrado para $Re=100$ dado por PEREIRA <i>et al.</i> [1]. tempos adimensionais (a) 1 (b) 30 (c) 75 . . .	6
2.2	Escoamento do fluxo ao redor de tubos MASSAROTTI <i>et al.</i> [2].	7
2.3	Linhas de fluxo do escoamento $Re=150$ entre tubos MASSAROTTI <i>et al.</i> [2].	7
2.4	Fluxo das vias aéreas humanas superiores LIU [3].	8
2.5	Contorno de formação de vórtices WANG e ZHANG [4].	9
2.6	Linhas de fluxo de um Vórtice não estacionário WANG e ZHANG [4].	9
2.7	Velocidade e linha de fluxo do escoamento em cavidades com tampa a $Re= 400$ JIN e SHEN [5].	11
2.8	Linhas de fluxo e vetores de velocidade do escoamento secundário em um tubo TAN <i>et al.</i> [6].	12
2.9	Convergência espacial da altura da superfície livre para o problema de propagação de ondas longas. BUACHART <i>et al.</i> [7].	13
2.10	Caso do perfil de profundidade da água. BUACHART <i>et al.</i> [7].	13
2.11	Esquema físico de nano-fuidos cobre e água de MINH TUAN NGUYEN e LEE [8].	14
2.12	Limites da pastilha de chip, analisado por KULKARNI <i>et al.</i> [9].	15
2.13	Padrão de velocidade KULKARNI <i>et al.</i> [9].	15
2.14	Soluções de LI e DUAN [10] para o problema do escoamento em cavidades com tampa com $Re = 1000$, utilizando CBS para estabilização de pressão. (a) Linhas de fluxo, CBS, $\Delta t = 0.1 \Delta t_c$. (b) Contornos de pressão, CBS, $\Delta t = 0.1 \Delta t_c$	16
2.15	posições das superfícies livres em movimento e das malhas correspondentes ao nível de tempo $t = 3,2$ s no estudo de DUAN e LI [11]	17
3.1	Representação gráfica da característica usada no método.	20
4.1	Fluxo de execução simplificado do simulador.	43

5.1	Malha utilizada no caso do escoamento entre placas. 3562 elementos triangulares e 1780 nós.	50
5.2	Representação das condições de contorno para o caso do canal. Comprimentos L adimensionais.	50
5.3	Velocidade na direção horizontal $x = 15$ para $Re = 100$	51
5.4	Comparação dos valores obtidos com a referência para velocidades na direção horizontal em $x = 15$ para $Re = 100$	51
5.5	Malha utilizada no caso da cavidade quadrada. 5000 elementos triangulares e 2601 nós.	53
5.6	Representação das condições de contorno.	54
5.7	Mapa de velocidade na direção horizontal u_1 para $Re = 1000$	54
5.8	Mapa de velocidade na direção vertical u_2 para $Re = 1000$	55
5.9	Velocidade na direção horizontal na linha central $x = 0,5$ para $Re = 1000$	56
5.10	Velocidade na direção vertical na linha central $y = 0,5$ para $Re = 1000$	57
5.11	Malha utilizada no caso do degrau. 6375 elementos triangulares e 3137 nós.	58
5.12	Representação das condições de contorno para o caso do degrau. Comprimentos L adimensionais.	58
5.13	Mapa de velocidade da velocidade na direção horizontal u_1 para $Re = 229$	59
5.14	Mapa de velocidade da velocidade na direção vertical u_2 para $Re = 229$	60
5.15	Mapa de velocidade da pressão p para $Re = 229$	60
5.16	Comparação dos valores obtidos pelo simulador em comparação com os dados experimentais obtidos por LIU [3] na posição $x = 4.0$	61
5.17	Comparação dos valores obtidos pelo simulador em comparação com os dados experimentais obtidos por LIU [3] na posição $x = 4.88$	62
5.18	Comparação dos valores obtidos pelo simulador em comparação com os dados experimentais obtidos por LIU [3] na posição $x = 6.11$	63
5.19	Comparação dos valores obtidos pelo simulador em comparação com os dados experimentais obtidos por LIU [3] na posição $x = 8.17$	64
5.20	Comparação dos valores obtidos pelo simulador em comparação com os dados experimentais obtidos por LIU [3] na posição $x = 14.29$	65
5.21	Malha utilizada no caso do obstáculo quadrado. 7114 elementos triangulares e contém 3553 nós.	66
5.22	Condições de contorno impostas para o caso do obstáculo quadrado.	66
5.23	Detalhe das linhas de fluxo para o caso do obstáculo quadrado para $Re = 100$. Contorno do objeto omitido para melhor visualização do padrão de linhas.	67

5.24	Malha utilizada no caso do banco de tubos. 9984 elementos e 4993 nós.	67
5.25	Representação das condições de contorno para o caso banco de tubos. Comprimentos L adimensionais.	68
5.26	Mapas de velocidade horizontal para o caso do banco de tubos com Re=150.	69
5.27	Mapas de velocidade vertical para o caso do banco de tubos com Re=150.	69
5.28	Mapas de pressão para o caso do banco de tubos com Re=150.	69

Lista de Tabelas

5.1	Valores dos desvios em cada posição horizontal medida com relação ao apresentado em LIU [3].	60
-----	--	----

Capítulo 1

Introdução

1.1 Apresentação do problema

No âmbito da Engenharia e da Ciência da Computação, a simulação de escoamento de fluidos se destaca como uma ferramenta crucial para compreender e prever o comportamento de líquidos e gases em diversos cenários. Essa área do conhecimento tem um papel fundamental em uma ampla gama de indústrias, desde o desenvolvimento de aeronaves e turbinas eólicas até a análise de processos petroquímicos e o estudo de fluxos sanguíneos.

Os métodos tradicionais de simulação de escoamento de fluidos baseados no Método dos Elementos Finitos (MEF) ou Método das Diferenças Finitas (MDF) demonstram ter grande utilidade e muitas variações de técnicas de modelagem e solução continuam surgindo durante as últimas décadas para aperfeiçoar os métodos já existentes ou propor novos. A busca por maior precisão, eficiência e robustez impulsiona o desenvolvimento dessas novas técnicas, sendo uma delas a técnica Characteristic-Based Method (CBS) que se apresenta como uma alternativa promissora. Algumas das características do método CBS que fazem dele uma forma atraente de ser usado como base para implementação de um simulador:

1. baseado em características: em vez de resolver diretamente as equações diferenciais parciais que governam o fluxo, o CBS acompanha as características do fluxo, proporcionando uma representação mais natural e intuitiva do escoamento;
2. preciso: o CBS demonstra maior precisão em capturar fenômenos complexos, como choques e ondas, em comparação com métodos tradicionais;
3. computacionalmente eficiente: o CBS pode ser implementado de forma mais eficiente, exigindo menos recursos computacionais para alcançar o mesmo nível de precisão de outros métodos;

4. robusto e adaptável: o CBS apresenta maior robustez em lidar com geometrias complexas e condições de contorno desafiadoras, além de se adaptar dinamicamente às características do fluxo.

Apesar de haver uma vasta gama de simuladores comerciais há poucas descrições detalhadas sobre o uso dessa técnica nesses simuladores, e mesmo que houvesse, normalmente esses softwares possuem código fechado o que impossibilita a compreensão de como são implementados os detalhes do programa. Além disso, o fato de o código ser proprietário e fechado dificulta ou mesmo proíbe que o usuário defina facilmente seus próprios modelos experimente variantes de métodos de solução.

Os principais objetivos deste projeto de dissertação residem em:

1. desenvolver um novo simulador de escoamento de fluidos baseado no CBS: este simulador irá explorar o potencial do CBS para servir como uma ferramenta de base para desenvolvimentos futuros que usem esta técnica de solução;
2. garantir que seja uma ferramenta precisa, eficiente e robusta para a simulação de diversos cenários de escoamento de fluidos;
3. fornecer a possibilidade de que se desenvolvam novas pesquisas em simulação de escoamento de fluidos a partir do uso do CBS, impulsionando o conhecimento científico nesta área, abrindo caminho para novas descobertas e aplicações.

As seguintes seções deste trabalho apresentam o WaveFE ¹, um simulador baseado em CBS implementado em linguagem Julia para solução de problemas de escoamento de fluidos que poderá ser usado como *framework* para desenvolvimento de experimentos computacionais e como base para implementação de outros métodos de solução e novos modelos.

1.2 Diretrizes do projeto

Algumas diretrizes foram elencados de modo a nortear o processo de construção do simulador:

1. simplicidade do código
2. rápida execução das simulações
3. modularidade
4. testabilidade

¹<https://github.com/gabrielantao/WaveFE>

A simplicidade e legibilidade do código foi uma das principais diretrizes do projeto e que indicaram para o uso da linguagem Julia para tal implementação. Essa diretriz visa a facilitar a compreensão futura de pesquisadores que queiram usar o simulador como material de estudo com o objetivo de compreender mais sobre a implementação de solução de problemas por MEF, especialmente com uso do CBS, e possam experimentar com o código como queiram. Além desses usuários, os mais experientes se beneficiam da simplicidade e legibilidade do código, já que podem implementar suas próprias adaptações, modelos, métodos de solução, etc. com maior facilidade podendo ter foco somente no problema físico e sua modelagem em si e não com detalhes de implementação ou sintaxe.

A segunda diretriz faz referência ao desejo que o simulador não somente seja capaz de gerar o resultado correto, mas também que o faça com o menor consumo de recursos da máquina onde é executado e no menor tempo possível. O baixo tempo de execução foi, portanto, uma das diretrizes definidas para que o programa pudesse rodar casos com um tempo aceitável para os casos de teste selecionados.

Uma outra propriedade definida como diretriz foi a modularidade do código, para que fosse possível serem testadas as partes isoladas do programa, assim como garantir melhor reaproveitamento de código. Além da facilidade de testar, a modularidade tende a facilitar a extensão do código com novas funcionalidades sem ter que alterar as partes que já estão implementadas [12]. Essa característica é fundamental, por exemplo, para que se possa acrescentar novos modelos sem ter que modificar código já implementado no núcleo do programa.

Outra característica almejada neste projeto foi a capacidade de realizar testes automáticos para todos os segmentos do programa (i.e. testes unitários) e testes de caso de simulação. O alto número de testes unitários apoiam não somente a garantia do nível de evidência da correção das funções, como a metodologia de testes propõe [12], como também servem de documentação do código, uma vez que os testes indicam como montar as entradas para determinada função e portanto serve como consulta de como usa-las.

O trabalho apresentado após este Capítulo 1 de breve introdução ao tema, se segue do Capítulo 2 onde é apresentada uma revisão sobre o método e suas diversos trabalhos com aplicações e resultados obtidos, alguns dos quais usados como referência para validação do simulador. O Capítulo 3 mostra a formulação do modelo a ser resolvido pelo simulador e a ideia geral da estratégia do método para solução dos problemas. O Capítulo 4 discute características de implementação e a estrutura geral do simulador. Já o Capítulo 5 apresenta resultados de validação do simulador, comparando-os com dados de referência de trabalhos previamente apresentados no Capítulo 2. Finalmente o Capítulo 6, encerra este texto com conclusões e sugestões implementações de extensões deste trabalho .

Capítulo 2

Revisão bibliográfica

2.1 Histórico do desenvolvimento de soluções de problemas de fluidos

Devido à presença de não-linearidade no termo convectivo da equação de Navier-Stokes, torna-se difícil simular o problema de escoamento com uma acurácia e tempo de processamento aceitáveis. Além disso, é fato conhecido que há forte dependência das condições de contorno e iniciais na solução, e ainda que pequenas perturbações (e.g. erros numéricos) causam grandes variações nas grandezas de interesse ao longo do tempo, essa situação acaba frequentemente comprometendo completamente o resultado da simulação [13].

Algumas estratégias foram desenvolvidas ao longo do tempo como forma de contornar o problema da instabilidade numérica ocasionada pela presença do termo convectivo da equação de Navier-Stokes. Por exemplo, BROOKS e HUGHES [14] propuseram o método de *Streamline Upwind Petrov-Galerkin* (SUPG) que utiliza uma generalização do esquema de *streamline upwind* combinado com a técnica de resíduos ponderados de Petrov-Galerkin para obter soluções acuradas para problemas multidimensionais.

Outra solução foi introduzida por DONEA [15] que propôs o uso do método conhecido como Taylor-Galerkin (TG) que emprega uma expansão em séries de Taylor adiantadas no tempo e inclui derivadas temporais de segunda e terceira ordem substituídas na equação de Navier-Stokes gerando uma forma de generalizada de discretização no tempo. LÖHNER *et al.* [16] apresentaram o método *Characteristic-Based* (CG) que é equivalente ao TG para o caso de variáveis escalares. A estratégia desse último método forma a base para o método *Characteristic Based Split* (CBS) que é utilizado neste projeto.

Todos os métodos citados têm objetivo de remover as instabilidades numéricas da solução do problema introduzidas pelo termo convectivo, porém, segundo ZIEN-

KIEWICZ e TAYLOR [17], o método CBS é o mais genérico e permite solucionar uma extensa quantidade de problemas físicos o que, segundo os autores, torna o método preferível em detrimento dos demais.

ZIENKIEWICZ e CODINA [18] introduziram o método CBS que consiste essencialmente em resolver a equação de Navier-Stokes em três etapas removendo a pressão da equação do momento e calculando uma velocidade intermediária e posteriormente usando essa velocidade intermediária para se calcular o campo de pressão. Com o campo de pressão é finalmente possível calcular o valor da velocidade real do próximo passo da iteração. Essa abordagem fornece a vantagem de que, sem o termo da pressão, a equação do momento se resume a uma equação de convecção-difusão tratável pelo método tradicional CG. Outro ponto positivo é que essa técnica permite uma melhora na estabilidade no cálculo da pressão e com isso possibilita a escolha arbitrária de funções de interpolação, tanto para a pressão quanto para a velocidade [19].

Em seguida a esse trabalho, ZIENKIEWICZ *et al.* [20] apresentaram resultados satisfatórios do método CBS na forma explícita para casos de fluido pouco compressível em alta velocidade utilizando aproximações lineares e de ordem quadrática para os elementos. Por fim, CODINA *et al.* [21] encerraram a série de publicações iniciais sobre o método CBS desenvolvendo a versão semi-implícita com uso de passos fracionais para fluidos incompressíveis e resolveram o problema da cavidade e o escoamento sobre o aerofólio NACA0012 obtendo resultados comparáveis aos experimentais.

Nas seções seguintes são apresentados alguns trabalhos de áreas distintas que mostram algumas das aplicações do CBS para resolução de algumas classes de problemas de escoamento de fluidos com transferência de calor, turbulência, escoamento em meios porosos e problemas com malhas móveis.

2.2 Problemas modelados com uso de CBS

Vários trabalhos apresentam resultados de problemas de transferência com modelagem e solução baseados em CBS. PEREIRA *et al.* [1] trazem resultados de testes para escoamento de fluido dentro de um canal com obstáculo quadrado cujas linhas de fluxo estão apresentadas na Figura 2.1, cuja superfície encontra-se com maior temperatura em relação ao resto do domínio. Em seguida fizeram testes semelhantes ao cenário com quadrado, mas desta vez em uma aplicação em que há como obstáculos uma série de tubos perpendiculares à direção do escoamento, com o objetivo de simular uma geometria de trocador de calor.

MASSAROTTI *et al.* [2] usou o CBS semi-implícito para resolver alguns casos envolvendo convecção natural com escoamento laminar com um dos contornos do

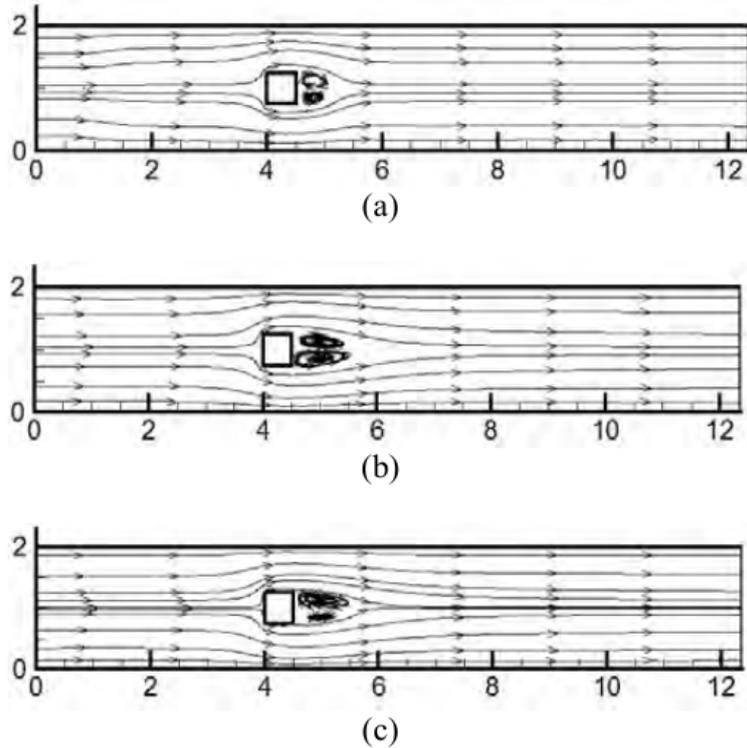


Figura 2.1: Linhas de fluxo para o caso do obstáculo quadrado para $Re=100$ dado por PEREIRA *et al.* [1]. tempos adimensionais (a) 1 (b) 30 (c) 75

domínio com temperatura $T = 1$ e temperatura inicial no restante do domínio $T = 0$. Ele simulou o caso da cavidade quadrada considerando a lateral esquerda com $T = 1$; o caso do degrau com contorno inferior sendo $T = 1$; e o caso do escoamento em torno da esfera com $T = 1$. Em seguida ele aplicou o método em um problema de convecção forçada semelhante àquele simulado por PEREIRA *et al.* [1]. Ele reportou excelente acurácia do método para esse caso, se comparados com resultados obtidos experimentalmente. Ele também simulou o escoamento em torno de tubos, semelhantes aos de um trocador de calor, com geometria da mostrada na Figura 2.2 e cujo resultado para as linhas de fluxo são apresentados na Figura 2.3, com destaque para regiões de recirculação de fluido a jusante dos tubos e outra maior após o último tubo.

Desde CHORIN [22] tem-se tentado usar métodos de separação e métodos de projeção para solução das equações de escoamento incompressível a partir do uso de um algum artifício de incompressibilidade artificial. F. ARPINO e NITHIRASU [23] apresentaram uma versão do método com compressibilidade artificial (AC) para a solução de problema de escoamento através de um meio poroso saturado. Neste trabalho, a técnica foi usada pela primeira vez para resolver as equações generalizadas de fluxo de meio poroso para problemas de convecção forçada. Eles checaram com sucesso o resultado após a convergência no estado estacionário com

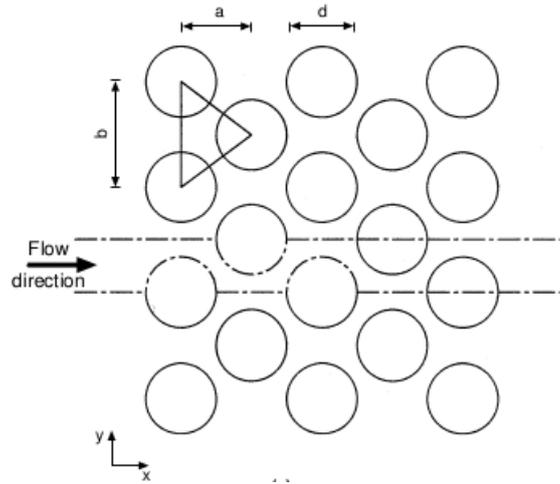


Figura 2.2: Escoamento do fluxo ao redor de tubos MASSAROTTI *et al.* [2].



Figura 2.3: Linhas de fluxo do escoamento $Re=150$ entre tubos MASSAROTTI *et al.* [2].

soluções numéricas e analíticas disponíveis de trabalhos anteriores. Uma das vantagens reportadas no uso do método foi a vantagem do esquema AC-CBS em permitir paralelização da solução. Essa característica dessa abordagem permitiu tempos significativamente menores comparado-se as soluções obtidas com AC-CBS àquelas obtidas com o método semi-implícito. Segundo os autores, a abordagem se mostrou robusta o suficiente para tratar problemas de convecção forçada e convecção natural através de meios porosos para escoamentos incompressíveis.

CORTELLESSA *et al.* [24] desenvolveram uma versão bifásica do método de AC-CBS com objetivo de estudar pela primeira vez fenômenos de transferência de massa em meios porosos com mudança de fase associada. O modelo formulado por eles serviu como alternativa para a análise teórica e simulação numérica de transporte multifásico fenômenos em meios porosos evitando-se tratamento por modelos baseados em escoamentos de fases separadas. A abordagem utilizada baseada no modelo bifásico, proporcionou uma nova observação para a compreensão de casos complicados de fluxos bifásicos em meios porosos. Como resultado, reduziu as equações diferenciais determinantes quase para metade, quando comparadas com o modelo clássico de escoamento separado, tornando-o adequado para simulações numéricas, mesmo com poucos recursos computacionais.

SCHREFLER [25] usaram um procedimento baseado em CBS para um meio po-

roso elástico deformável saturado, no qual a velocidade do fluido é governado pela equação de Darcy. Neste trabalho o modelo matemático é totalmente acoplado e avaliou-se o escoamento incompressível dentro de um meio com grãos sólidos incompressíveis. Eles verificaram a implementação do modelo comparando com soluções analíticas e numéricas disponíveis e obtiveram bons resultados e ainda demonstram que o procedimento elimina muitas das oscilações presentes em outros tipos de solução.

LIU [3] trouxe em sua tese uma abordagem fazendo uso do esquema AC-CBS e semi-implícito associado à abordagem de *Reynolds Average Navier-Stokes* (RANS) para tratar do problema do escoamento turbulento incompressível com o modelo de turbulência Spalart-Allmaras. Ele acusou excelente precisão em ambos os esquemas de CBS para versões do problema da caixa quadrada, degrau e escoamento em torno do cilindro (problemas tridimensionais) tendo boa concordância para o escoamento de fluxo constante na entrada. O coeficiente de arrasto médio e o coeficiente de sustentação também convergiram para o resultado esperado por simulações anteriores. Um modelo mais complexo envolvendo o fluxo em vias aéreas humanas superiores também foi simulado com sucesso, cujo resultado está apresentado na Figura 2.4 demonstrando a capacidade do CBS de lidar com esse tipo geometria.

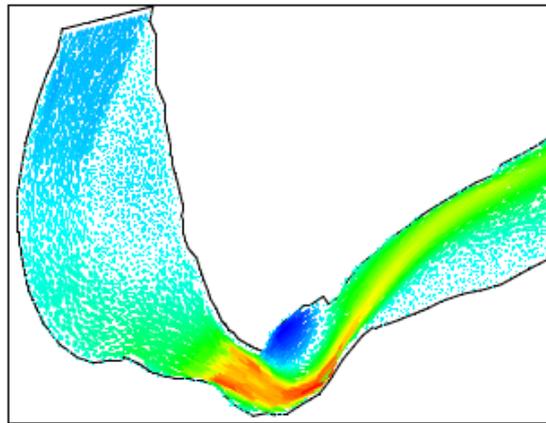


Figura 2.4: Fluxo das vias aéreas humanas superiores LIU [3].

Um esquema usando *Arbitrary Lagrangian–Eulerian* combinado com CBS, ALE-CBS, foi proposto inicialmente por NITHIARASU [26] para resolver o problema de escoamento de superfície livre, que envolve algum nível de movimentação da malha. Ele utilizou o esquema CBS para solucionar as equações de Navier-Stokes na formulação ALE e uma estratégia de suavização Laplaciana para tratar as distorções na malha. O autor utilizou o esquema explícito com compressibilidade artificial para solução dos problemas de validação da quebra da barragem (*broken dam*) e o problema da propagação da onda numa cavidade (*solitary wave propagation*) em duas dimensões para um fluido incompressível. Os resultados mostraram boa concor-

dância com os resultados experimentais, embora o autor sugira que outros métodos sejam usados no caso maiores amplitudes de onda.

NOBARI e NADERAN [27] e NOBARI e GHAZANFARIAN [28] investigaram o escoamento de um fluido incompressível em torno de um cilindro oscilando através da técnica de ALE-CBS. Nos casos simulados foi analisado o comportamento de formação de vórtices, no primeiro caso para cilindro oscilando e, no segundo, para o cilindro rotacionando em torno de seu eixo central. Em ambos os casos o método CBS foi empregado na sua forma semi-implícita para o problema bidimensional.

Posteriormente WANG e ZHANG [4] simularam o escoamento em torno de um cilindro oscilando do ponto de vista da dinâmica não linear de formação dos vórtices, como apresentado nas Figuras 2.5 e 2.6. Eles fizeram uso de um esquema semi-implícito para um fluido incompressível e encontraram boa concordância com resultados experimentais e numéricos previamente publicados, mostrando que o método resolve bem o problema do escoamento em torno de corpos em movimento. Além dos pontos discutidos, também tiveram efeitos positivos no padrão de escoamento dos vórtices cilíndricos, investigado por WANG e ZHANG [4], analisados do ponto de vista da dinâmica não linear, e considerando as regras de topologia para a relação entre nós e selas (incluindo semi-selas). Indicando assim, que o método numérico proposto pode ser utilizado para a análise da estabilidade da dinâmica não linear de problemas complexos de interação fluido-objeto. Já BAO *et al.* [29] utilizaram o método para avaliar o padrão de formação de vórtices escoamento ao redor de dois cilindros, um estacionário e um que oscila transversalmente ao escoamento.



Figura 2.5: Contorno de formação de vórtices WANG e ZHANG [4].

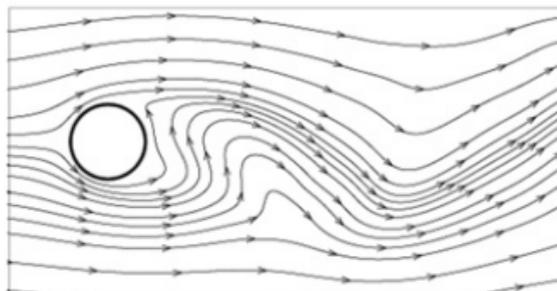


Figura 2.6: Linhas de fluxo de um Vórtice não estacionário WANG e ZHANG [4].

SUN *et al.* [30] apresentou um esquema alternativo ao tradicional ALE-CBS para fluidos incompressíveis e viscosos. Eles associaram o esquema modificado com o método de atualização de malha proposto por BLOM [31] a fim de remover a influência da velocidade da malha que aparece no esquema tradicional. Os problemas de validação de quebra da barragem e da oscilação do cilindro em duas dimensões foram resolvidos com sucesso, indicando a viabilidade do uso do método ALE-CBS proposto para problemas com malha móvel. De acordo com a análise de SUN *et al.* [30], os regimes CBS para malhas fixas puderam ser aplicados mais facilmente a problemas de malha móvel. Obtendo-se resultados satisfatórios quando o esquema CBS proposto é aplicado ao problema do escoamento em torno de um cilindro oscilante. O esquema estudado apresentou uma precisão superior à do que o método ALE-CBS existente.

WANG *et al.* [32] utilizou o método dos elementos finitos suavizados baseados em células (CS-FEM), com base no modelo de fase discreta (DPM), associados com o método semi-implícito CBS para o estudo da fase fluida, que por sua vez, foi empregado para reduzir as oscilações de espaço e pressão resultantes da solução numérica das equações de Navier-Stokes discretizadas pelo CS-FEM. A associação desses métodos foi utilizada para resolver movimentos de partículas sólidas diluídas induzidas por um escoamento laminar incompressível. Assim, para captar com precisão a velocidade do fluido numa posição arbitrária da partícula dentro de elementos quadrilaterais, foi introduzida a interpolação de coordenadas de valor médio. Para assegurar alta precisão na simulação das trajetórias das partículas, empregaram o método numérico de Runge-Kutta de quarta ordem na resolução das equações de movimento. Através de diversos estudos de caso, verificaram a eficácia do método proposto em prever a influência do escoamento sobre as partículas, tanto em termos de trajetória quanto de distribuição espacial, em problemas complexos de dinâmica dos fluidos computacional (CFD) utilizando o método CBS.

No estudo de JIN e SHEN [5], foi elaborado um novo método de elementos finitos para simular escoamentos por convecção mista em cavidades retangulares bidimensionais, analisando também padrões de velocidade em problemas complexos, como se mostra na Figura 2.7. O algoritmo proposto, baseado em CBS, permite uma eficiente separação das equações de Navier-Stokes, facilitando a resolução numérica, a utilização do método do gradiente conjugado pré-condicionado para a parte difusiva e do método CBS para a parte convectiva garante a precisão e a estabilidade nos objetivos do desenvolvimento e resultados. A validação do modelo foi realizada através de simulações de escoamentos em cavidades, demonstrando a eficácia da abordagem proposta com o uso do CBS.

CODINA e ZIENKIEWICZ [33] comparou dois métodos de estabilização para a aproximação por elementos finitos das equações de Navier-Stokes incompressíveis, o

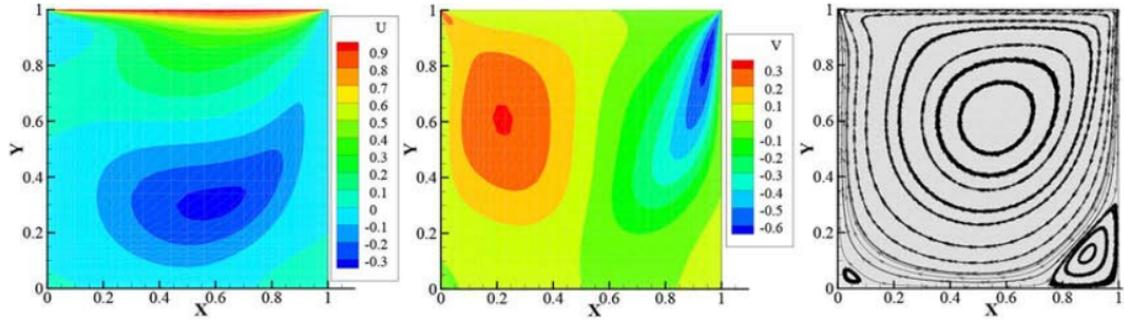


Figura 2.7: Velocidade e linha de fluxo do escoamento em cavidades com tampa a $Re= 400$ JIN e SHEN [5].

método de divisão das equações baseado em características (CBS) combinado com o método de Galerkin e o método de Galerkin de mínimos quadrados (GLS). O método CBS, empregado para lidar com escoamentos dominados por convecção, utiliza uma técnica de divisão clássica que, em certos casos, permite o uso de interpolações iguais para velocidade e pressão. O método GLS, por sua vez, adiciona um termo de estabilização baseado em mínimos quadrados às equações de Galerkin. Apesar de suas formulações distintas, uma análise de Fourier do problema linearizado revelou que ambos os métodos possuem mecanismos de estabilização semelhantes, desde que o parâmetro de estabilização do GLS seja identificado com o passo de tempo implícito na abordagem CBS.

Outra abordagem na utilização do método CBS, foi a analisada por TAN *et al.* [6] para a resolução de um método numérico de elementos finitos, baseado na combinação de LES com o esquema CBS, foi desenvolvido para prever escoamentos turbulentos. O método CBS foi aplicado para discretizar a equação de movimento no domínio temporal, seguindo as linhas características, enquanto o domínio espacial foi discretizado por um algoritmo de divisão, permitindo o cálculo separado da velocidade e da pressão. Diversas simulações numéricas foram realizadas com escoamento turbulento em uma cavidade cúbica e em dois tubos de seção circular. Os resultados obtidos foram comparados com dados experimentais e simulações numéricas diretas, validando o modelo proposto. De maneira singular, no tubo de seção circular de 90° , foi identificada uma estrutura de quatro vórtices, com um par adicional de vórtices próximo à parede interna da curva, não relatada em estudos anteriores. Os resultados teóricos demonstraram uma elevada aderência aos resultados obtidos. A análise das linhas de fluxo da velocidade, analisadas neste estudo, são encontradas na Figura 2.8.

O CBS foi combinado com o Método dos Elementos Espectrais (SEM) pelo estudo de YE *et al.* [34]. Esta associação, de início limitada por instabilidades numéricas causadas pelo *aliasing*, resultou em uma nova abordagem, baseada na

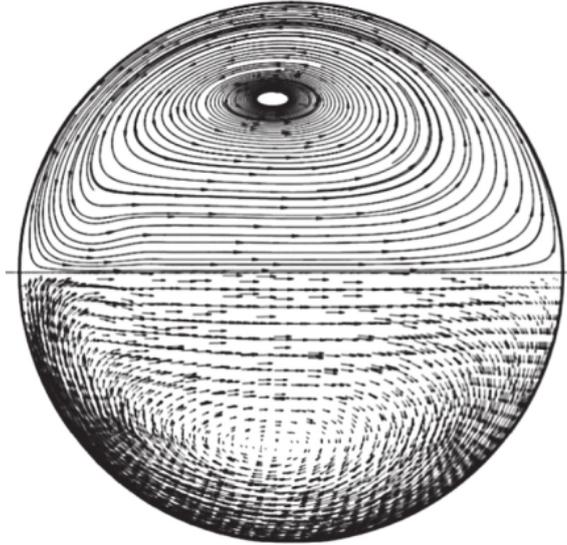


Figura 2.8: Linhas de fluxo e vetores de velocidade do escoamento secundário em um tubo TAN *et al.* [6].

sobre-integração, para estabilizar o método CBS-SEM. A sobre-integração permitiu uma aproximação mais precisa dos termos não lineares, reduzindo o erro de *aliasing* e expandindo o intervalo de estabilidade. Através de simulações numéricas, a eficácia da técnica proposta foi comprovada em diversos cenários.

O método CBS também foi utilizado como referência na comparação de resultados analíticos por BUACHART *et al.* [7], no desenvolvimento de um novo método numérico baseado em pontos finitos para simular escoamentos em águas pouco profundas, encontrado na Figura 2.10, utilizando uma nuvem global de pontos e um esquema iterativo robusto, este estudo obteve aproximações precisas para as funções de forma e gradientes, que foram incorporadas a solucionadores de Riemann. Os resultados numéricos demonstraram eficiente concordância com soluções analíticas e experimentais. A abordagem proposta mostrou-se particularmente eficaz em lidar com a complexidade dos termos fonte, como o declive da batimetria e os termos de cisalhamento da base, que frequentemente causam instabilidades em outros métodos.

CELIK e EDIS [35] conduziu uma análise numérica detalhada de escoamentos em microcondutos com degraus invertidos, considerando condições de deslizamento e salto de temperatura nas paredes. Aplicando o método CBS, simularam-se 81 casos distintos, abrangendo uma ampla gama de condições de escoamento. Os resultados obtidos permitiram estabelecer correlações entre os parâmetros adimensionais do escoamento e as taxas de transferência de calor e massa, fornecendo informações valiosas para o projeto de dispositivos microfluídicos. A análise demonstrou a capacidade do método numérico CBS em prever com precisão o comportamento complexo desses escoamentos, permitindo otimizar o desempenho de dispositivos microfluídicos para atender a requisitos específicos, como maximizar a transferência

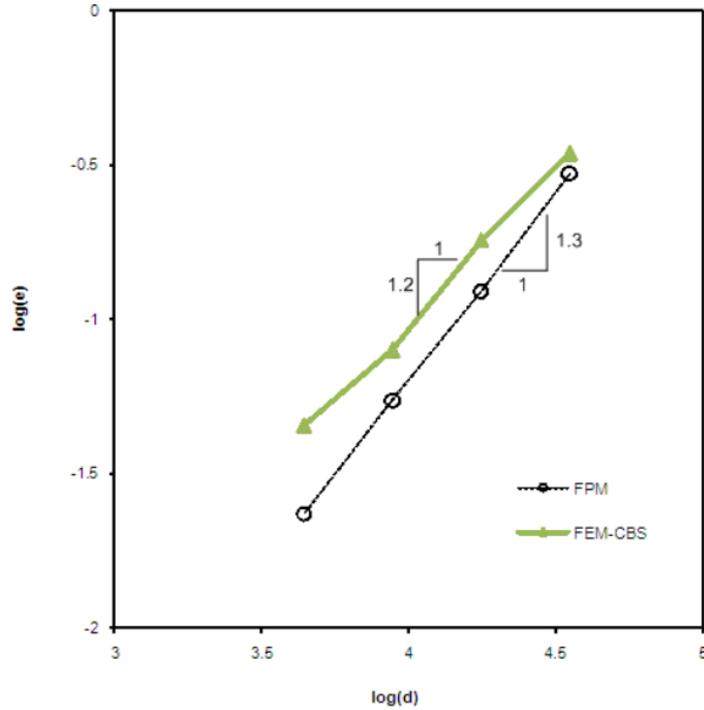


Figura 2.9: Convergência espacial da altura da superfície livre para o problema de propagação de ondas longas. BUACHART *et al.* [7].

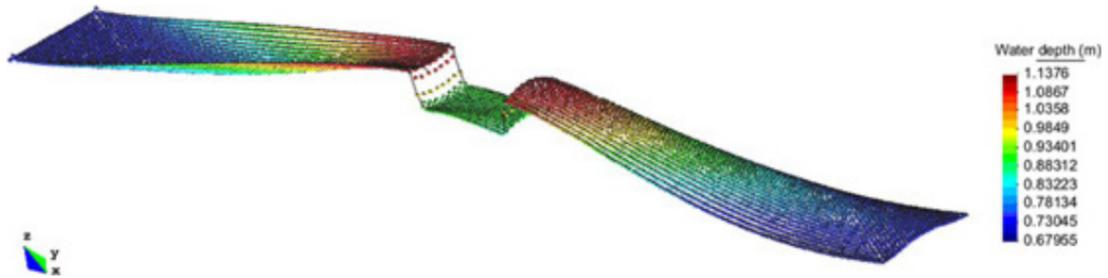


Figura 2.10: Caso do perfil de profundidade da água. BUACHART *et al.* [7].

de calor ou minimizar a queda de pressão.

ABADI e JAFARI [36] desenvolveram um estudo numérico detalhado da transferência de calor por convecção natural laminar em torno de dois cilindros elípticos confinados em uma cavidade fechada. Utilizando um solucionador CBS para as equações de Navier-Stokes, os autores investigaram o impacto da variação do espaçamento vertical entre os cilindros, da orientação dos mesmos e do número de Rayleigh nas características do escoamento e na transferência de calor. Os resultados obtidos contribuíram significativamente para a compreensão dos fenômenos de convecção natural em geometrias complexas, fornecendo informações valiosas para o projeto de sistemas térmicos.

MINH TUAN NGUYEN e LEE [8] apresentaram uma análise numérica detalhada da convecção natural de nano-fluidos de cobre e água em cavidades porosas apresentado na Figura 2.13, empregando o método dos elementos finitos com o esquema CBS, investigaram os efeitos combinados de diversos parâmetros, como o número de Rayleigh, o número de Darcy e a fração de volume de nanopartículas. Os resultados obtidos revelaram que a interação entre esses parâmetros influencia significativamente a transferência de calor e o escoamento do fluido. A análise dos regimes de escoamento permitiu identificar as condições ótimas para maximizar a transferência de calor considerando as propriedades do nano-fluido e as características do meio poroso, permitindo resultados satisfatórios pelo método CBS.

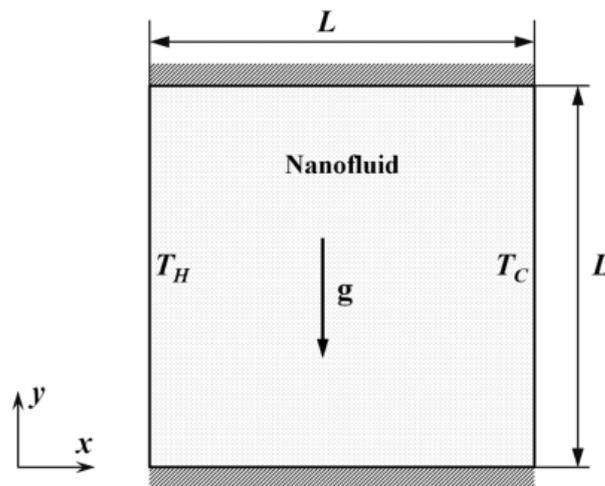


Figura 2.11: Esquema físico de nano-fluidos cobre e água de MINH TUAN NGUYEN e LEE [8].

Com o objetivo de otimizar o processo de encapsulamento de pastilhas flip-chip, KULKARNI *et al.* [9] utilizou um modelo numérico baseado no método CBS, permitindo assim, simular o escoamento do fluido de encapsulamento na cavidade da pastilha, considerando a interface livre entre o fluido e o ar, demonstrado na Figura 2.12. Os resultados obtidos fornecem informações valiosas para o projeto de processos de encapsulamento mais eficientes, permitindo reduzir defeitos e melhorar a qualidade dos produtos finais.

O método CBS mostrou-se uma ferramenta eficaz para a simulação numérica do processo de encapsulamento em embalagens eletrônicas, estudada por KULKARNI *et al.* [9]. Ao aplicar o método CBS em sua forma semi-discreta, foi possível obter resultados precisos para os perfis frontais do fluido, permitindo uma melhor compreensão dos fenômenos físicos envolvidos no processo, como analisado nos padrões de velocidade, apresentados na Figura 2.13. Essa ferramenta pode ser utilizada para otimizar o projeto de embalagens, reduzindo defeitos e aumentando a confiabilidade dos produtos eletrônicos.

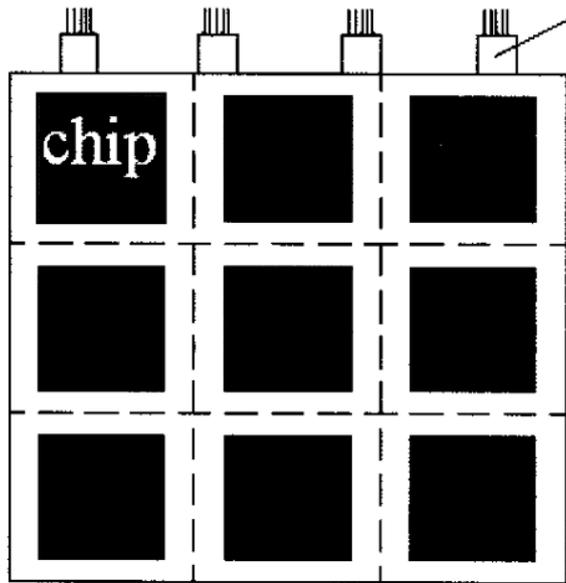


Figura 2.12: Limites da pastilha de chip, analisado por KULKARNI *et al.* [9].

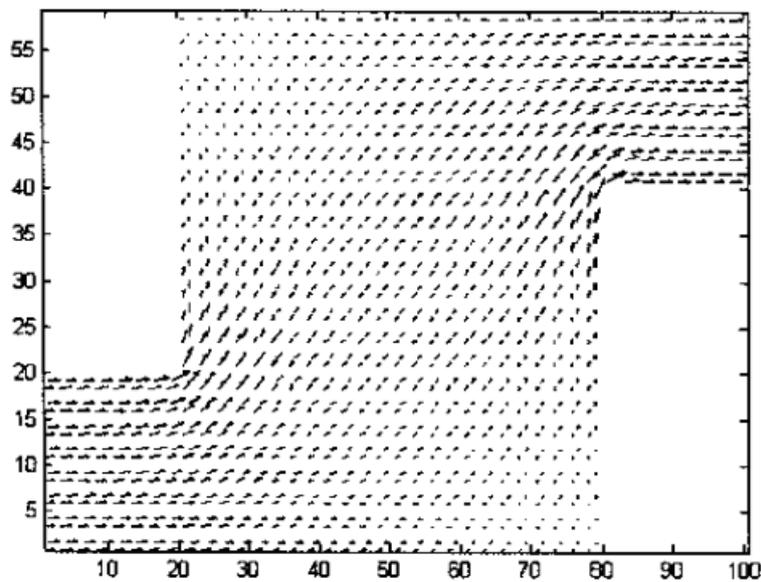


Figura 2.13: Padrão de velocidade KULKARNI *et al.* [9].

No estudo de LI e DUAN [10] foram propostos dois novos esquemas estabilizados para o método dos passos fracionários em malhas livres, baseados no método CBS e de Taylor-Galerkin. A introdução de um procedimento iterativo para o termo convectivo permitiu o uso de passos de tempo significativamente maiores, tornando os métodos mais eficientes. Com essa associação dos métodos, foi desenvolvida uma nova resolução para a estabilização da incompressibilidade, evitando o cálculo de derivadas espaciais de alta ordem. Dois casos foram explorados no estudo, o escoamento em cavidade com tampa, como demonstrado na Figura 2.14, e o escoamento de Poiseuille no plano, os resultados numéricos demonstraram a robustez e a pre-

cisão dos esquemas propostos, tornando-os ferramentas poderosas para a simulação como a analisada. Comprovando assim a capacidade e o desempenho do esquema CBS em casos de passos fracionados sem malha em problemas de escoamento incompressível.

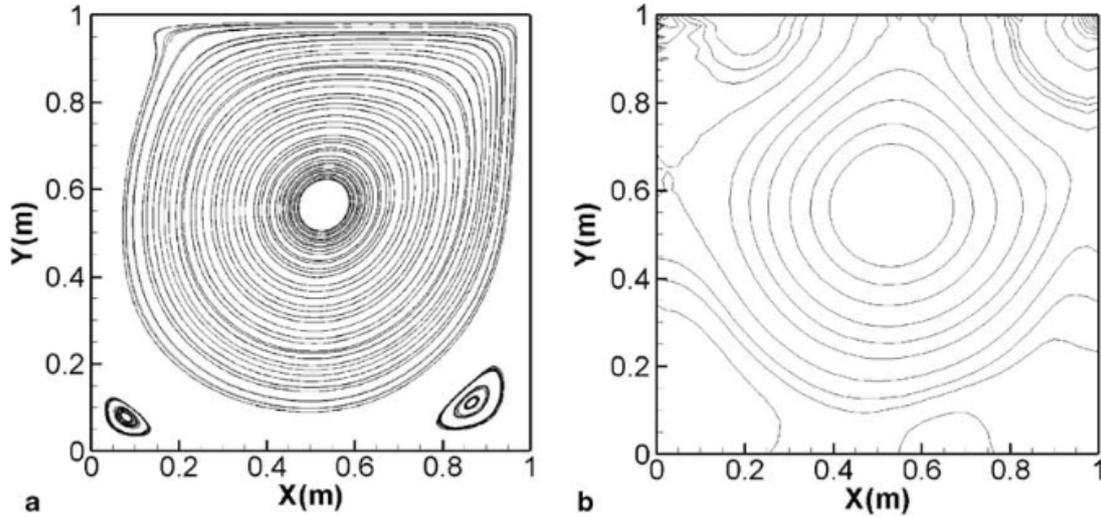


Figura 2.14: Soluções de LI e DUAN [10] para o problema do escoamento em cavidades com tampa com $Re = 1000$, utilizando CBS para estabilização de pressão. (a) Linhas de fluxo, CBS, $\Delta t = 0.1 \Delta t_c$. (b) Contornos de pressão, CBS, $\Delta t = 0.1 \Delta t_c$.

Por fim, alguns trabalhos que merecem alguma menção, embora sejam baseados em versões do tipo *meshfree*, são aqueles publicados por DUAN e LI [11] e ZHANG *et al.* [37]. Em ambos os casos apresentam-se abordagens alternativas de solucionar a formulação ALE através do método CBS sem, entretanto, fazer uso de malha (*meshfree*), para por exemplo, analisar as distribuições de velocidade como demonstrado na Figura 2.15 e assim evitando necessidade de suavização e outras dificuldades típicas do uso da malha.

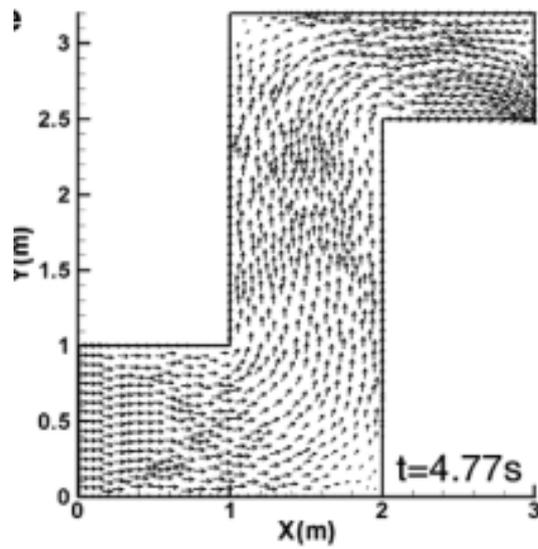


Figura 2.15: posições das superfícies livres em movimento e das malhas correspondentes ao nível de tempo $t = 3,2$ s no estudo de DUAN e LI [11] .

Capítulo 3

Formulação das equações

3.1 Modelagem do problema de convecção-difusão

O problema típico do processo de convecção-difusão de uma grandeza vetorial pode ser escrito na forma de leis de conservação a partir do balanço de uma quantidade vetorial ϕ_i com fluxos representados pelos tensores F_{ij} e G_{ij} adentrando determinado volume de controle e Q_i sendo um termo fonte (vetorial). Sendo assim, o modelo transiente pode ser escrito na forma

$$\begin{aligned}\frac{\partial \phi_i}{\partial t} + \frac{\partial F_{ij}}{\partial x_i} + \frac{\partial G_{ij}}{\partial x_i} + Q_i &= 0 \\ F_{ij} &= F_{ij}(\phi_i) \\ G_{ij} &= G_{ij}\left(\frac{\partial \phi_i}{\partial x_j}\right) \\ Q_i &= Q_i(x_i, \phi_i)\end{aligned}\tag{3.1}$$

Essa equação é conhecida como Equação de Transporte e o termo F_{ij} pode ser entendido como um fluxo de natureza convectiva ao passo que G_{ij} é o termo de natureza difusiva. Vale ressaltar que essa é uma modelagem geral sendo útil não somente para problemas em mecânica dos fluidos, mas também outras classes de problemas físicos [17].

Caso a grandeza de interesse seja um escalar $\phi_i = \phi$, então os fluxos passam de tensores de segunda para tensores de primeira ordem e o termo fonte passa a ser um escalar. Nesse caso, os termos de fluxo e fonte podem ser

$$F_{ij} = F_i = U_i \phi \tag{3.2}$$

$$G_{ij} = G_i = -k \frac{\partial \phi}{\partial x_i} \tag{3.3}$$

$$Q_i = Q \tag{3.4}$$

sendo U_i uma velocidade normalmente conhecida *a priori*, ϕ é a grandeza escalar transportada pelo fenômeno de convecção-difusão e k é um coeficiente relacionado à difusão de ϕ . O termo Q é uma fonte externa da grandeza ϕ que é adicionada ao ou removida do sistema. Substituindo a Equação 3.2, 3.3 e 3.4 na Equação 3.1 se obtém a forma

$$\frac{\partial \phi}{\partial t} + U_i \frac{\partial \phi}{\partial x_i} + \phi \frac{\partial U_i}{\partial x_i} - \frac{\partial}{\partial x_i} \left(k \frac{\partial \phi}{\partial x_i} \right) + Q = 0 \quad (3.5)$$

Equações na forma auto-adjunta podem ser aproximadas pelo método de Galerkin pois este fornece o menor erro possível na norma da energia e assim se obtém uma aproximação ótima [38]. A Equação 3.5 evidencia que, exceto em casos em que a velocidade é muito baixa e a convecção pode ser desprezada, essa equação deve ser tratada de forma especial já que ela perde a propriedade de ser auxiliar com a presença do termo convectivo.

3.2 Método das características

A abordagem do método das características permite contornar a presença do termo convectivo com uma transformação de variáveis e escrita da Equação de Transporte, no contexto relevante da Mecânica dos Fluidos, na forma de transporte unicamente de grandezas escalares. Para isso, inicialmente a Equação 3.5 pode ser reescrita tomando-se uma mudança de coordenadas na direção das características, tal que $\phi = \phi(x'_i, t)$ de modo que para $x'_i = \text{cte}$

$$\frac{\partial}{\partial t} \phi(x'_i, t) = \frac{\partial}{\partial x'_i} \phi(x'_i, t) \frac{\partial x'_i}{\partial t} + \frac{\partial}{\partial t} \phi(x'_i, t) \frac{\partial t}{\partial t} = \frac{\partial \phi}{\partial x'_i} U_i + \frac{\partial \phi}{\partial t} \quad (3.6)$$

que pode ser substituído na Equação 3.5 para se obter

$$\frac{\partial \phi}{\partial t} + \phi \frac{\partial U_i}{\partial x'_i} - \frac{\partial}{\partial x'_i} \left(k \frac{\partial \phi}{\partial x'_i} \right) + Q = 0 \quad (3.7)$$

A mudança de variáveis que origina a Equação 3.7 garante que haja somente operadores auto-adjuntos no espaço, de modo que pode-se aplicar uma aproximação espacial a partir do método de da característica de Galerkin (*characteristics-Galerkin procedure*). Embora a mudança de variáveis traga essa vantagem, ela também gera um problema que é justamente o movimento do sistema de coordenadas e isso pode ser contornado a partir de uma aproximação local de segunda ordem. Já a derivada no tempo pode ser discretizada com um esquema de diferenças finitas de primeira ordem, por exemplo.

$$\begin{aligned} \frac{1}{\Delta t}(\phi|_{\mathbf{x}}^{n+1} - \phi|_{\mathbf{x}-\delta}^n) + \theta \left[\phi \frac{\partial U_i}{\partial x'_i} - \frac{\partial}{\partial x'_i} \left(k \frac{\partial \phi}{\partial x'_i} \right) + Q \right]_{\mathbf{x}}^{n+1} \\ + (1 - \theta) \left[\phi \frac{\partial U_i}{\partial x'_i} - \frac{\partial}{\partial x'_i} \left(k \frac{\partial \phi}{\partial x'_i} \right) + Q \right]_{\mathbf{x}-\delta}^n = 0 \end{aligned} \quad (3.8)$$

em que $\theta \in [0, 1]$. É importante observar que se $\theta = 0$ a equação fica na forma explícita, ao passo que valores $0 < \theta < 1$ correspondem à forma semi-implícita e $\theta = 1$ à forma completamente implícita.

A Figura 3.1 apresenta a característica e demonstra, juntamente com a avaliação dos termos da Equação 3.8, a necessidade de se considerar os valores para um mesmo ponto do domínio. Há duas formas distintas de fazê-lo: a primeira forma é realizar aproximações para ter os termos no passo $n + 1/2$ (i.e. $\theta = 1/2$), tendo uma discretização centrada com esquema de Crank-Nicolson. Outra maneira é ter os termos no passo $n + 1$ o que faz surgir alguns termos que contribuem para estabilização numérica do método [18].

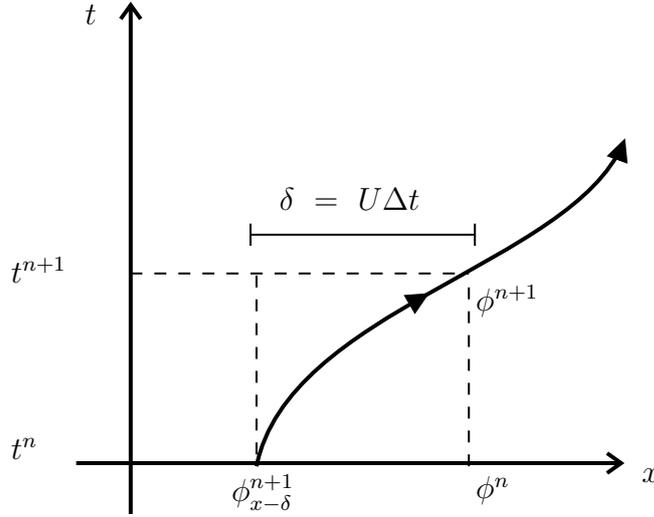


Figura 3.1: Representação gráfica da característica usada no método.

É preciso obter aproximações dos termos $(\cdot)|_{\mathbf{x}-\delta}^n$ da Equação 3.8 para tê-los no passo n e na posição \mathbf{x} . Para manter a simplicidade da dedução, será assumido por hipótese que o divergente seja nulo ($\partial U_i / \partial x_i = 0$) e que o problema é unidimensional. Sendo assim, pode-se lançar mão de aproximações por séries de Taylor para os termos dessa equação, ou seja

$$\phi|_{\mathbf{x}-\delta}^n \approx \phi^n - \delta \frac{\partial \phi^n}{\partial x} + \frac{\delta^2}{2} \frac{\partial}{\partial x} \left(\frac{\partial \phi^n}{\partial x} \right) + O(\delta^3) \quad (3.9)$$

$$\frac{\partial}{\partial x'} \left(k \frac{\partial \phi}{\partial x'} \right)_{\mathbf{x}-\delta}^n \approx \frac{\partial}{\partial x} \left(k \frac{\partial \phi}{\partial x} \right)^n - \delta \frac{\partial}{\partial x} \left[\frac{\partial}{\partial x} \left(k \frac{\partial \phi}{\partial x} \right)^n \right] + O(\delta^2) \quad (3.10)$$

$$Q|_{x-\delta}^n \approx Q^n - \delta \frac{\partial Q^n}{\partial x} + O(\delta^2) \quad (3.11)$$

sendo que o $\delta = \bar{U} \Delta t$ a distância percorrida pela partícula na direção x e \bar{U} uma velocidade média ao longo da característica. A velocidade média pode ser aproximada de algumas formas, a adotada aqui é média

$$\bar{U} = \frac{U^{n+1} + U|_{x-\delta}^n}{2} \quad (3.12)$$

e o valor de $U|_{x-\delta}^n$ pode ser aproximado por uma série de Taylor

$$U|_{x-\delta}^n \approx U^n - U^n \Delta t \frac{\partial U^n}{\partial x} + O(\Delta t^2) \quad (3.13)$$

Escolhendo um valor de $\theta = 1/2$ e substituindo a Equação 3.13 na Equação 3.12 é possível avaliar os termos δ nas equações das expansões das séries de Taylor da Equação 3.8 na sua forma unidimensional (dispensada a notação de índices). Após as devidas manipulações algébricas pode-se obter a forma aproximada da equação

$$\begin{aligned} \frac{\phi^{n+1} - \phi^n}{\Delta t} = & -U^{n+1/2} \frac{\partial \phi^n}{\partial x} + \frac{\Delta t}{2} U^n \frac{\partial U^n}{\partial x} \frac{\partial \phi^n}{\partial x} + \frac{\Delta t}{2} U^{n+1} U^{n+1} \frac{\partial^2 \phi}{\partial x^2} \\ & + \frac{\partial}{\partial x} \left(k \frac{\partial \phi}{\partial x} \right)^{n+1/2} - \frac{\Delta t}{2} U^{n+1} \frac{\partial}{\partial x} \left[\frac{\partial}{\partial x} \left(k \frac{\partial \phi}{\partial x} \right) \right]^n \\ & - Q + \frac{\Delta t}{2} U^{n+1} \frac{\partial Q}{\partial x} \end{aligned} \quad (3.14)$$

sendo os valores de $(\cdot)^{n+1/2}$ as médias das parcelas em $n+1$ e n , isso é

$$U^{n+1/2} = \frac{U^{n+1} + U^n}{2} \quad (3.15)$$

$$\frac{\partial}{\partial x} \left(k \frac{\partial \phi}{\partial x} \right)^{n+1/2} = \frac{1}{2} \frac{\partial}{\partial x} \left(k \frac{\partial \phi}{\partial x} \right)^{n+1/2} + \frac{1}{2} \frac{\partial}{\partial x} \left(k \frac{\partial \phi}{\partial x} \right)^n \quad (3.16)$$

$$Q^{n+1/2} = \frac{Q^{n+1} + Q^n}{2} \quad (3.17)$$

Se, por outro lado, for feita uma aproximação dos termos médios anteriores na forma de $(\cdot)^{n+1/2} = (\cdot)^n + O(\Delta t)$ então é possível se obter uma equação na forma explícita representada por

$$\begin{aligned} \phi^{n+1} = & \phi^n - \Delta t \left[U \frac{\partial \phi}{\partial x} - \frac{\partial}{\partial x} \left(k \frac{\partial \phi}{\partial x} \right) + Q \right]^n \\ & + \frac{\Delta t^2}{2} U^n \frac{\partial}{\partial x} \left[U \frac{\partial \phi}{\partial x} - \frac{\partial}{\partial x} \left(k \frac{\partial \phi}{\partial x} \right) + Q \right]^n \end{aligned} \quad (3.18)$$

A simples recolocação da notação indicial já fornece uma representação mul-

tidimensional da Equação 3.18. Além disso, a hipótese de divergente zero feita anteriormente pode ser relaxada, de modo que essa equação valha também para os casos em que $\partial U_i/\partial x_i \neq 0$. Sendo assim, a Equação 3.18 pode ser reescrita de forma generalizada como

$$\begin{aligned} \phi^{n+1} = \phi^n - \Delta t \left[\frac{\partial(U_j\phi)}{\partial x_j} - \frac{\partial}{\partial x_i} \left(k \frac{\partial\phi}{\partial x_i} \right) + Q \right]^n \\ + \frac{\Delta t^2}{2} U_k^n \frac{\partial}{\partial x_k} \left[\frac{\partial(U_j\phi)}{\partial x_j} - \frac{\partial}{\partial x_i} \left(k \frac{\partial\phi}{\partial x_i} \right) + Q \right]^n \end{aligned} \quad (3.19)$$

Em posse da Equação 3.19 é possível usar o Método dos Resíduos Ponderados de Galerkin para se obter a discretização a partir da aproximação $\phi = \mathbf{N}\tilde{\phi}$ para se obter a equação matricial na forma explícita sem os termos de mais alta ordem e termos fonte

$$\mathbf{M}(\tilde{\phi}^{n+1} - \tilde{\phi}^n) = -\Delta t[(\mathbf{C}\tilde{\phi}^{n+1} + \mathbf{K}\tilde{\phi}^n + \mathbf{f}^n) + \Delta t(\mathbf{K}_u\tilde{\phi}^n + \mathbf{f}_s^n)] \quad (3.20)$$

Vale notar que o método descrito provoca o surgimento dos termos \mathbf{K}_u e \mathbf{f}_s^n devido à discretização ao longo das características. Em especial a aparição do termo \mathbf{K}_u é responsável justamente pela estabilização das oscilações numéricas típicas da presença da convecção na Equação de Transporte. As matrizes da Equação 3.20 são obtidas a partir do cálculo de

$$\begin{aligned} \mathbf{M} &= \int_{\Omega} \mathbf{N}^T \mathbf{N} d\Omega \\ \mathbf{C} &= \int_{\Omega} \mathbf{N}^T \frac{\partial(U_i\mathbf{N})}{\partial x_i} d\Omega \\ \mathbf{K} &= \int_{\Omega} \frac{\partial\mathbf{N}^T}{\partial x_i} k \frac{\partial\mathbf{N}}{\partial x_i} d\Omega \\ \mathbf{f} &= \int_{\Omega} \mathbf{N}^T Q d\Omega + \text{b.t.} \\ \mathbf{K}_u &= \frac{1}{2} \int_{\Omega} \frac{\partial(U_i\mathbf{N}^T)}{\partial x_i} \frac{\partial(U_i\mathbf{N})}{\partial x_i} d\Omega \\ \mathbf{f}_s &= \frac{1}{2} \int_{\Omega} \frac{\partial(U_i\mathbf{N}^T)}{\partial x_i} Q d\Omega + \text{b.t.} \end{aligned}$$

em que b.t. representa o termo da integral ao longo do contorno.

Essas equações envolvem termos de aproximação N conhecidos como *funções de forma* consistindo de funções contínuas definidas em um domínio, no caso do Método do Elementos Finitos, usualmente no nível do domínio do elemento. Essas funções formam um espaço de funções, que inclui funções que têm certas proprie-

dades de suavidade e integrabilidade. Em problemas reais, esses espaços de funções são infinitas e complexas demais para se resolver diretamente, e por isso se trabalha em subespaços finitos, onde a solução aproximada é possível (elementos).

Para construir esses subespaços finitos cada elemento do domínio é associado a uma função de forma N que define a contribuição desse elemento para a solução geral. As duas principais classes dessas funções são as de Lagrange e Hermite:

- Funções de Lagrange: usadas para interpolar pontos específicos dentro de um elemento. Elas pertencem ao espaço de funções contínuas e são úteis em problemas onde a continuidade da função é exigida, mas não necessariamente da sua derivada. No MEF, essas funções criam uma aproximação usando valores nodais (apenas os valores das funções nos pontos de controle), formando um subespaço de funções contínuas.
- Funções de Hermite: ao contrário das funções de Lagrange, as funções de Hermite também consideram derivadas nas interpolações. Assim, elas criam uma aproximação em subespaços onde tanto a função quanto sua derivada primeira são contínuas. São úteis para problemas que requerem continuidade de derivadas.

Cada conjunto de funções de forma, seja de Lagrange, Hermite, ou outros tipos, gera um subespaço finito do espaço de funções contínuas no domínio. A solução aproximada que obtemos no MEF é um elemento desse subespaço finito. O uso de diferentes funções de forma determina qual subespaço será utilizado e, portanto, influencia a precisão e as propriedades de suavidade da solução aproximada. Essas funções são representadas na forma polinomial e são fundamentais para solução do problema.

Uma versão ainda mais detalhada desta dedução geral da Equação 3.19 pode ser encontrada em ZIENKIEWICZ e CODINA [18] e em ZIENKIEWICZ e TAYLOR [17] e a discretização pelo Método dos Elementos Finitos pode ser encontrada em ZIENKIEWICZ e TAYLOR [38] e em NITHIARASU *et al.* [19].

3.3 Separação das curvas características (CBS)

Até aqui se apresentou a forma geral da dedução das equações usando o método da separação das curvas características, válido para o transporte de um grandeza escalar genérica ϕ . A partir desta seção é desenvolvida e discutida a especialização dessas equações para o contexto do transporte de grandezas de interesse do problema de escoamento de fluidos solucionado pelo WaveFE e, sendo assim, a discussão é feita em termos do princípio de conservação de massa, *momentum* e energia.

Devido à natureza não auto-adjunta das equações, a aproximação de Galerkin ótima não pode ser diretamente utilizada. Apesar disso, mediante a uma separação específica das equações a parte não auto-adjunta, com estrutura semelhante à equação de difusão por convecção, torna-se suscetível ao método de Galerkin característico. Embora diversos métodos alternativos levem a termos de estabilização equivalentes, o método CBS tira vantagem de sua interpretação física mais clara. Além disso, ao contrário das equações não conservativas, que podem levar a soluções não físicas ou imprecisas, as equações de conservação garantem a consistência dos resultados e a preservação das quantidades físicas relevantes.

ZIENKIEWICZ e TAYLOR [17] afirmam que o algoritmo apresentado constitui a ferramenta mais geral atualmente disponível para a simulação numérica de escoamentos de fluidos. Em particular, para escoamentos viscosos em regimes de baixa velocidade, o método apresenta uma forte analogia com os métodos utilizados para sólidos de comportamento elásticos incompressíveis, permitindo a aplicação de técnicas de elementos finitos de alta ordem que satisfaçam a condição de Babuška-Brezzi.

As equações da dinâmica dos fluidos são ferramentas versáteis para modelar uma ampla variedade de fenômenos, desde o escoamento de líquidos até a propagação de ondas em gases e a forma geral das equações pode ser usada para escoamentos compressíveis e incompressíveis. Todas as propriedades essenciais dessas equações podem ser descritas pelas seguintes três equações: equação da conservação da massa, equação da conservação de *momentum* e equação de conservação de energia.

A equação para conservação de massa:

$$\frac{\partial \rho}{\partial t} = - \frac{\partial U_i}{\partial x_i} \quad (3.21)$$

Para escoamentos de baixa compressibilidade

$$\frac{\partial \rho}{\partial t} = \frac{1}{c^2} \frac{\partial p}{\partial t} = - \frac{\partial U_i}{\partial x_i} \quad (3.22)$$

Considerando c como velocidade do som que depende de E , p , e ρ ,

$$c^2 = \frac{\partial p}{\partial \rho} = \frac{\gamma p}{\rho} \quad (3.23)$$

Por sua vez γ é a razão dos calores específicos igual a c_p/c_v , no qual c_p é o calor específico a pressão constante e c_v é o calor específico a volume constante, e definimos o fluxo de massa como

$$U_i = \rho u_i \quad (3.24)$$

para um fluido com uma pequena compressibilidade, a velocidade do som pode ser

expressa como

$$c^2 = \frac{K}{\rho} \quad (3.25)$$

no qual K é o módulo de elasticidade aparente. De acordo com a utilização, se recorre a uma relação adequada para c^2 .

A equação para conservação de *momentum*:

$$\frac{\partial U_i}{\partial t} = -\frac{\partial}{\partial x_j} (u_j U_i) + \frac{\partial \tau_{ij}}{\partial x_j} - \frac{\partial p}{\partial x_i} + \rho g_i \quad (3.26)$$

A equação para conservação de energia:

$$\frac{\partial(\rho E)}{\partial t} = -\frac{\partial}{\partial x_i} (u_i \rho E) + \frac{\partial}{\partial x_i} \left(k \frac{\partial T}{\partial x_i} \right) - \frac{\partial}{\partial x_i} (u_i p) + \frac{\partial}{\partial x_i} (\tau_{ij} u_j) + \rho g_i u_i \quad (3.27)$$

Em todos os casos acima apresentados, u_i são as variáveis de velocidade, ρ é a densidade, E é a energia específica, p é a pressão, T é a temperatura absoluta, ρg_i representa as forças de corpo e outros termos fonte, k é a condutividade térmica e τ_{ij} são as variáveis de tensão desviatórias (*deviatoric stress*) dadas pela equação a seguir

$$\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} \delta_{ij} \frac{\partial u_k}{\partial x_k} \right) \quad (3.28)$$

sendo δ_{ij} o delta de Kronecker e igual a 1 se $i = j$ e 0 se $i \neq j$. Em geral, μ , a viscosidade dinâmica, na equação anterior é uma função da temperatura, $\mu(T)$, e serão utilizadas relações apropriadas, se forem necessárias. As equações são preenchidas por uma correlação que associa a pressão e a densidade. Uma das alternativas mais utilizadas para esta relação é a lei universal dos gases que se aplica quando o escoamento é compressível, ou seja

$$p = \rho R T \quad (3.29)$$

em que R é a constante universal dos gases.

3.4 Formas adimensionais das equações

As equações determinantes descritas até agora são frequentemente escritas na forma não-dimensional. As escalas usadas para adimensionalizar estas equações variam dependendo da natureza do escoamento. Pode-se usar a escala seguinte nos cálculos de escoamento, apesar de ser possível formas alternativas:

$$\bar{t} = \frac{tu_\infty}{L}, \quad \bar{x}_i = \frac{x_i}{L}, \quad \bar{\rho} = \frac{\rho}{\rho_\infty}, \quad \bar{p} = \frac{p}{\rho_\infty u_\infty^2}$$

$$\bar{u}_i = \frac{u_i}{u_\infty}, \quad \bar{E} = \frac{E}{u_\infty^2}, \quad \bar{T} = \frac{Tc_p}{u_\infty^2}, \quad \bar{c}^2 = \frac{c^2}{u_\infty^2}$$

onde uma barra na parte superior das variáveis indica uma quantidade não dimensional, o subscrito ∞ representa uma quantidade de escoamento livre e L é um comprimento de referência. Aplicando as escalas acima às equações determinantes e reorganizando-as, se alcança a seguinte forma:

Conservação da massa:

$$\frac{\partial \bar{\rho}}{\partial \bar{t}} = -\frac{\partial \bar{U}_i}{\partial \bar{x}_i}$$

Conservação de *momentum*:

$$\frac{\partial \bar{U}_i}{\partial \bar{t}} = -\frac{\partial}{\partial \bar{x}_j} (\bar{u}_j \bar{U}_i) + \frac{1}{\text{Re}} \frac{\partial (\bar{v} \bar{\tau}_{ij})}{\partial \bar{x}_j} - \frac{\partial \bar{p}}{\partial \bar{x}_i} + \bar{\rho} \bar{g}_i$$

em que

$$\text{Re} = \frac{u_\infty L}{\nu_\infty}, \quad \bar{g}_i = \frac{g_i L}{u_\infty^2}, \quad \bar{\nu} = \frac{\nu}{\nu_\infty}$$

são o número de Reynolds, as forças não dimensionais de corpo e o coeficiente de viscosidade, respetivamente. Na equação anterior, ν é a viscosidade cinemática igual a μ/ρ , sendo μ a viscosidade dinâmica.

Conservação de Energia:

$$\frac{\partial (\bar{\rho} \bar{E})}{\partial \bar{t}} = -\frac{\partial}{\partial \bar{x}_i} (\bar{u}_i \bar{\rho} \bar{E}) + \frac{1}{\text{RePr}} \frac{\partial}{\partial \bar{x}_i} \left(k^* \frac{\partial \bar{T}}{\partial \bar{x}_i} \right) - \frac{\partial}{\partial \bar{x}_i} (\bar{u}_j \bar{p}) + \frac{1}{\text{Re}} \frac{\partial}{\partial \bar{x}_i} (\bar{v} \bar{x}_{ij} \bar{u}_j)$$

em que Pr é o número de Prandtl e k^* é o coeficiente de condutividade dado pelas equações

$$\text{Pr} = \frac{\mu_\infty c_p}{k_\infty}, \quad k^* = \frac{k}{k_\infty}$$

onde k_∞ é a condutividade térmica de referência, sendo a equação de estado

$$\bar{p} = \frac{\bar{\rho} R \bar{T}}{c_p} = \bar{\rho} \bar{R} \bar{T} = \bar{\rho} \frac{(\gamma - 1)}{\gamma} \bar{T}$$

Na equação acima são utilizados $R = c_p - c_v$ e $\bar{R} = (c_p - c_v)/c_p$. As seguintes formulações de equações não dimensionais são úteis para relacionar a velocidade do

som, a temperatura, a pressão, a energia, etc. e com isso usa-se

$$\bar{E} = \frac{\bar{T}}{\gamma} + \frac{1}{2}\bar{u}_i\bar{u}_i\bar{c}^2 = (\gamma - 1)\bar{T}\bar{p} = (\gamma - 1)\left(\bar{\rho}\bar{E} - \frac{1}{2}\frac{\bar{U}_i\bar{U}_i}{\bar{\rho}}\right) \quad (3.30)$$

3.5 O método da separação das características generalizado

O separação das características segue o processo inicialmente introduzido por CHORIN [22] para problemas de escoamentos incompressíveis num contexto de diferenças finitas. Nesta seção estende-se o método para resolver as equações da dinâmica dos fluidos de ambas as formas, compressível e incompressível, usando o procedimento de Galerkin.

Embora a separação original de CHORIN [22] não pudesse ser usada num algoritmo totalmente explícito, a forma com CBS pode ser aplicada a escoamentos totalmente compressíveis, tanto na forma explícita como na semi-implícita. A separação fornece um algoritmo totalmente explícito mesmo no caso do escoamento incompressível para problemas em estado estacionário, utilizando agora uma *compressibilidade artificial* que não afeta a solução em estado estacionário; o estado não estacionário pode ser recuperado através do passo de tempo duplo. Quando existe compressibilidade real, como nos escoamentos de gás, as vantagens computacionais da forma explícita contrastam bem, quando comparadas com os de outros algoritmos atualmente utilizados [17].

Discretização temporal

Nesta etapa, apenas será considerada a solução de 3.21 e 3.26 com as variáveis u_i e p . A possibilidade de extensão para incluir energia e quaisquer outras variáveis será tratada depois de estas estarem calculadas. Se discretiza a equação 3.27 no tempo usando o processo de Galerkin neste estágio. Com exceção do termo de pressão, esta equação é semelhante à equação de convecção-difusão 3.7. Esse termo pode, no entanto, ser tratado como uma quantidade conhecida desde que se tenha uma forma independente de avaliar a pressão. Antes de avançar com o algoritmo, repete-se a equação 3.27 abaixo para utilização com o método de Galerkin:

$$\frac{\partial U_i}{\partial t} = -\frac{\partial}{\partial x_j}(u_j U_i) + \frac{\partial \tau_{ij}}{\partial x_j} - \frac{\partial p}{\partial x_i} + \rho g_i \quad (3.31)$$

Agora é necessário introduzir a separação para uma aproximação adequada. Em todos os procedimentos, os valores da solução (U_i^{n+1}, p^{n+1}) no instante t^{n+1} têm de ser determinados a partir dos valores conhecidos (U_i^n, p^n) no instante t^n . Duas

abordagens alternativas são aplicáveis e são descritas como *separação A* e *separação B*, respectivamente. Em cada uma delas, se assume que durante o acréscimo de tempo $\Delta t = t^{n+1} - t^n$

$$U_i^{n+1} = U_i^n + \Delta U_i^* + \Delta U_i^{**} \quad (3.32)$$

Para a discretização no tempo, usa-se a aproximação para o intervalo de tempo $t^n \leq t \leq t^{n+1}$

$$\frac{\partial U_i}{\partial t} = \frac{U_i^{n+1} - U_i^n}{\Delta t} = \frac{\Delta U_i^*}{\Delta t} + \frac{\Delta U_i^{**}}{\Delta t} \quad (3.33)$$

Utilizando a equação 3.19 e substituindo ϕ por U_i , se descreve

$$\begin{aligned} U_i^{n+1} - U_i^n = & \Delta t \left[-\frac{\partial}{\partial x_j} (u_j U_i)^n + \frac{\partial \tau_{ij}^n}{\partial x_j} + (\rho g_i)^n \right] - \Delta t \frac{\partial p^{n+\theta_2}}{\partial x_i} \\ & + \frac{\Delta t^2}{2} u_k \frac{\partial}{\partial x_k} \left[\frac{\partial}{\partial x_j} (u_j U_i) - \frac{\partial \tau_{ij}^n}{\partial x_j} - \rho g_i \right]^n \\ & + \frac{\Delta t^2}{2} u_k \frac{\partial}{\partial x_k} \left(\frac{\partial p^{n+\theta_2}}{\partial x_i} \right) \end{aligned} \quad (3.34)$$

em que

$$\frac{\partial p^{n+\theta_2}}{\partial x_i} = (1 - \theta_2) \frac{\partial p^n}{\partial x_i} + \theta_2 \frac{\partial p^{n+1}}{\partial x_i} \quad (3.35)$$

ou

$$\frac{\partial p^{n+\theta_2}}{\partial x_i} = \frac{\partial p^n}{\partial x_i} + \theta_2 \frac{\partial \Delta p}{\partial x_i} \quad (3.36)$$

na qual

$$\Delta p = p^{n+1} - p^n \quad (3.37)$$

é necessário ser considerado que os termos da velocidade e da tensão desviatória são determinados em t^n , enquanto a pressão é determinada em $t^{n+\theta_2}$. Isto é certamente permitido, uma vez que se demonstra que o método é totalmente consistente.

Utilizando as variáveis auxiliares ΔU_i^* e ΔU_i^{**} se divide a 3.34 em duas partes. Na primeira forma, todos os termos de gradiente de pressão da equação 3.34 s ao resolvidos; na segunda, se mantêm nessa equação o gradiente de pressão correspondente ao início do passo, ou seja, $\partial p^n / \partial x_i$. Embora pareça que a separação B possa ser mais precisa, há outras razões para o desempenho da separação A. O Split A é recomendada universalmente para problemas em estado estacionário, no entanto, para problemas transientes, o Split B, com estabilidade de pressão adicional, ou o Split A, com duplo passo de tempo, pode dar um resultado ligeiramente melhor [17].

Split A

Nesta forma, uma variável auxiliar ΔU_i^* é adotada e se desprezam os termos de terceira ordem, possibilitando

$$\Delta U_i^* = \Delta t \left[-\frac{\partial}{\partial x_j} (u_j U_i) + \frac{\partial \tau_{ij}}{\partial x_j} + \rho g_i + \frac{\Delta t}{2} u_k \frac{\partial}{\partial x_k} \left(\frac{\partial}{\partial x_j} (u_j U_i) - \rho g_i \right) \right]^n \quad (3.38)$$

Esta equação é resolvida por um passo de tempo explícito aplicado à forma discretizada, sendo agora possível uma solução completa. A correção está disponível quando o incremento de pressão é avaliado:

$$\Delta U_i^{**} = -\Delta t \frac{\partial p^{n+\theta_2}}{\partial x_i} + \frac{\Delta t^2}{2} u_k \frac{\partial^2 p^n}{\partial x_k \partial x_i} \quad (3.39)$$

Da equação 3.21 temos

$$\Delta \rho = -\Delta t \frac{\partial U_i^{n+\theta_1}}{\partial x_i} = -\Delta t \left[\frac{\partial U_i^n}{\partial x_i} + \theta_1 \frac{\partial \Delta U_i}{\partial x_i} \right]$$

Substituindo ΔU_i pela variável auxiliar intermédia conhecida ΔU_i^* , utilizando as equações 3.33 e 3.39 e reorganizando e negligenciando os termos de terceira ordem e de ordem superior, obtemos

$$\Delta \rho = -\Delta t \left[\frac{\partial U_i^n}{\partial x_i} + \theta_1 \frac{\partial \Delta U_i^*}{\partial x_i} - \Delta t \theta_1 \left(\frac{\partial^2 p^n}{\partial x_i \partial x_i} + \theta_2 \frac{\partial^2 \Delta p}{\partial x_i \partial x_i} \right) \right] \quad (3.40)$$

A equação 3.40 é auto-adjunta na variável $\Delta \rho$ que é a incógnita. Agora, portanto, um procedimento padrão do tipo Galerkin pode ser utilizado de forma ótima para a aproximação espacial da Equação 3.40. As equações podem ser resolvidas após a discretização espacial na seguinte ordem:

1. Equação 3.38 para obter ΔU_i^*
2. Equação 3.40 para obter $\Delta \rho$;
3. Equação 3.39 para obter ΔU_i^{**} estabelecendo assim os valores de U_i e p da lei da energia e dos gases a t^{n+1}

Depois de completar o cálculo para estabelecer ΔU_i e $\Delta \rho$, a equação da energia é tratada independentemente e o valor de $(\rho E)^{n+1}$ é obtido pelo processo de Galerkin.

Split B

Nesta fração, é introduzida também uma variável auxiliar ΔU_i^* , retendo agora os valores conhecidos de $\partial p^n / \partial x_i$, ou seja

$$\Delta U_i^* = \Delta t \left[-\frac{\partial}{\partial x_j} (u_j U_i) + \frac{\partial \tau_{ij}}{\partial x_j} - \frac{\partial p}{\partial x_i} + \rho g_i + \frac{\Delta t}{2} u_k \frac{\partial}{\partial x_k} \left(\frac{\partial}{\partial x_j} (u_j U_i) + \frac{\partial p}{\partial x_i} - \rho g_i \right) \right]^n \quad (3.41)$$

Nesse momento é possível escrever a correção como

$$\Delta U_i^{**} = -\theta_2 \Delta t \frac{\partial \Delta p}{\partial x_i} \quad (3.42)$$

ou a correção é menor do que se assume para o Split A, Equação 3.39. Além disso, se utilizar a forma totalmente explícita com $\theta_2 = 0$, não é necessária qualquer correção da velocidade da massa (ΔU_i^{**}). O cálculo das alterações de densidade como no Split A:

$$\Delta \rho = -\Delta t \left[\frac{\partial U_i^n}{\partial x_i} + \theta_1 \frac{\partial \Delta U_i^*}{\partial x_i} - \Delta t \theta_1 \theta_2 \frac{\partial^2 \Delta p}{\partial x_i^2} \right] \quad (3.43)$$

As etapas de solução seguem os mesmos passos que no Split A. Vale ressaltar que o Split B não possui as propriedades de estabilização da pressão da separação A onde se encontra incompressibilidade.

3.6 Discretização espacial das equações

Em todas as equações apresentadas a seguir, é utilizado o procedimento de Galerkin padrão para a discretização espacial, uma vez que tal foi plenamente justificado para o método de Galerkin. Nessa etapa se emprega a aproximação espacial utilizando funções de forma de elementos finitos padrão como

$$\begin{aligned} U_i &= \mathbf{N}_u \tilde{\mathbf{U}}_i & \Delta U_i &= \mathbf{N}_u \Delta \tilde{\mathbf{U}}_i & \Delta U_i^* &= \mathbf{N}_u \Delta \tilde{\mathbf{U}}_i^* & \Delta U_i^{**} &= \mathbf{N}_u \Delta \tilde{\mathbf{U}}_i^{**} \\ u_i &= \mathbf{N}_u \tilde{\mathbf{u}}_i & p &= \mathbf{N}_p \tilde{\mathbf{p}} & \rho &= \mathbf{N}_\rho \tilde{\boldsymbol{\rho}} \end{aligned} \quad (3.44)$$

na equação acima

$$\begin{aligned} \tilde{\mathbf{U}}_i &= \left[U_i^1 \quad U_i^2 \quad \dots \quad U_i^a \quad \dots \quad U_i^m \right]^T \\ \mathbf{N} &= \left[N^1 \quad N^2 \quad \dots \quad N^a \quad \dots \quad N^m \right] \end{aligned} \quad (3.45)$$

em que a é o número de identificação do nó, variando entre 1 e m .

A seguinte forma da Equação 3.38 é obtida para a aproximação Galerkin padrão (as funções de ponderação são as funções de forma):

$$\begin{aligned}
& \int_{\Omega} N_u^a \Delta U_i^* d\Omega \\
&= -\Delta t \left[\int_{\Omega} N_u^a \frac{\partial}{\partial x_j} (u_j U_i) d\Omega + \int_{\Omega} \frac{\partial N_u^a}{\partial x_j} \tau_{ij} d\Omega - \int_{\Omega} N_u^a (\rho g_i) d\Omega \right]^n \\
&+ \frac{\Delta t^2}{2} \left[\int_{\Omega} \frac{\partial}{\partial x_k} (u_k N_u^a) \left(-\frac{\partial}{\partial x_j} (u_j U_i) + \rho g_i \right) d\Omega \right]^n \\
&+ \Delta t \left[\int_{\Gamma} N_u^a \tau_{ij} n_j d\Gamma \right]^n
\end{aligned} \tag{3.46}$$

Nesse momento, os termos viscosos e estabilizadores são integrados e o último termo é a integral de fronteira resultante da integração da contribuição viscosa. Uma vez que o resíduo nas fronteiras pode ser negligenciado, outras contribuições de fronteira dos termos estabilizadores são negligenciáveis.

Adotando-se a notação matricial quando a formulação de elementos finitos é efetuada, por conveniência. As relações de tensão desviatórias e taxa de deformação abaixo:

$$\tau_{ij} = 2\mu \left(\dot{\epsilon}_{ij} - \frac{1}{3} \delta_{ij} \dot{\epsilon}_{kk} \right)$$

em que a quantidade entre parêntesis é a taxa de deformação desviatória. No modelo acima

$$\dot{\epsilon}_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right)$$

e

$$\dot{\epsilon}_{kk} = \frac{\partial u_k}{\partial x_k}$$

Defini-se agora a taxa de deformação em três dimensões através de um vetor de seis componentes, como indicado abaixo: ¹

$$\dot{\epsilon} = \left[\dot{\epsilon}_{11} \quad \dot{\epsilon}_{22} \quad \dot{\epsilon}_{33} \quad 2\dot{\epsilon}_{12} \quad 2\dot{\epsilon}_{23} \quad 2\dot{\epsilon}_{31} \right]^T$$

Com um vetor \mathbf{m} definido como

$$\mathbf{m} = \left[1 \quad 1 \quad 1 \quad 0 \quad 0 \quad 0 \right]^T$$

encontra-se uma taxa de deformação volumétrica como

$$\dot{\epsilon}_v = \dot{\epsilon}_{ii} = \dot{\epsilon}_{11} + \dot{\epsilon}_{22} + \dot{\epsilon}_{33} = \mathbf{m}^T \dot{\epsilon}$$

A taxa de deformação desviatória pode agora ser escrita simplesmente como

$$\dot{\boldsymbol{\varepsilon}}^d = \dot{\boldsymbol{\varepsilon}} - \frac{1}{3} \mathbf{m} \dot{\varepsilon}_v = \left(\mathbf{I} - \frac{1}{3} \mathbf{m} \mathbf{m}^T \right) \dot{\boldsymbol{\varepsilon}} = \mathbf{I}_d \dot{\boldsymbol{\varepsilon}}$$

em que

$$\mathbf{I}_d = \left(\mathbf{I} - \frac{1}{3} \mathbf{m} \mathbf{m}^T \right)$$

e, portanto

$$\mathbf{I}_d = \frac{1}{3} \begin{bmatrix} 2 & -1 & -1 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 \\ -1 & -1 & 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3 \end{bmatrix}$$

As tensões são escritas de forma semelhante na forma vetorial como

$$\boldsymbol{\sigma} = \left[\sigma_{11} \quad \sigma_{22} \quad \sigma_{33} \quad \sigma_{12} \quad \sigma_{23} \quad \sigma_{31} \right]^T$$

em que σ_{11} é igual a $\tau_{11} - p$ ao passo que σ_{12} é igual a τ_{12} , e assim por diante. Pode-se assumir, portanto, que as tensões desviatórias são proporcionais às taxas de deformação desviatórias e pode se escrever

$$\boldsymbol{\sigma}^d = \mathbf{I}_d \boldsymbol{\sigma} = \mu \mathbf{I}_0 \dot{\boldsymbol{\varepsilon}}^d = \mu \left(\mathbf{I}_0 - \frac{2}{3} \mathbf{m} \mathbf{m}^T \right) \dot{\boldsymbol{\varepsilon}}$$

em que a matriz diagonal \mathbf{I}_0 é a matriz diagonal.

Para completar a derivação matricial, as velocidades e as taxas de deformação têm de ser adequadamente relacionadas e pode-se verificar que, utilizando as definições tensoriais de deformação, possibilitando a escrita de

$$\dot{\boldsymbol{\varepsilon}} = \mathcal{S} \mathbf{u}$$

onde

$$\mathbf{u} = \left[u_1 \quad u_2 \quad u_3 \right]^T$$

e \mathcal{S} é uma matriz de taxa de deformação (operador) definida abaixo:

$$\mathcal{S} = \begin{bmatrix} \frac{\partial}{\partial x_1} & 0 & 0 \\ 0 & \frac{\partial}{\partial x_2} & 0 \\ 0 & 0 & \frac{\partial}{\partial x_3} \\ \frac{\partial}{\partial x_2} & \frac{\partial}{\partial x_1} & 0 \\ 0 & \frac{\partial}{\partial x_3} & \frac{\partial}{\partial x_2} \\ \frac{\partial}{\partial x_3} & 0 & \frac{\partial}{\partial x_1} \end{bmatrix}$$

Finalmente, se pode notar que a ligação direta entre as taxas de deformação e as velocidades envolverá uma matriz \mathbf{B} definida simplesmente por

$$\mathbf{B} = \mathcal{S}\mathbf{N}_u$$

e sendo assim a solução para ΔU_i^* na forma matricial é a

Etapa 1:

$$\Delta \tilde{\mathbf{U}}^* = -\mathbf{M}_u^{-1} \Delta t \left[\left(\mathbf{C}_u \tilde{\mathbf{U}} + \mathbf{K}_\tau \tilde{\mathbf{u}} - \mathbf{f} \right) - \Delta t \left(\mathbf{K}_u \tilde{\mathbf{U}} + \mathbf{f}_s \right) \right]^n \quad (3.47)$$

em que as quantidades com e as matrizes de discretização são dadas por

$$\begin{aligned} \mathbf{M}_u &= \int_{\Omega} \mathbf{N}_u^T \mathbf{N}_u \, d\Omega & \mathbf{C}_u &= \int_{\Omega} \mathbf{N}_u^T (\nabla (\mathbf{u} \mathbf{N}_u)) \, d\Omega \\ \mathbf{K}_\tau &= \int_{\Omega} \mathbf{B}^T \mu \left(\mathbf{I}_0 - \frac{2}{3} \mathbf{m} \mathbf{m}^T \right) \mathbf{B} \, d\Omega & \mathbf{f} &= \int_{\Omega} \mathbf{N}_u^T \rho \mathbf{g} \, d\Omega + \int_{\Gamma} \mathbf{N}_u^T \mathbf{t}^d \, d\Gamma \end{aligned}$$

onde \mathbf{g} é $\begin{bmatrix} g_1 & g_2 & g_3 \end{bmatrix}^T$ e \mathbf{t}^d é a tração correspondente às componentes de tensão desviatórias.

Na Equação (3.51) \mathbf{K}_u e \mathbf{f}_s provêm dos termos introduzidos pela discretização ao longo das caraterísticas. Após integração por partes [i.e., a partir de (3.34)], as expressões para \mathbf{K}_u e \mathbf{f}_s são

$$\mathbf{K}_u = -\frac{1}{2} \int_{\Omega} [\nabla^T (\mathbf{u} \mathbf{N}_u)]^T [\nabla^T (\mathbf{u} \mathbf{N}_u)] \, d\Omega$$

E

$$\mathbf{f}_s = -\frac{1}{2} \int_{\Omega} [\nabla^T (\mathbf{u} \mathbf{N}_u)]^T \rho \mathbf{g} \, d\Omega$$

e $\nabla^T = [\partial/\partial x_1, \partial/\partial x_2, \partial/\partial x_3]$.

A forma fraca da equação densidade-pressão é

$$\begin{aligned}
\int_{\Omega} N_p^a \Delta \rho \, d\Omega &= -\Delta t \int_{\Omega} N_p^a \frac{\partial}{\partial x_i} \left(U_i^n + \theta_1 \Delta U_i^* - \theta_1 \Delta t \frac{\partial p^{n+\theta_2}}{\partial x_i} \right) d\Omega \\
&= \Delta t \int_{\Omega} \frac{\partial N_p^a}{\partial x_i} \left[U_i^n + \theta_1 \left(\Delta U_i^* - \Delta t \frac{\partial p^{n+\theta_2}}{\partial x_i} \right) \right] d\Omega \\
&\quad - \Delta t \int_{\Gamma} N_p^a \left[U_i^n + \theta_1 \left(\Delta U_i^* - \Delta t \frac{\partial p^{n+\theta_2}}{\partial x_i} \right) \right] n_i \, d\Gamma
\end{aligned}$$

Além disso, só é discretizado ρ diretamente em problemas de escoamentos de gases compressíveis e, por conseguinte, a seguir p se mantém como variável principal.

A discretização espacial da equação acima resulta em

Etapa 2:

$$(\mathbf{M}_p + \Delta t^2 \theta_1 \theta_2 \mathbf{H}) \Delta \tilde{\mathbf{p}} = \Delta t \left[\mathbf{G} \tilde{\mathbf{U}}^n + \theta_1 \mathbf{G} \Delta \tilde{\mathbf{U}}^* - \Delta t \theta_1 \mathbf{H} \tilde{\mathbf{p}}^n - \mathbf{f}_p \right] \quad (3.48)$$

que pode ser resolvido para $\Delta \tilde{\mathbf{p}}$. As novas matrizes que surgem aqui são

$$\mathbf{H} = \int_{\Omega} (\nabla \mathbf{N}_p)^T \nabla \mathbf{N}_p \, d\Omega \quad (3.49)$$

$$\mathbf{M}_p = \int_{\Omega} \mathbf{N}_p^T \left(\frac{1}{c^2} \right)^n \mathbf{N}_p \, d\Omega \quad (3.50)$$

$$\mathbf{G} = \int_{\Omega} (\nabla \mathbf{N}_p)^T \mathbf{N}_u \, d\Omega \quad (3.51)$$

$$\mathbf{f}_p = \Delta t \int_{\Gamma} \mathbf{N}_p^T \mathbf{n}^T \left[\tilde{\mathbf{U}}^n + \theta_1 \left(\Delta \tilde{\mathbf{U}}^* - \Delta t \nabla p^{n+\theta_2} \right) \right] d\Gamma \quad (3.52)$$

Tendo \mathbf{f}_p com a informação das condições de fronteira. A forma fraca da etapa de correção da Equação 3.39 é

$$\begin{aligned}
\int_{\Omega} N_u^a \Delta U_i^{**} \, d\Omega &= \int_{\Omega} N_u^a \Delta U_i \, d\Omega - \int_{\Omega} N_u^a \Delta U_i^* \, d\Omega \\
&= -\Delta t \int_{\Omega} N_u^a \left(\frac{\partial p^n}{\partial x_i} + \theta_2 \frac{\partial \Delta p}{\partial x_i} \right) d\Omega \\
&\quad - \frac{\Delta t^2}{2} \int_{\Omega} \frac{\partial}{\partial x_j} (u_j N_u^a) \frac{\partial p^n}{\partial x_i} \, d\Omega
\end{aligned}$$

A etapa final para o cálculo da velocidade U_i^{n+1} se dá na seguinte forma:

Etapa 3:

$$\Delta \mathbf{U}^{**} = \Delta \tilde{\mathbf{U}} - \Delta \tilde{\mathbf{U}}^* = -\mathbf{M}_u^{-1} \Delta t \left[\mathbf{G}^T (\tilde{\mathbf{p}}^n + \theta_2 \Delta \tilde{\mathbf{p}}) + \frac{\Delta t}{2} \mathbf{P} \tilde{\mathbf{p}}^n \right]$$

em que

$$\mathbf{P} = \int_{\Omega} (\nabla (\mathbf{u}\mathbf{N}_u))^T \nabla \mathbf{N}_p \, d\Omega$$

No final desta fase, os valores de $\tilde{\mathbf{U}}^{n+1}$ e $\tilde{\mathbf{p}}^{n+1}$ estão completamente determinados, mas o cálculo da energia $(\rho E)^{n+1}$ é necessário para os problemas de escoamento de gás, de modo a que a pressão e a densidade possam ser relacionadas através da temperatura. A equação da energia é idêntica na sua forma à do problema escalar de convecção-difusão observando-se que p, U_i são agora conhecidos. A forma fraca da equação da energia é escrita usando a aproximação característica de Galerkin como

$$\begin{aligned} & \int_{\Omega} N_E^k \Delta (\rho E)^{n+1} \, d\Omega \\ &= \Delta t \left[- \int_{\Omega} N_E^k \frac{\partial}{\partial x_i} (u_i (\rho E + p)) \, d\Omega - \int_{\Omega} \frac{\partial N_E^k}{\partial x_i} \left(\tau_{ij} u_j + k \frac{\partial T}{\partial x_i} \right) \, d\Omega \right]^n \\ &+ \frac{\Delta t^2}{2} \left[\int_{\Omega} \frac{\partial}{\partial x_j} (u_j N_E^k) \left[\frac{\partial}{\partial x_i} (-u_i (\rho E + p)) \right] \, d\Omega \right]^n \\ &+ \Delta t \left[\int_{\Gamma} N_E^k \left(\tau_{ij} u_j + k \frac{\partial T}{\partial x_i} \right) n_i \, d\Gamma \right]^n \end{aligned}$$

Etapa 4:

Com as aproximações adicionais

$$\rho E = \mathbf{N}_E \tilde{\mathbf{E}} \quad \text{and} \quad T = \mathbf{N}_T \tilde{\mathbf{T}}$$

Tem-se

$$\begin{aligned} \Delta \tilde{\mathbf{E}} = -\mathbf{M}_E^{-1} \Delta t \left[\mathbf{C}_E \tilde{\mathbf{E}} + \mathbf{C}_p \tilde{\mathbf{p}} + \mathbf{K}_T \tilde{\mathbf{T}} + \mathbf{K}_{\tau E} \tilde{\mathbf{u}} + \mathbf{f}_e \right. \\ \left. - \Delta t \left(\mathbf{K}_{uE} \tilde{\mathbf{E}} + \mathbf{K}_{up} \tilde{\mathbf{p}} + \mathbf{f}_{cs} \right) \right]^n \end{aligned} \quad (3.53)$$

onde $\tilde{\mathbf{E}}$ contém os valores nodais de ρE e, mais uma vez, as matrizes e os vetores de forçamento são semelhantes aos obtidos anteriormente e dados como

$$\begin{aligned} \mathbf{M}_E &= \int_{\Omega} \mathbf{N}_E^T \mathbf{N}_E \, d\Omega \\ \mathbf{C}_E &= \int_{\Omega} \mathbf{N}_E^T \nabla^T (\mathbf{u} \mathbf{N}_E) \, d\Omega \\ \mathbf{C}_p &= \int_{\Omega} \mathbf{N}_E^T \nabla^T (\mathbf{u} \mathbf{N}_p) \, d\Omega \\ \mathbf{K}_T &= \int_{\Omega} (\nabla \mathbf{N}_E)^T k \nabla \mathbf{N}_T \, d\Omega \end{aligned}$$

$$\mathbf{K}_{\tau E} = \int_{\Omega} \mathbf{B}^T \mu \mathbf{u}_{av} \left(\mathbf{I}_0 - \frac{2}{3} \mathbf{m} \mathbf{m}^T \right) \mathbf{B} d\Omega$$

$$\mathbf{K}_{uE} = -\frac{1}{2} \int_{\Omega} (\nabla^T (\mathbf{u} \mathbf{N}_E))^T (\nabla \mathbf{N}_E) d\Omega$$

$$\mathbf{f}_e = \int_{\Gamma} \mathbf{N}_E^T \mathbf{n}^T (\mathbf{t}^d \mathbf{u} + k \nabla T) d\Gamma$$

$$\mathbf{K}_{up} = -\frac{1}{2} \int_{\Omega} (\nabla^T (\mathbf{u} \mathbf{N}_E))^T (\nabla \mathbf{N}_p) d\Omega$$

O termo \mathbf{f}_{es} contém os termos fonte. Para o problema em tridimensional

$$\boldsymbol{\tau} = \begin{bmatrix} \tau_{11} & \tau_{12} & \tau_{13} \\ \tau_{21} & \tau_{22} & \tau_{23} \\ \tau_{31} & \tau_{32} & \tau_{33} \end{bmatrix}$$

3.7 Equações discretizadas para a separação B

Seguindo o mesmo procedimento pode-se obter a separação B cuja discretização conduz a:

Etapa 1:

$$\Delta \tilde{\mathbf{U}}_i^* = -\mathbf{M}_u^{-1} \Delta t \left[\left(\mathbf{C}_u \tilde{\mathbf{U}} + \mathbf{K}_{\tau} \tilde{\mathbf{u}} + \mathbf{G}^T \tilde{\mathbf{p}} - \mathbf{f} \right) - \Delta t \left(\mathbf{K}_u \tilde{\mathbf{U}} + \mathbf{f}_s + \frac{\Delta t}{2} \mathbf{P} \tilde{\mathbf{p}} \right) \right]_{3.6}^n \quad (3.65)$$

com as mesmas matrizes do separação A exceto o termo de forçamento \mathbf{f} que é

$$\mathbf{f} = \int_{\Omega} \mathbf{N}_u^T \rho \mathbf{g} d\Omega + \int_{\Gamma} \mathbf{N}_u^T \bar{\mathbf{t}} d\Gamma \quad (3.66)$$

Etapa 2:

$$(\mathbf{M}_p + \Delta t^2 \theta_1 \theta_2 \mathbf{H}) \Delta \tilde{\mathbf{p}} = \Delta t \left[\mathbf{G} \tilde{\mathbf{U}}^n + \theta_1 \mathbf{G} \Delta \tilde{\mathbf{U}}^* - \mathbf{f}_p \right]^n \quad (3.67)$$

Etapa 3:

$$\Delta \tilde{\mathbf{U}}^{**} = -\mathbf{M}_u^{-1} \Delta t \left[\theta_2 \mathbf{G}^T \Delta \tilde{\mathbf{p}} \right] \quad (3.68)$$

Step 4: O mesmo do separação A.

3.8 Diagonalização da matriz de massa

Nas etapas 1, 2 e 3 do CBS, a solução geralmente requer inversão das matrizes de massa \mathbf{M}_u e \mathbf{M}_p . Se feita a diagonalização dessas matrizes, essas etapas podem ser solucionadas de forma explícita a partir da estratégia de aproximação da matriz por uma forma diagonal (*lumped*). A forma comumente usada envolve somar os

valores dos elementos das linhas e atribuir esta soma ao elemento correspondente da diagonal principal daquela linha da matriz.

Essa estratégia pode ser usada em problemas com solução para o estado estacionário e faz com que as Etapas 1 e 3 sejam fáceis de resolver, além disso os erros envolvidos são desprezíveis já que os termos que envolvem variação no tempo desaparecem em uma solução no estado estacionário. Apesar disso, para problemas transientes os erros são altos e esse método não deve ser utilizado.

3.9 Forma explícita e semi-implícita

O algoritmo de separação A ou B contém uma parte explícita na primeira etapa e na segunda etapa, quando da determinação do incremento de pressão, pode ser tornado explícito ou implícito, a depender da escolha de θ_2 e do valor da velocidade do som. Os valores de θ_1 sempre será considerado de modo que atenda aos critérios de estabilidade $1/2 \leq \theta_1 \leq 1$. Os esquemas CBS vão ser totalmente explícitos ou semi-implícitos dependendo da escolha do parâmetro θ_2 como zero ou diferente de zero, respectivamente. É fundamental mencionar que a forma totalmente explícita só é viável como esquema de solução para fluxos compressíveis para os quais $c \neq \infty$, muito embora essa restrição pode ser relaxada caso se use uma abordagem com aplicação de uma compressibilidade artificial.

Na forma totalmente explícita assume-se

$$\frac{1}{2} \leq \theta_1 \leq 1 \text{ e } \theta_2 = 0$$

Em geral, as limitações do intervalo de tempo para as equações de convecção-difusão são aplicáveis, o que equivale dizer que se os efeitos da viscosidade forem desprezíveis o passo de tempo deve respeitar

$$\Delta t \leq \frac{h}{c + |\mathbf{u}|} \quad (3.69)$$

em que h é o tamanho do elemento, se assume apenas elementos lineares unidimensionais. Para problemas de fluxo bidimensionais e tridimensionais, a determinação do tamanho do elemento é difícil e envolve a determinação do mínimo tamanho do elemento por exemplo a partir da definição da menor das distâncias dos nós com relação às arestas/faces opostas. Uma forma simples é usar o tamanho de arestas vizinhas a cada ponto da malha para o valor de h .

Já para a forma semi-implícita o algoritmo continua a ser condicionalmente estável e assume-se os valores para os parâmetros

$$\frac{1}{2} \leq \theta_1 \leq 1 \quad \text{e} \quad \frac{1}{2} \leq \theta_2 \leq 1 \quad (3.70)$$

O passo de tempo permitido é governado pelo passo crítico da relação característica-Galerkin explícita resolvida na etapa 1. Este é o problema padrão de difusão por convecção discutido e os mesmos limites de estabilidade se aplicam

$$\Delta t \leq \Delta t_u = \frac{h}{|\mathbf{u}|}$$

$$\Delta t \leq \Delta t_v = \frac{h^2}{2\nu}$$

em que ν é a viscosidade cinemática. Para este caso pode-se usar o critério do menor valor de intervalo de tempo de modo a garantir a estabilidade independentemente do domínio da influência da viscosidade no escoamento, ou seja,

$$\Delta t = \min(\Delta t_u, \Delta t_v)$$

Para problemas ligeiramente compressíveis ou incompressíveis em que \mathbf{M}_p é pequeno ou zero, a forma semi-implícita é eficiente e deve-se notar que a matriz \mathbf{H} é constante durante todo processo de cálculo. Portanto \mathbf{H} pode ser fatorado em suas partes triangulares, uma vez que leva a um procedimento direto econômico. A equação implícita pode então ser solucionada via métodos iterativos como gradiente conjugado, embora possa também ser resolvida por métodos diretos.

3.10 Condições de contorno

Há basicamente três tipos de condições de contorno para domínios com contorno fisicamente definidos, são elas:

- Limites sólidos com condições de não-deslizamento: se assume que há adesão do fluido nas paredes do contorno sólido de modo que todos os componentes da velocidade sejam nulos;
- Limites sólidos em fluxo invíscido (condições de deslizamento): se assume esta condição quando se despreza a viscosidade do fluxo e o componente perpendicular à parede do sólido é especificada, normalmente com valor zero quando a parede é estática;
- Condições de limite de tração prescritas: esse tipo inclui a tração zero no caso de superfícies livres de fluidos ou quaisquer trações prescritas, como aquelas causadas pelo vento imposta na superfície de um aerofólio, por exemplo.

As principais condições de contorno usadas em problemas que envolvem convecção de calor são:

- condições de Dirichlet: quando são prescritas a temperatura, pressão e/ou velocidade;
- condições de Neumann: quando são prescritas como condições de contorno o valor do fluxo.

Se um valor das componentes de velocidade, o valor da temperatura ou da pressão for fornecido em um nó de fronteira, o valor será imposto nesses nós através da definição direta do valor nodal no momento da montagem das matrizes ou antes da solução do sistema linear resultante. Alternativamente, as equações discretas correspondentes àqueles nós de valor prescrito podem ser removidas antes da solução do sistema linear, uma vez que já são previamente conhecidas. Analogamente se prescrevem os valores dos fluxos, quando necessário, no momento da montagem das matrizes de modo a satisfazer a imposição da condição de contorno.

Capítulo 4

Arquitetura do simulador

4.1 Ferramentas utilizadas para o projeto

O projeto do programa incluiu a utilização de diversas ferramentas para criação do ambiente de desenvolvimento, testes, geração de malha, visualização de resultados, etc. As ferramentas utilizadas para o simulador estão descritas nesta seção.

O programa foi desenvolvido utilizando a linguagem Julia ¹ que se propõe ser uma linguagem moderna e de alta performance para aplicações numéricas. A linguagem possui um compilador baseado em LLVM e suporta compilação *Just In Time* (JIT) para múltiplas plataformas; sua sintaxe é dinâmica tendo definição opcional de tipos, muito embora muitas vezes recomendada; possui um vasto repositório de pacotes de uso geral e permite a criação e o gerenciamento de um ambiente reproduzível para o desenvolvimento no qual se pode instalar pacotes de forma isolada de outros projetos eventualmente desenvolvidos numa mesma máquina ou ser reproduzido em outra máquina de forma idêntica.

Os pacotes Julia utilizados para o desenvolvimento do projeto foram:

1. ArgParse: para leitura dos parâmetros de entrada do programa
2. BSON: geração de arquivos de regressão para os teste unitários
3. Debugger: depurador Julia
4. Gmsh: pacote para leitura e criação de malhas diretamente em Julia, de modo independente da interface gráfica
5. HDF5: para geração dos arquivos de resultados no formato homônimo ao pacote
6. Interpolations: para interpolar resultados gerados com relação aos resultados de referência

¹<https://julialang.org/>

7. `IterativeSolvers`: para solução das equações matriciais do modelo
8. `LinearAlgebra`: para criação e operações básicas com matrizes
9. `LoggingExtras`: registro dos eventos do simulador em arquivos de texto
10. `Preconditioners`: contém pacote de condicionadores para as matrizes
11. `ProgressMeter`: exibe barra de progresso para indicação visual do estado da simulação
12. `ReTest`: ferramenta de testes unitários de Julia
13. `Test`: idem
14. `ReferenceTests`: gerador de arquivos de para os teste de regressão
15. `SparseArrays`: para criação e operações básicas com matrizes esparsas
16. `Statistics`: para cálculos básicos, como média e outros
17. `TOML`: para leitura dos arquivos de entrada no formato de arquivo homônimo ao pacote

Como usual em projetos usando Julia, toda a descrição dos pacotes com seus respectivos *hash* de identificação encontra-se no arquivo `simulator/src/Project.toml`, juntamente com metadados que descrevem o projeto.

Para geração de malhas foi selecionado o `Gmsh` ² descrito em GEUZAINÉ e REMACLE [39] pela sua simplicidade de uso, interface gráfica para criação e visualização das malhas geradas e pela possibilidade de gerá-las diretamente via *script* utilizando sua API de código Julia. A aplicação também permite a criação de grupos de nós, característica fundamental para definição das condições de contorno dos casos de teste simulados.

O projeto foi desenvolvido utilizando o gerenciador de ambientes `Pixi` ³ capaz de criar e gerenciar ambientes de desenvolvimento baseado em pacotes `Conda` ⁴ e funcionando de forma muito semelhante a este. O `Pixi` utiliza arquivos contendo as dependências, comandos, metadados, etc. necessários para construção e execução do projeto. No caso deste projeto, o `Pixi` foi usado para instalar no ambiente contendo o `Gmsh` e a linguagem Julia de forma a se ter fixadas as suas versões pra evitar possíveis incompatibilidades com versões previamente instaladas na máquina do usuário ou desenvolvedor do projeto.

²<https://gmsh.info/>

³<https://prefix.dev/>

⁴<https://conda.io/projects/conda/en/latest/index.html>

Para visualização foi utilizado o pacote matplotlib para geração de figuras aqui presentes, no entanto este pacote não faz parte da base de código do simulador e foi somente configurado para ser usado com proposito de geração das figuras de forma manual para este trabalho.

4.2 Fluxo de execução no simulador

A sequência de execução do simulador está esquematizada na Figura 4.1. Inicialmente o programa carrega os dados de entrada do caso: arquivo de malha, arquivo de condições iniciais/contorno e arquivo de dados da simulação; com os arquivos importados o programa é capaz de checar a validade dos dados configurados do programa e decidir, caso haja algum problema dado na configuração do arquivo o programa encerra antes de iniciar qualquer simulação.

Em seguida o programa carrega o modelo selecionado, ou seja, carrega a função `run_iteration` do modelo que é responsável por executar cálculos particulares daquele modelo, para um passo de tempo. O carregamento do modelo também implica o carregamento das funções de montagem no nível do elemento que é justamente o que define o modelo, juntamente com as equações que definem quais funções de montagem devem ser chamadas para formar as equações e em que sequência.

Com o modelo carregado e configurado o programa é capaz de executar o *loop* principal. Nesse estágio o simulador executa a atualização de dados dos elementos tais como cálculo de propriedades geométricas e passo de tempo e também atualiza o valor das variáveis do modelo. Em seguida as funções de montagem configuradas para o modelo são chamadas para cada elemento da malha, independente do tipo do elemento, com objetivo de montar as matrizes globais. Esse processo é feito para cada uma das equações do modelo, na ordem configurada dentro da função `run_iteration`. Montadas as matrizes globais, são aplicadas as condições de contorno em ambos os membros da equação que são então solucionadas. Obtidos resultados, eles são gravados no arquivo de saída e é checada a convergência das variáveis para checar se o problema já atingiu o regime permanente. Se o regime permanente foi atingido então a gravação da saída é encerrada e o arquivo de resultados é fechado, caso contrário é checado se foi atingido o número máximo de iterações configuradas para se decidir pelo fim da simulação ou pela execução de uma nova chamada de iteração do modelo `run_iteration`.

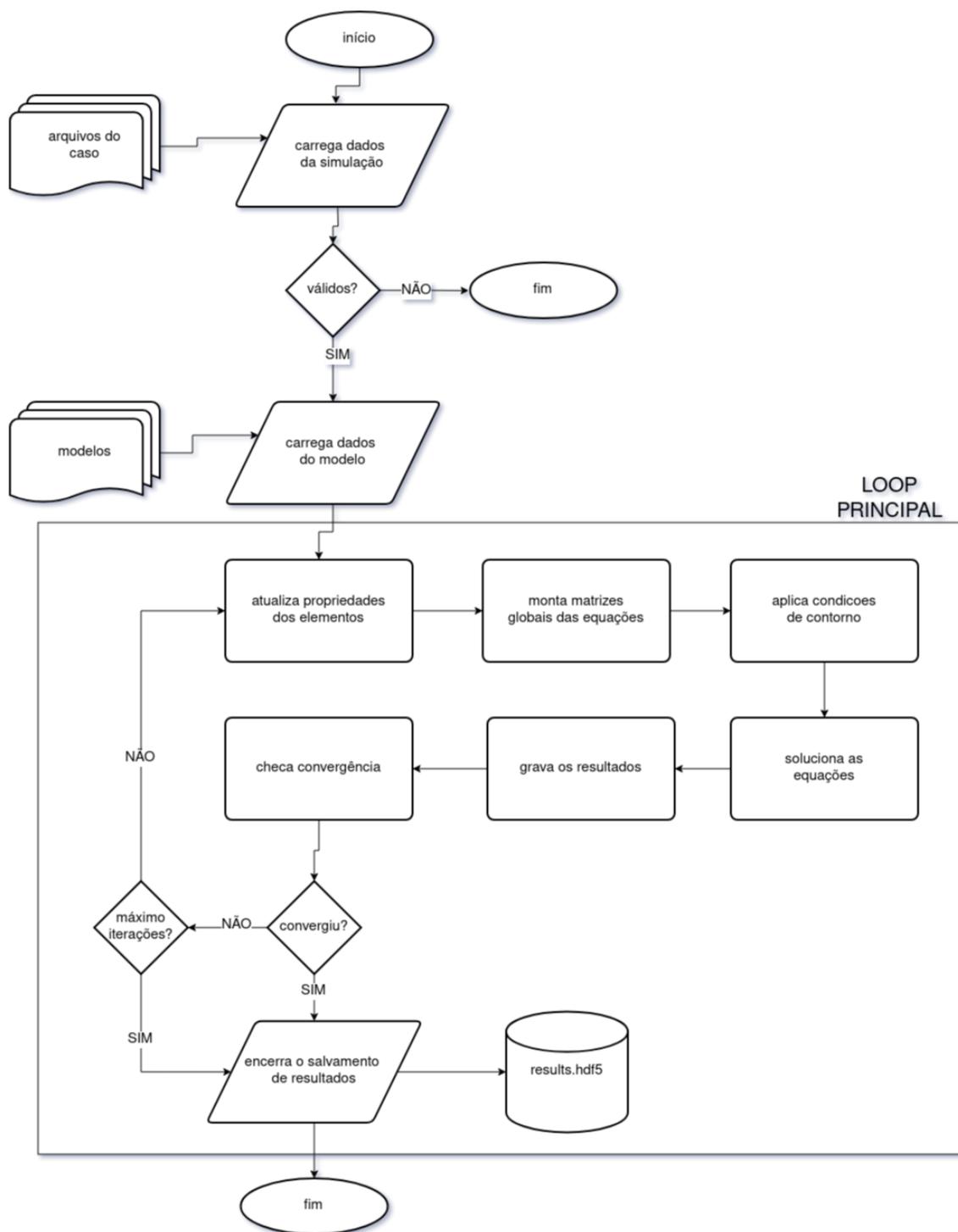


Figura 4.1: Fluxo de execução simplificado do simulador.

4.3 Estrutura do código

Como já mencionado, o código foi estruturando tendo em vista a modularidade das partes constituintes e com a ideia de reutilização das unidades do programa e facilitar a execução dos testes e basicamente a estrutura de diretórios do programa tenta refletir essa diretriz. De forma resumida, os arquivos do código estão dispostos da seguinte forma:

1. `/.pixi` diretório criado e gerenciado automaticamente pelo Pixi. Possui todos os arquivos de instalações de dependências do projeto, programas, bibliotecas, etc.
2. `/docs` arquivos de documentação geral do projeto
3. `/simulator/src` possui os arquivos de código-fonte em si incluindo:
 - (a) `/core` com todas o código do núcleo do simulador, usados pelos demais módulos do programa
 - (b) `/mesh` possui as rotinas de importação de dados da malha, criação dos nós e elementos da malha e de cálculos de propriedades dos elementos
 - (c) `/models` onde reside a descrição das equações a serem resolvidas pelos modelos usados pelo simulador e rotinas de montagem no nível dos elementos
 - (d) `/schema` possui o esquema dos arquivos de entrada do simulador, bem como os validadores lógico das entradas do simulador
4. `/simulator/test` inclui todos os arquivos de testes unitários e testes de caso do programa
5. `main.jl` ponto de entrada para execução do simulador
6. demais arquivos na raiz do projeto importantes para o desenvolvimento do projeto

O núcleo do simulador é a parte com as funções de base e é a parte mais importante do programa porque reúne todas as funcionalidades básicas necessárias para executar a simulação. Essencialmente o núcleo do simulador é responsável por:

1. montar as matrizes
2. montar a estrutura com dados do caso de teste
3. montar as condições de contorno a partir de arquivos de entrada

4. selecionar o modelo e método de solução
5. gravar as saídas do programa
6. rodar o *loop* principal do simulador
7. solucionar as equações matriciais
8. gerenciar e atualizar as variáveis do modelo corrente

O diretório que contém os modelos também tem importância fundamental. Dentro dele vão subdiretórios com, cada um deles, um conjunto de arquivos contendo a descrição das montagens das equações a partir da montagem dos elementos individuais usados pelo modelo. O simulador usa o modelo selecionado no arquivo de entrada para a simulação e trata de forma genérica a montagem das matrizes a partir das funções de montagem configuradas nos arquivos do diretório do modelo. Isso implica que a adição de novos modelos pressupõe a inclusão de um novo diretório com a montagem das equações para o novo modelo e a inclusão no arquivo de registro de modelos é o que precisa ser feito; não é necessário alterar nada no núcleo do programa.

Já o que diz respeito às validações dos dados de entrada, há o código contido no diretório `/simulator/src/schema` é responsável justamente por definir a estrutura dos arquivos de entrada da simulação e validar os arquivos de entrada para que se certifique que estes possuem valores coerentes para cada um dos campos desses arquivos e evitar que o simulador encerre sua execução com uma falha devido as entradas erradas.

4.4 Arquivos de entrada

O usuário interessado em usar o programa deve criar um caso de simulação que não é mais que um diretório com três arquivos contendo a malha no formato `.msh` (criado pelo Gmsh), arquivo de condições de iniciais/contorno do problema e um arquivo com os dados gerais da simulação de entrada do simulador dentro de um diretório com nome desejado para o caso. O simulador é capaz de importar somente arquivos do tipo `.msh`, gerados pelo Gmsh, o nome do arquivo é arbitrário, ele é informado dentro do arquivo de entradas.

O arquivo das condições de contorno `conditions.toml` possui informações gerais do caso e uma lista de condições iniciais, contendo cada item da lista:

1. `description` uma breve descrição da condição (opcional)
2. `group_name` nome do grupo de nós a que se deve aplicar a condição

3. `value` valor aplicado à variável
4. `unknown` nome da variável a que se deve aplicar o valor

Esse mesmo arquivo também possui a lista com as condições de contorno em que cada item da lista possui os dados de:

1. `description` uma breve descrição da condição (opcional)
2. `group_name` nome do grupo de nós a que se deve aplicar a condição
3. `condition_type` o tipo de condição de contorno, no momento o simulador suporta apenas a aplicação de condições de contorno do tipo 1 (Dirichlet)
4. `unknown` nome da variável a que se deve aplicar o valor
5. `value` valor aplicado à variável

O usuário pode criar este arquivo de condições de contorno manualmente ou, a depender da complexidade da malha, elaborar um *script* para criar o arquivo automaticamente.

Esse arquivo com os dados da simulação `simulation.toml` deve ser criado, por exemplo, copiando um arquivo de algum outro caso e editando manualmente os dados contidos nele. O arquivo de simulação possui dados usados para a simulação e possui as seguintes seções:

1. `general` metadados da simulação
2. `simulation` contém dados sobre o modelo usado, número máximo de passos de tempo, se a simulação deve ser executada no modo transiente, e nome do arquivo de malha. Também possui subseções para selecionar a tolerância usada para definir a convergência no cálculo de cada uma das variáveis.
3. `parameter` seleciona parâmetros (e.g. números adimensionais) importantes para a simulação em questão
4. `solver` inclui dados para o solver usado, por exemplo, o tipo de solver, pre-condicionador e tolerâncias da convergência
5. `output` contém dados sobre quais tipos de dados são gravados na saída, se são gravados e com qual frequência de salvamento

4.5 Arquivos de saída

Os dados de saída do simulador são escritos de acordo com o que foi configurado pelo usuário nos arquivos de entrada da simulação e são exportados em um arquivo binário `.hdf5`. O arquivo de saída possui basicamente a seguinte estrutura:

1. `version` versão do arquivo de saída
2. `description` breve descrição da simulação
3. `result/<unknown>/t_<timestep>` são listas baseadas no nome da variável cujo valor está sendo salvo na saída e o passo de tempo, este vai de zero ao passo de tempo final da simulação.
4. `convergence/<unknown>/t_<timestep>` são listas com dados de convergência baseadas no nome da variável calculada em determinado passo de tempo para determinada variável; o passo de tempo vai de zero até N para o passo de tempo final da simulação.
5. `success` status de sucesso da simulação, verdadeiro se tudo ocorreu bem ou falso caso contrário
6. `total_steps` número total de passos de tempo computados durante a simulação
7. `total_elapsed_time` tempo total transcorrido do início ao final da simulação
8. `mesh/nodes/positions/t_<current_step>` salva a matriz com a posição dos nós da malha; para malhas estáticas salva somente a malha no momento inicial
9. `mesh/<element_name>/connectivity/t_<current_step>` salva a matriz de conectividade para cada grupo de elementos; se a malha for híbrida, por exemplo de quadriláteros e triângulos, então ele salva separadamente a conectividade dos triângulos e quadriláteros como `mesh/triangles/connectivity/t_<current_step>` e `mesh/quadrilaterals/connectivity/t_<current_step>`, respectivamente.

Depois de ser criado esse arquivo pode ser lido por qualquer software ou biblioteca que seja capaz de interpretar os dados HDF5 e o usuário pode ler os campos de interesse para realizar algum tipo de pós-processamento e análise.

4.6 Testes unitários e testes de caso

Todo o desenvolvimento do projeto foi orientado a testes, muitas vezes escrevendo-os antes mesmo das implementações em si, seguindo a metodologia de *Test Driven Development* (TDD) [12]. Ter uma boa quantidade de testes é fundamental, sobretudo para esse tipo de aplicação para garantir que as partes não funcionam como esperado, mesmo depois de alterações em partes do código ou adição de novas funcionalidades. O simulador possui mais de 200 testes unitários, cobrindo a maior parte das funções implementadas. Os testes são rodados automaticamente executando o *script* para executar todos os testes

```
pixi run test-unit -a
```

ou somente pode-se executar um grupo específico de testes executando

```
pixi run test-unit nome_do_grupo_de_testes
```

Além dos testes unitários, há também os chamados testes de casos, que são casos com resultados conhecidos que ficam salvos dentro do subdiretório **reference**. Os testes adicionados como testes de casos serão apresentados na seção de resultados deste texto e servem para garantir o funcionamento correto do programa, por exemplo como forma de checar se após determinada modificação no programa ele ainda é capaz de gerar o resultado correto ou se, por outro lado, a modificação gerou algum efeito colateral que faça o simulador não mais se comportar como esperado fazendo-o gerar resultados inesperados. De forma análoga aos testes unitários, é possível também executar todos os testes de caso de uma vez fazendo:

```
pixi run test-case -a
```

ou somente pode-se executar um grupo específico de testes executando

```
pixi run test-unit nome_do_grupo_de_testes
```

Capítulo 5

Resultados

5.1 Montagem dos experimentos

A máquina utilizada para realização dos testes de caso de validação/verificação possui um processador Ryzen 7 5700x com 8 núcleos e 16 *threads*, executando o simulador em única *thread*, e 16 GB de memória RAM Kingston DDR4 com frequência de 3200MHz. A máquina utilizada tem instalado Linux Arch com kernel 6.11.6-zen1-1-zen como sistema operacional e a versão utilizada da linguagem Julia é a versão 1.10.2 e do Pixi usado no ambiente é 0.27.1.

Todos os testes foram montados utilizando o Gmsh como gerador de malha, a partir de *script* que faz uso da API de código Julia do Gmsh para montagem da malha e como facilitador da montagem das condições de contorno quando foi conveniente. O arquivo de dados da simulação foi então preenchido para cada uma das simulações. As simulações foram então executadas e o arquivo de saída foi então aberto com um outro *script* de pós-processamento com objetivo de checar os resultados e plotar os resultados apresentados nas seções seguintes.

As simulações foram conduzidas utilizando o método CBS semi-implícito implementado atualmente no simulador e com resultados para o regime permanente. Considera-se que o fluido é Newtoniano e escoamento incompressível isotérmico de modo que o valor da velocidade c é considerada infinita de modo que o termo M_p da equação do passo 2 é desprezado, além disso a equação de energia do passo 4 não é resolvida. O valor das constantes $\theta_1 = \theta_2 = 1$ para todas as simulações.

O valor das variáveis a seguir referem-se a grandezas adimensionais de tempo e de velocidade. Além disso, a medida usada para quantificar os desvios foi o erro na forma da norma L2, representado por $\|\epsilon\|_2$, com relação aos dados de referência simulados em trabalhos passados.

5.2 Caso do escoamento entre placas

O presente caso, também conhecido como caso do canal, é o caso clássico e mais básico de escoamento de fluido no interior de um região delimitada por placas planas. As condições de contorno impostas estão descritas na Figura 5.2, assim como as dimensões do canal. Aqui aplica-se condição de velocidade horizontal unitária na entrada do canal (à esquerda) e pressão de referência nula no final do canal (à direita); as paredes superior e inferior possuem condição de não-eskorregamento. O escoamento se dá a partir da esquerda até seu fim do lado direito (vide Figura 5.2) e o perfil das velocidades passa a tomar forma parabólica a medida que o fluido percorre o canal até finalmente assumir sua forma do perfil final quando o escoamento está totalmente desenvolvido. Este caso foi configurado para executar com $Re = 100$, com 3562 elementos triangulares e 1780 nós e sua execução completa durou 9 segundos. A malha está indicada na Figura 5.1.

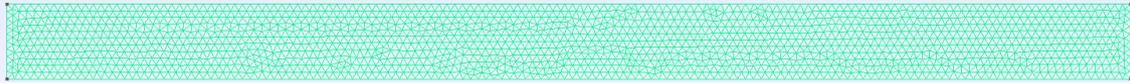


Figura 5.1: Malha utilizada no caso do escoamento entre placas. 3562 elementos triangulares e 1780 nós.

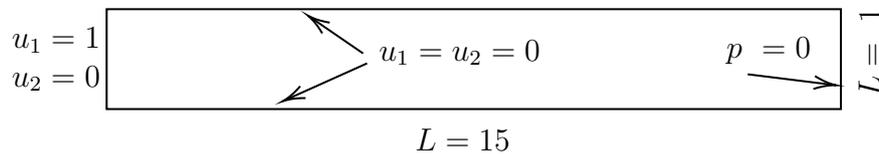


Figura 5.2: Representação das condições de contorno para o caso do canal. Comprimentos L adimensionais.

O mapa de velocidades da Figura 5.3 traz a forma do escoamento com padrão parabólico, acelerando a velocidade no centro conforme o escoamento se desenvolve até não mais alterar o padrão de cores quando o escoamento se desenvolve completamente, de acordo com o esperado. O perfil de velocidade na direção horizontal pode ser comparado com o resultado analítico para este caso e para validação deste resultado foi utilizado os dados extraídos de NITHIARASU *et al.* [19] com a equação de $u_1 = -5,449864978(-x^2 + x)$ e estão apresentados na Figura 5.4 para o perfil de velocidades horizontal na saída do canal.

Os dados obtidos comparados na Figura 5.4 deixam claro a concordância excelente dos dados do simulador com a referência contando um erro $\|\epsilon\|_2 = 0,00536$. Portanto, indica o funcionamento correto da simulação para este caso inicial mais simples.

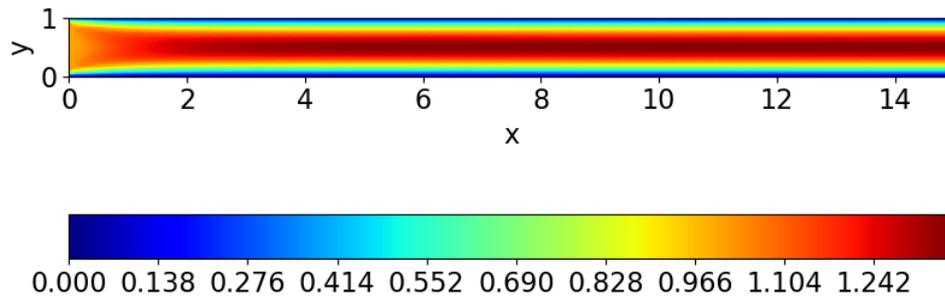


Figura 5.3: Velocidade na direção horizontal $x = 15$ para $Re = 100$.

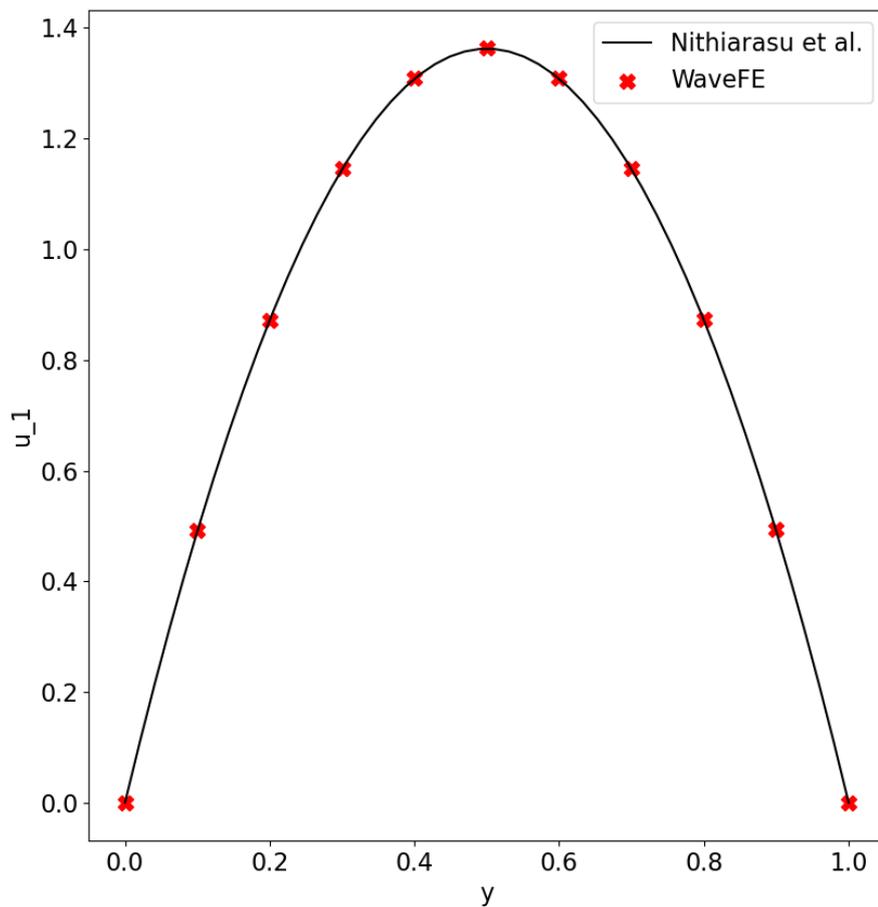


Figura 5.4: Comparação dos valores obtidos com a referência para velocidades na direção horizontal em $x = 15$ para $Re = 100$.

5.3 Caso da cavidade quadrada

Este caso consiste de uma cavidade quadrada de largura unitária preenchida de fluido. Neste problema é simulado o movimento lateral da parede superior no sentido da esquerda para a direita da figura o que provoca inicialmente um determinado fluxo que circula inicialmente no sentido horário no interior da cavidade; as demais paredes possuem condição de não-escorregamento. A Figura 5.6 mostra as condições de contorno impostas no problema. Inicialmente o fluido encontra-se em repouso e a pressão dentro da caixa foi arbitrada em $p = 0.0001$. Para a simulação foi escolhido $Re = 1000$, fluido viscoso e incompressível. A malha utilizada possui 5000 elementos triangulares e 2601 nós e o tempo total de simulação foi de 22 segundos. A malha está representada pela Figura 5.5.

Este caso clássico fornece a possibilidade de avaliar o comportamento do simulador frente à uma dinâmica bidimensional mais complexa que o caso do canal, uma vez que se espera um certos padrões de circulação de fluido no interior da cavidade análogos àquelas regiões de recirculação de fluido mostradas na Figura 2.13. A depender do número de Reynolds utilizado para a simulação, pode-se observar diferentes padrões de recirculação que devem obrigatoriamente ser capturados pela simulação.

A Figura 5.7 e a Figura 5.8 apresentam respectivamente o mapa de velocidades horizontal e vertical dentro da cavidade no regime permanente. Destaca-se na Figura 5.7 a região de maior velocidade sendo a do topo da cavidade justamente onde há a movimentação lateral do fluido na direção da esquerda para a direita do fluido. Na parte superior da cavidade observa-se as velocidades positivas e uma região de transição para velocidades de magnitude negativa o que indica que há regiões de circulação no interior da cavidade devido justamente devido à movimentação provocada em decorrência da condição de topo do domínio. Um padrão semelhante, porém se observado lateralmente, se observa para as velocidades na direção vertical da Figura 5.8.

A comparação com os resultados apresentados por GHIA *et al.* [40] estão mostrados na Figura 5.9 para o caso do perfil de velocidades na direção horizontal nas posições da linha de centro da cavidade ($x = 0,5$) e na Figura 5.10 o perfil de velocidades na direção vertical nas posições da linha central ($y = 0,5$). Dessa forma é possível investigar o perfil de velocidades em ambas as direções numa posição de referência no interior da caixa para capturar o comportamento do fluido e viabilizar a comparação com os dados de referência.

Em acordo com os mapas de velocidade, a avaliação das velocidades indicam um comportamento de transição das velocidades na direção horizontal (Figura 5.9) desde o valor zero na base até o valor unitário no seu topo, passando por velocidades

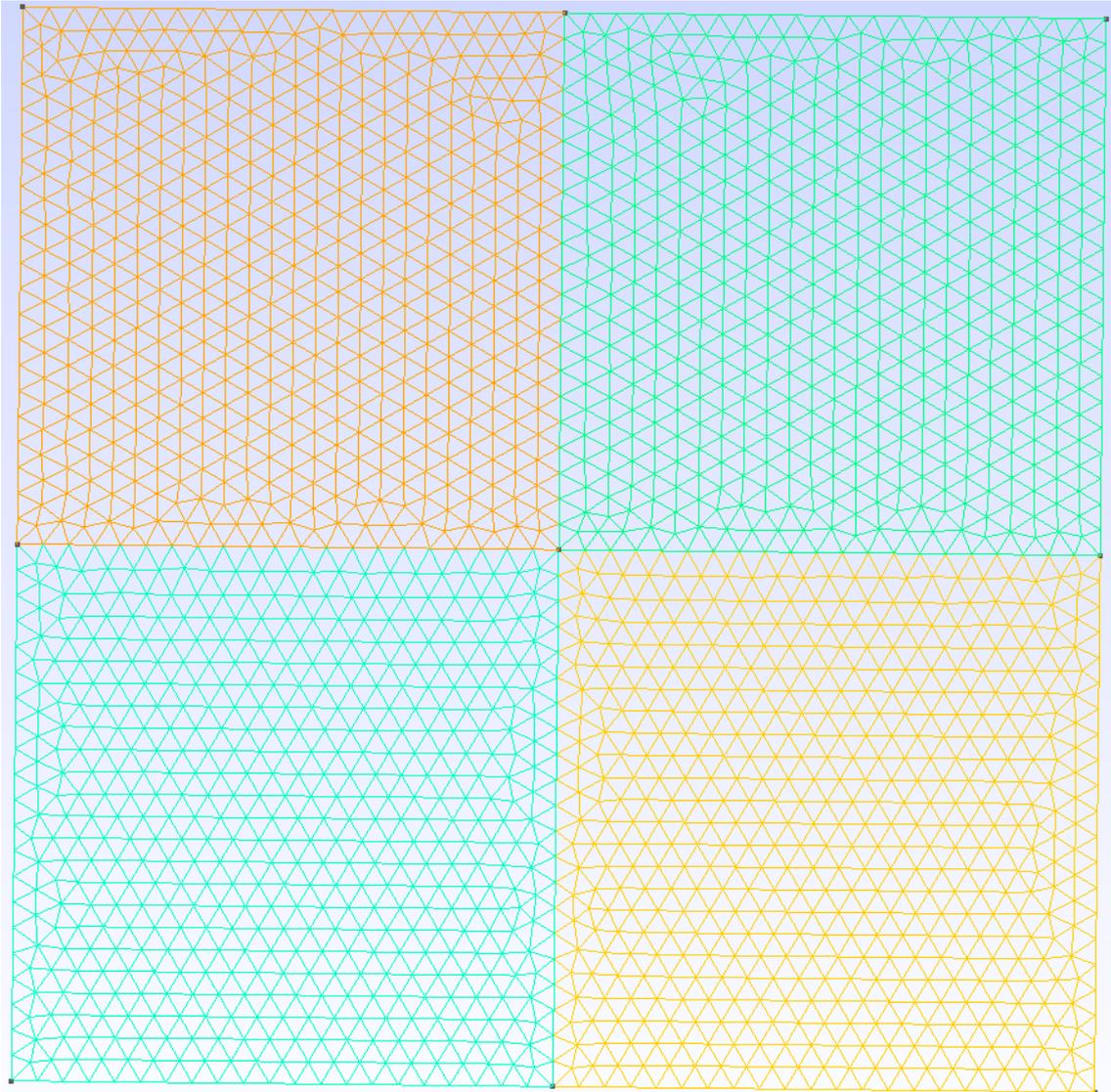


Figura 5.5: Malha utilizada no caso da cavidade quadrada. 5000 elementos triangulares e 2601 nós.

negativas na base até velocidades positivas no topo, confirmando o padrão de escoamento com regiões de circulação de fluido já mostrados no mapa de velocidades. O comportamento de transição nas velocidades verticais mostradas na Figura 5.10 também se apresentam coerentes com o mostrado no mapa de velocidades, transacionando de valores positivos para negativos e sendo zero nas extremidades esquerda e direita da caixa devido à imposição das condições de contorno. Além disso, os resultados comparados com a referência mostram uma excelente concordância, com desvio de $\|\epsilon\|_2 = 0,051$ para u_1 e de $\|\epsilon\|_2 = 0,021$ para u_2 , e mais uma vez mostra evidência da aproximação da implementação com relação à referência.

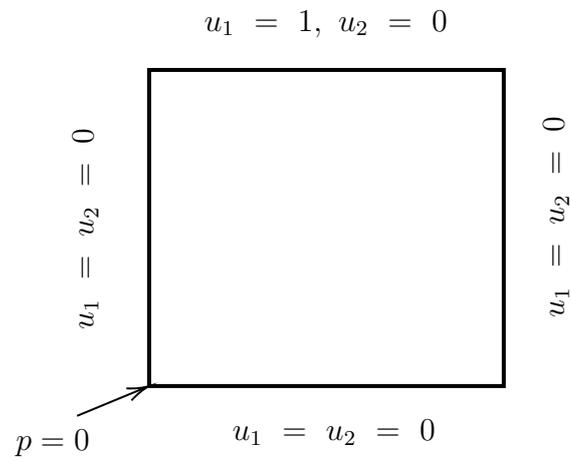


Figura 5.6: Representação das condições de contorno.

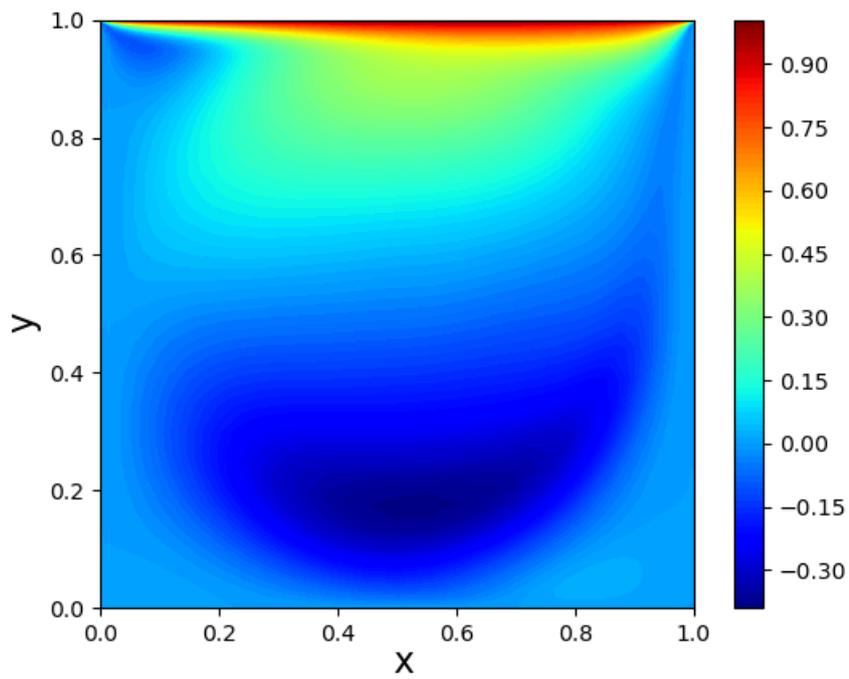


Figura 5.7: Mapa de velocidade na direção horizontal u_1 para $Re = 1000$.

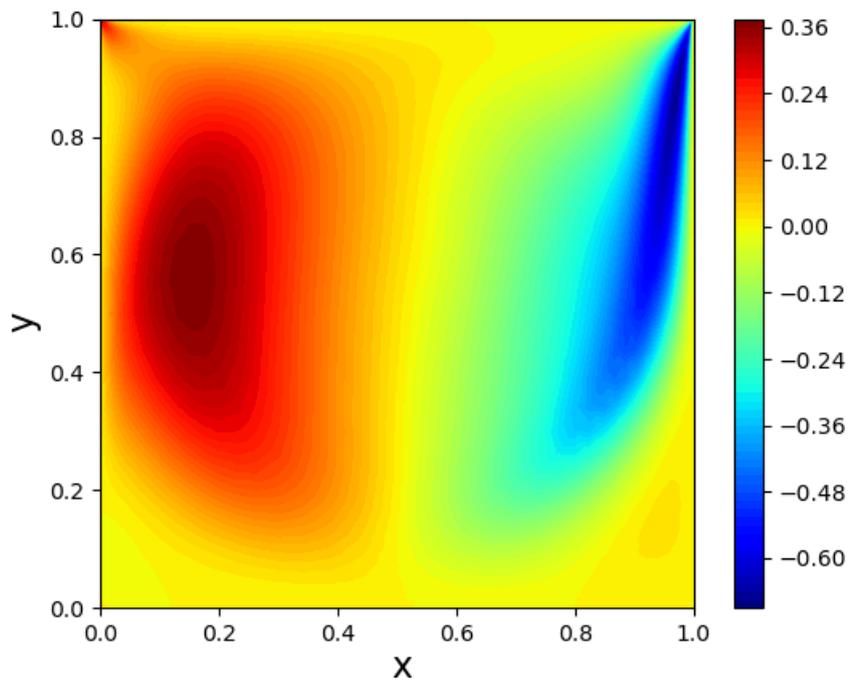


Figura 5.8: Mapa de velocidade na direção vertical u_2 para $Re = 1000$.

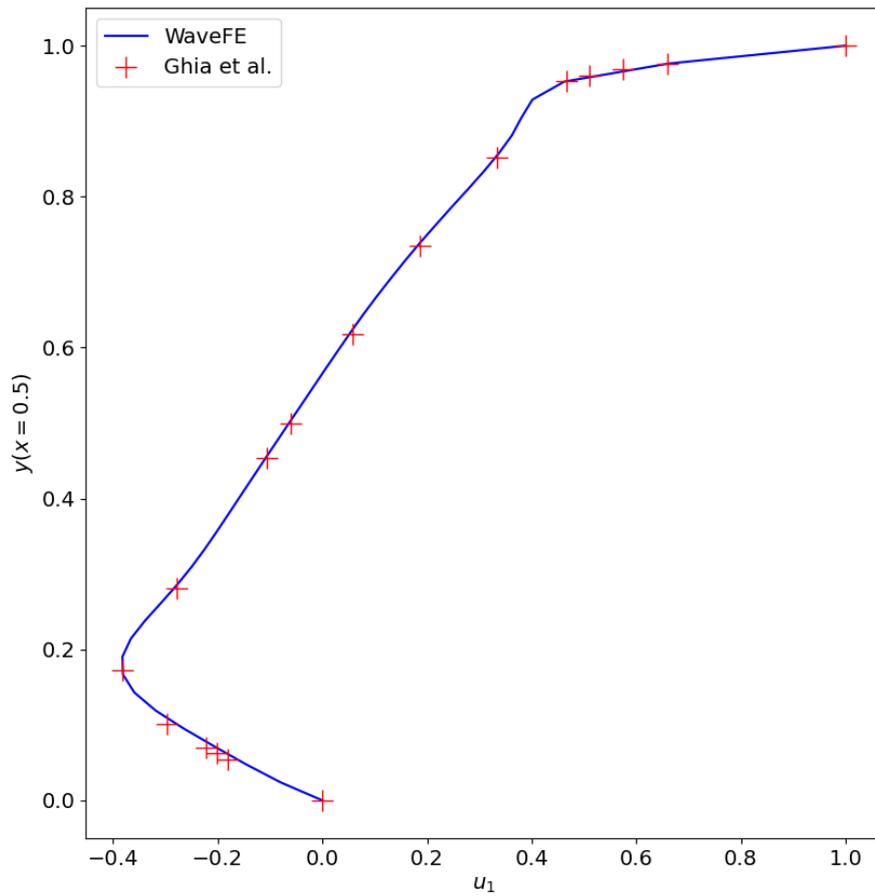


Figura 5.9: Velocidade na direção horizontal na linha central $x = 0,5$ para $Re = 1000$.

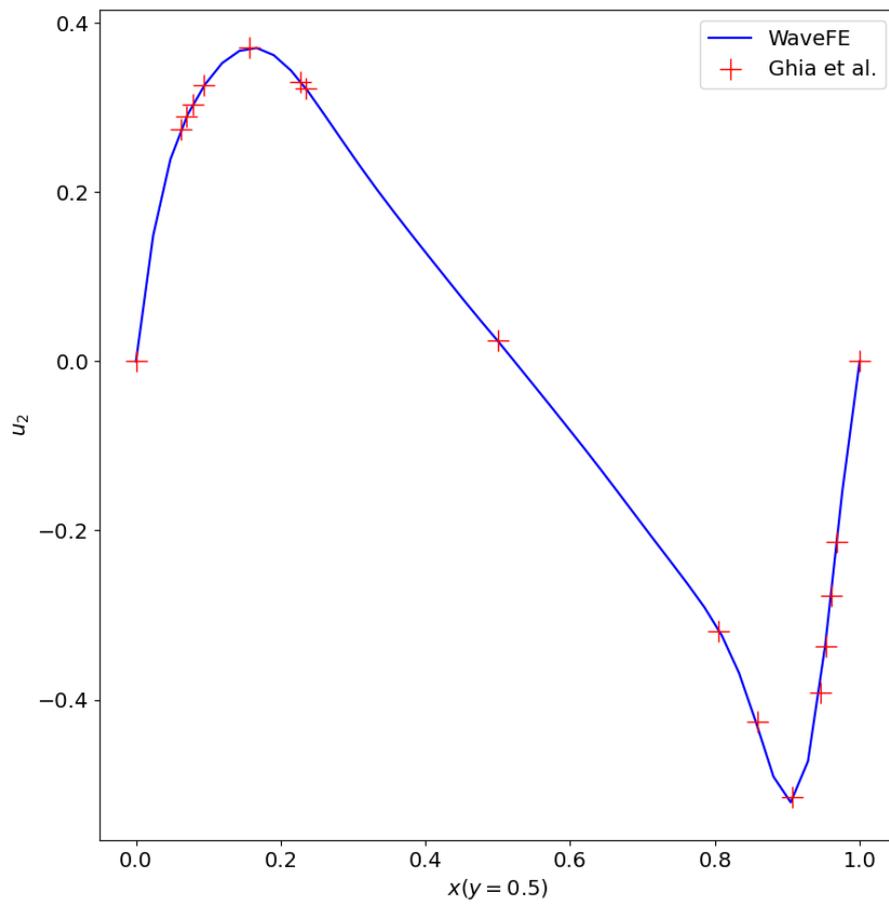


Figura 5.10: Velocidade na direção vertical na linha central $y = 0,5$ para $Re = 1000$.

5.4 Caso do degrau

Este caso também conhecido como *backward-facing step* consiste numa variação do caso do canal; aqui a geometria apresenta um desnível que faz formar o degrau que dá nome ao caso. A Figura 5.12 apresenta um esquemático com as condições de contorno definidas: imposição de um perfil parabólico na entrada do canal (lado esquerdo da figura), a pressão de referência (nula) é atribuída na saída (lado direito da figura) e às paredes superiores e inferiores atribuiu-se os valores nulos para as velocidades. O perfil parabólico como condição de contorno de entrada do canal em acordo com o utilizado na referência LIU [3] usados para validação. A malha utilizada para esse caso possui 6375 elementos e triangulares 3137 nós e o tempo total de simulação foi de 141 segundos. A malha está apresentada na Figura 5.11.

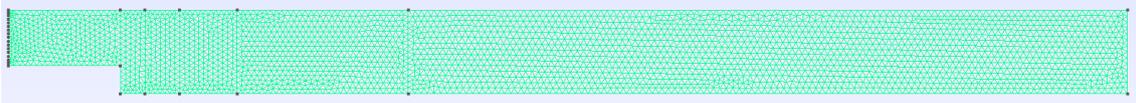


Figura 5.11: Malha utilizada no caso do degrau. 6375 elementos triangulares e 3137 nós.

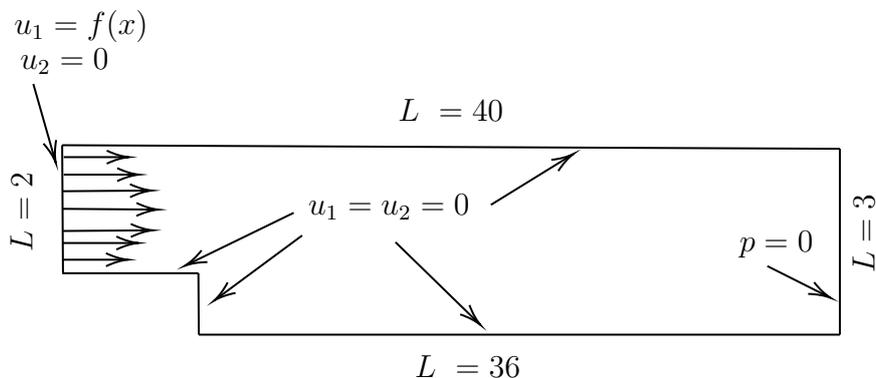


Figura 5.12: Representação das condições de contorno para o caso do degrau. Comprimentos L adimensionais.

A Figura 5.13, Figura 5.14 e Figura 5.15 apresentam o mapa de velocidade horizontal u_1 , u_2 e pressão p , respectivamente, da simulação executada com número de Reynolds de 229. O mapa de velocidades para a velocidade horizontal mostra um padrão semelhante ao observado no caso do canal para posições próximas à saída, com uma leve correção de na altura da velocidade máxima dado um perfil de velocidade qualquer após o degrau. Esse fato torna este caso estar em concordância com o caso do canal previamente discutido, ou seja, dada uma posição horizontal distante o suficiente do desnível provocado pelo degrau, o escoamento tende a se aproximar da solução do canal.

Também se observa na 5.13 a separação no escoamento desde a saída do fluido da região do degrau até aproximadamente a posição do comprimento 15 com a formação da região de circulação logo abaixo, próximo ao degrau. Já o resultado para o padrão de pressões apresentado na Figura 5.15, ele está coerente com relação ao exposto em NITHIARASU *et al.* [19], formando o mesmo padrão de baixas pressões na região do degrau e maiores pressões entre a posição horizontal de 20 a 35 .

O mapa de velocidades da velocidade vertical da Figura 5.14 indica a formação de uma região de velocidades negativas próximas ao degrau que é está de acordo com exposto por NITHIARASU *et al.* [19]. Se observa algumas regiões com padrões ondulados no final do canal, bem como pequenas regiões de coloração mais avermelhada na parte superior do domínio, próxima ao fim do degrau (próximo da coordenada $x=4$), tanto na Figura 5.14 como na Figura 5.15 e essas aparições podem indicar algum nível de instabilidade numérica nesta simulação específica, possivelmente em decorrência de instabilidade no cálculo da pressão.

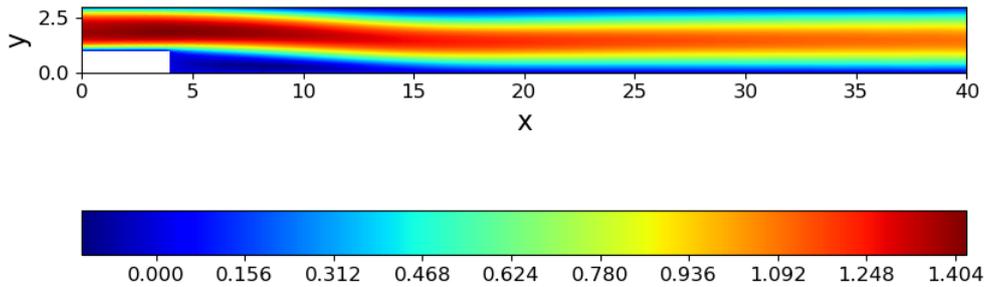


Figura 5.13: Mapa de velocidade da velocidade na direção horizontal u_1 para $Re = 229$.

Apesar da indicação de possíveis instabilidades numéricas nas velocidades na direção vertical, a comparação dos resultados com os perfis de velocidade horizontal nas posições de referência indicaram excelente concordância com apresentado em NITHIARASU *et al.* [19] e LIU [3]. A Tabela 5.1 apresenta os valores dos desvios $\|\epsilon\|_2$ para cada uma das posições medidas com relação aos valores obtidos no trabalho de LIU [3].

A Figura 5.16, Figura 5.17, Figura 5.18, Figura 5.19 e Figura 5.20 mostram os resultados da simulação comparando os perfis de velocidades horizontais em diferentes posições horizontais ao longo da geometria do domínio. Os valores de velocidade seguem a mesma tendência de valores obtidos experimentalmente DENHAM e PATRICK [41] havendo maiores desvios sobretudo no ponto de maior velocidade para alguns dos perfis e nos pontos de menor velocidade em outros.

Além disso, a Figura 5.17, Figura 5.18, Figura 5.19 apresentam valores de ve-

Tabela 5.1: Valores dos desvios em cada posição horizontal medida com relação ao apresentado em LIU [3].

posição x	$\ \epsilon\ _2$
4,00	0,054
4,88	0,066
6,11	0,097
8,17	0,084
14,29	0,121

locidades negativas na região inferior do domínio, o que está em consonância com o que se observou nos mapas de velocidade mostrados previamente em que é uma região de valores de velocidades na direção horizontal logo em seguida do desnível do degrau o que também foi medido experimentalmente. Por outro lado, a 5.20 já mostra um perfil que mais se assemelha com aquele obtido para o caso do primeiro caso de testes do escoamento entre placas paralelas, o que confirma o que se esperava: ambos os casos devem convergir para o mesmo padrão parabólico de perfil de escoamento dada uma certa distância longa o suficiente para que o escoamento se desenvolva completamente.

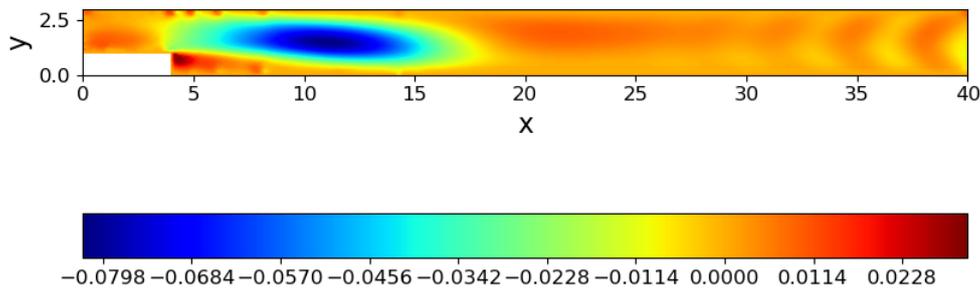


Figura 5.14: Mapa de velocidade da velocidade na direção vertical u_2 para $Re = 229$.

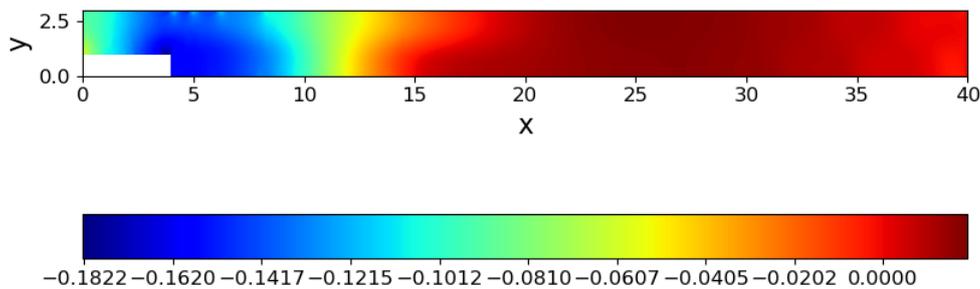


Figura 5.15: Mapa de velocidade da pressão p para $Re = 229$.

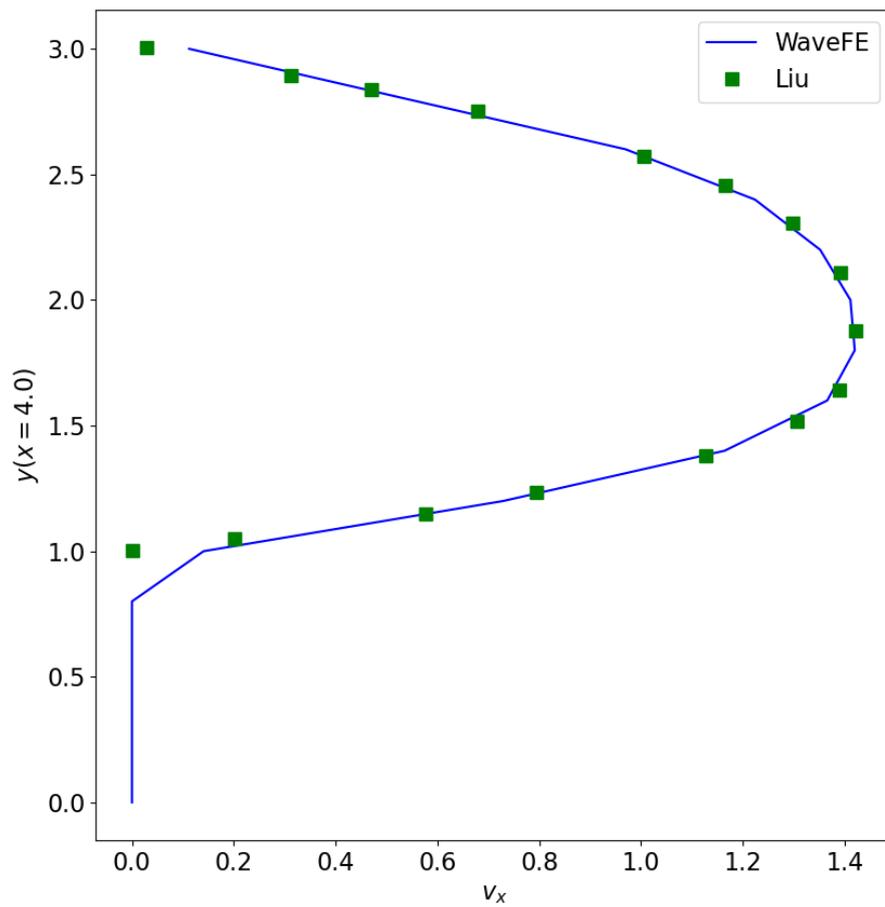


Figura 5.16: Comparação dos valores obtidos pelo simulador em comparação com os dados experimentais obtidos por LIU [3] na posição $x = 4.0$

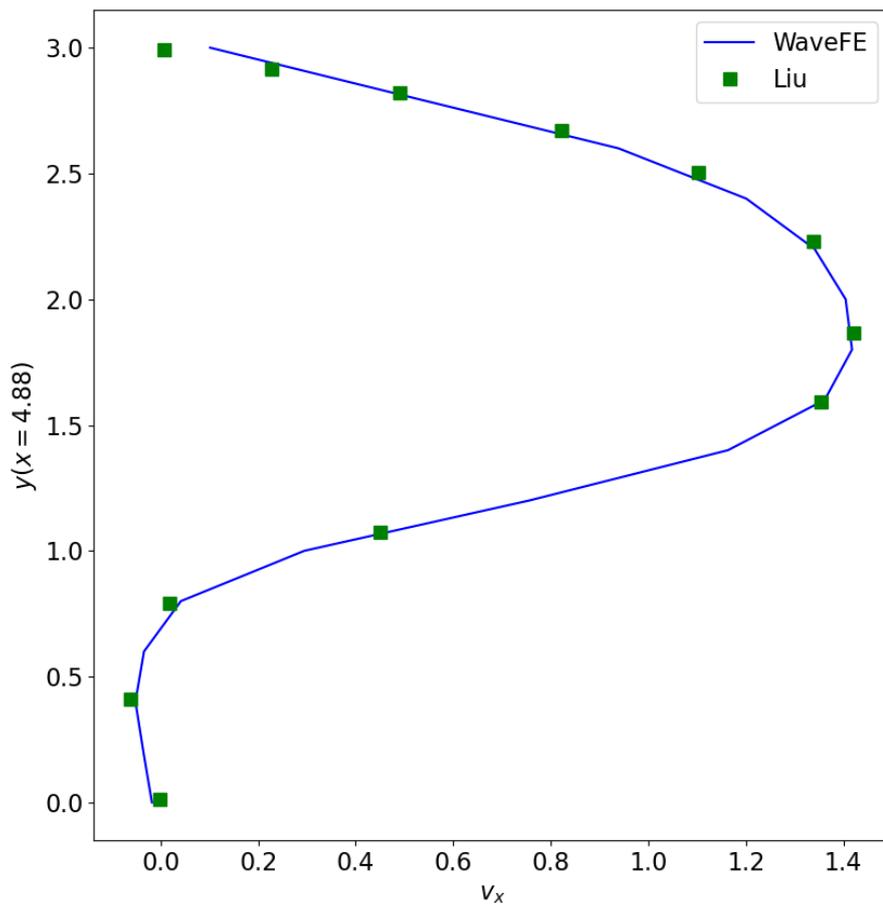


Figura 5.17: Comparação dos valores obtidos pelo simulador em comparação com os dados experimentais obtidos por LIU [3] na posição $x = 4.88$

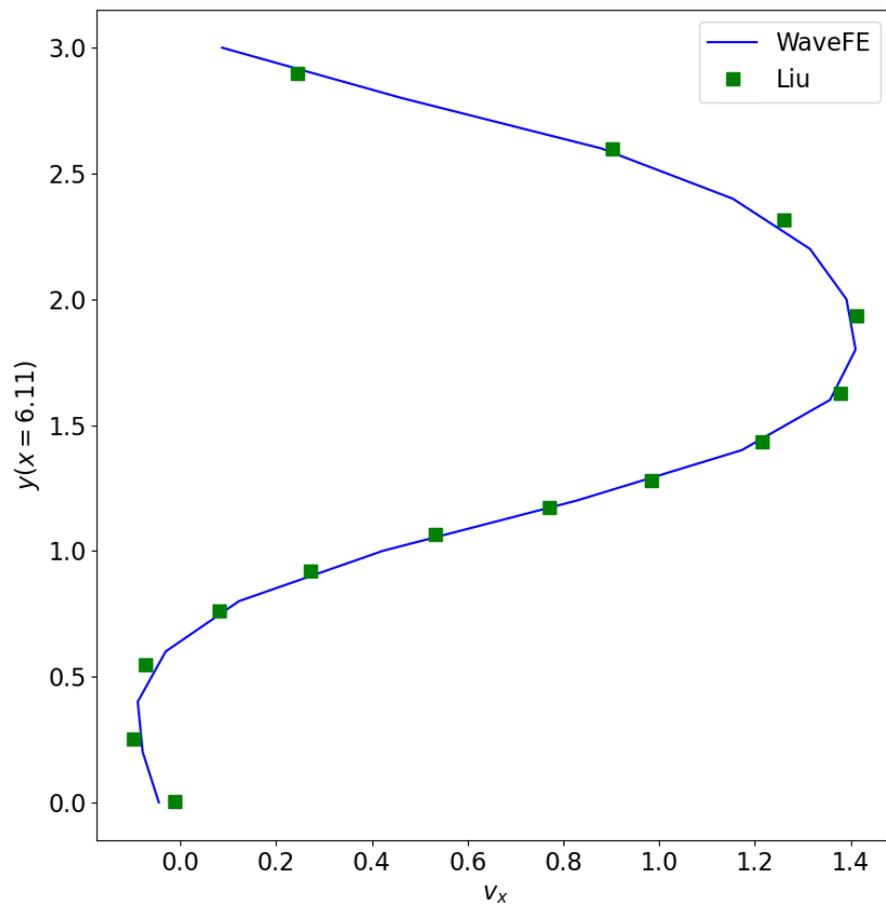


Figura 5.18: Comparação dos valores obtidos pelo simulador em comparação com os dados experimentais obtidos por LIU [3] na posição $x = 6.11$

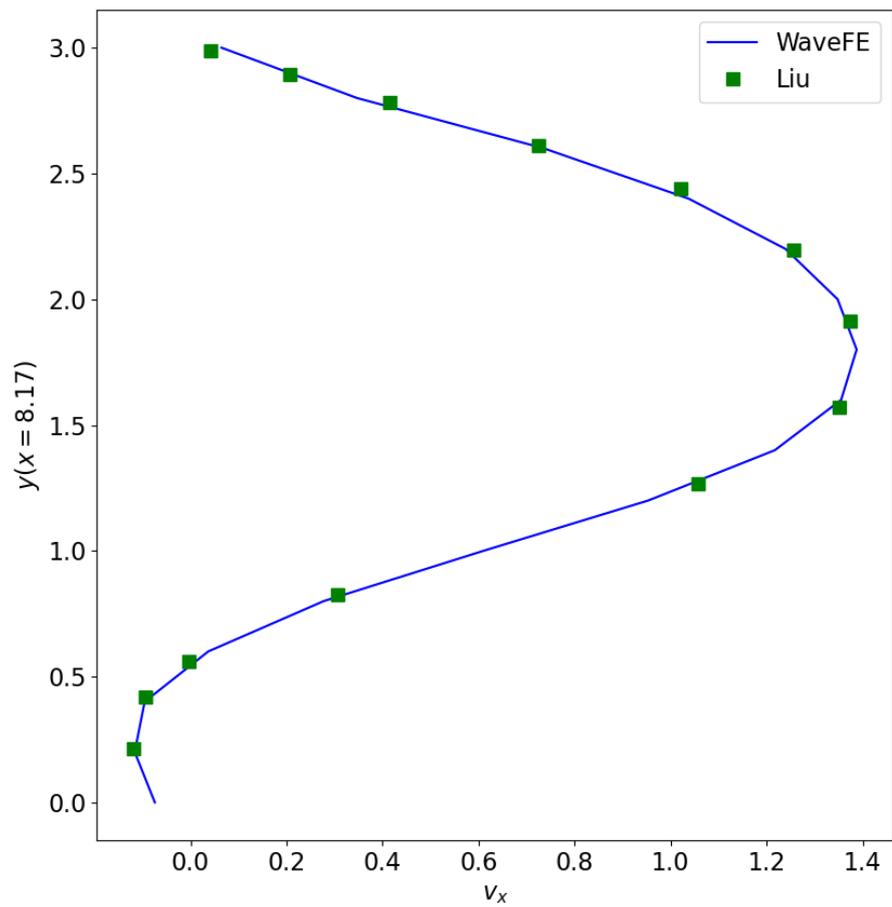


Figura 5.19: Comparação dos valores obtidos pelo simulador em comparação com os dados experimentais obtidos por LIU [3] na posição $x = 8.17$

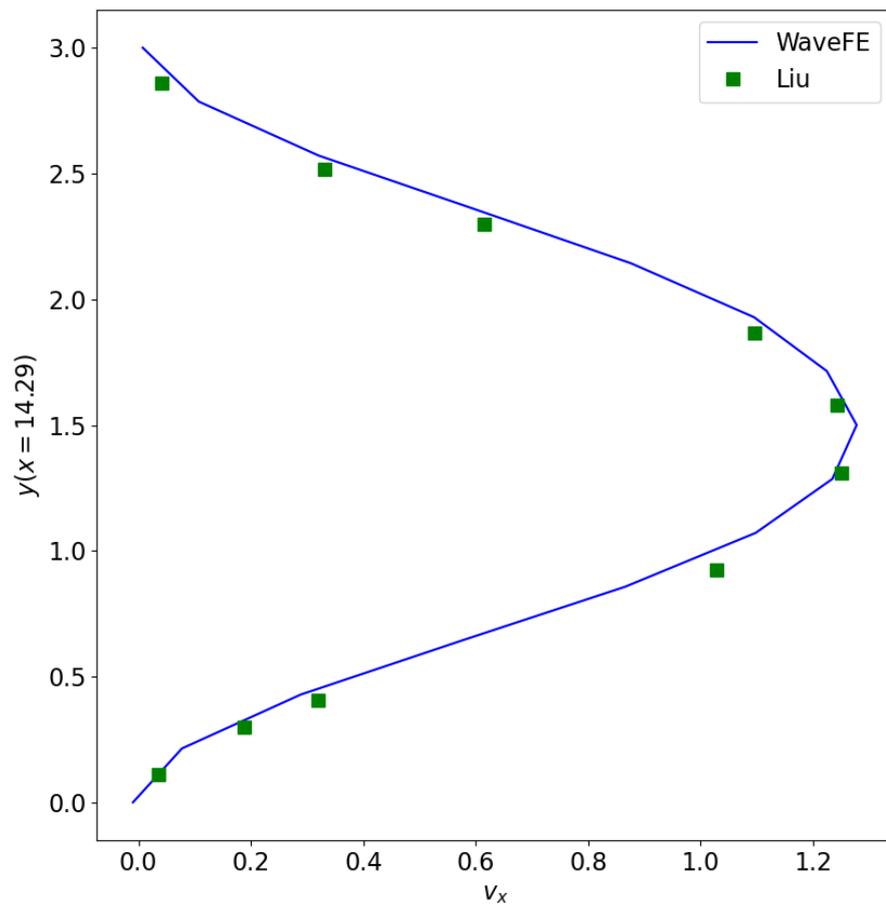


Figura 5.20: Comparação dos valores obtidos pelo simulador em comparação com os dados experimentais obtidos por LIU [3] na posição $x = 14.29$

5.5 Caso do obstáculo quadrado

Este caso consiste numa outra variação do caso do escoamento entre placas, mas com um obstáculo quadrado com posicionado no ponto médio da vertical. número de Reynolds de 100. A malha utilizada contém 3553 nós e 7114 elementos e a simulação completa durou 36 segundos. Este caso de simulação visa a replicar os mesmos resultados obtidos por PEREIRA *et al.* 1 e utilizando-se o mesmo objeto quadrado de mesmas dimensões. As condições de contorno estão mostradas na Figura 5.22 que consistem das mesmas definidas para o caso do canal acrescidas da condição de não-escorregamento na superfície do obstáculo. A Figura 5.21 mostra a malha utilizada neste caso.

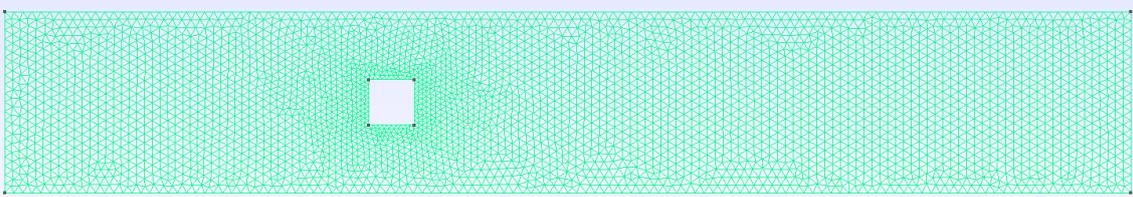


Figura 5.21: Malha utilizada no caso do obstáculo quadrado. 7114 elementos triangulares e contém 3553 nós.

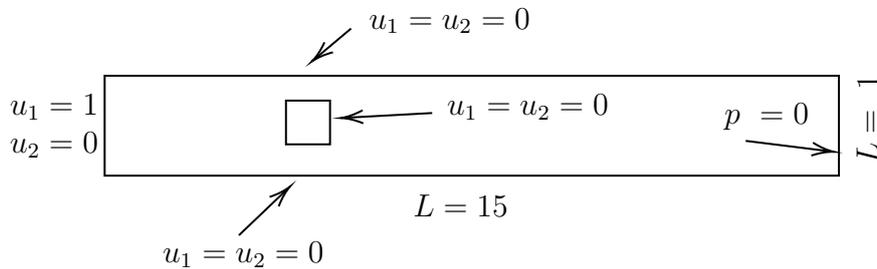


Figura 5.22: Condições de contorno impostas para o caso do obstáculo quadrado.

Este caso de validação tem como objetivo avaliar como a simulação se comporta caso haja o escoamento em torno de algum tipo de obstáculo no centro do domínio. Espera-se nesse caso que haja alguma forma de simetria em torno do objeto para o número de Reynolds selecionado e, para a geometria de obstáculo selecionada, que haja recirculação de fluido a jusante deste objeto, o mesmo padrão encontrado por PEREIRA *et al.* 1. A simetria implica as linha de fluxo no entorno do objeto tendam a desviar deste e que a linha de fluxo central siga a tendência de manter-se no centro como se atravessasse o objeto.

A Figura 5.23 apresenta as linhas de fluxo para o caso na qual se observa uma região de recirculação a jusante do obstáculo muito semelhante ao apresentado por PEREIRA *et al.* 1, com as mesmas duas linhas fechadas mostradas na Figura 2.1.

Essas linhas fechadas seguem o sentido horário para a superior e anti-horário para a região inferior o que está coerente com o esperado se pensar nas linhas de fluxo próximas obstáculo como cisalhando o fluido nessa região e portanto fazendo que haja esse padrão de circulação em circuito fechado nesta região e na linha central o fluido indo no sentido contrário ao restante das linhas de fluxo representando o escoamento, isto é, o fluido indo ao encontro do obstáculo nesta região.

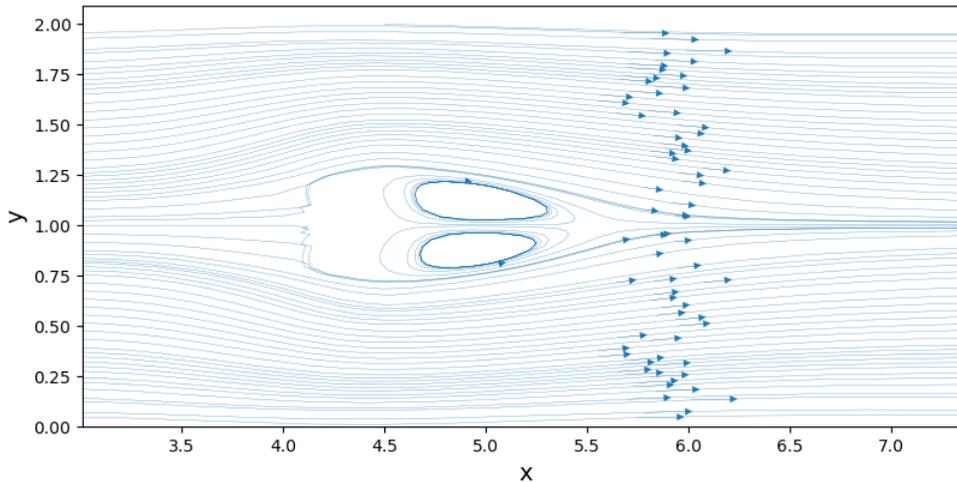


Figura 5.23: Detalhe das linhas de fluxo para o caso do obstáculo quadrado para $Re = 100$. Contorno do objeto omitido para melhor visualização do padrão de linhas.

5.6 Caso do banco de tubos

A geometria de banco de tubos simulada neste caso é a mesma da Figura 2.2. Esta geometria é uma versão simplificada do equivalente de um corte transversal em uma geometria de um alguns tipos de trocadores de calor tal qual foi apresentado na Figura 2.2, e o domínio considera apenas uma pequena região de escoamento entre tubos adjacentes representados pelos semi-círculos da mesma forma que o domínio simplificado dessa. A Figura 5.24 mostra a malha utilizada.



Figura 5.24: Malha utilizada no caso do banco de tubos. 9984 elementos e 4993 nós.

A Figura 5.25 apresenta as condições de contorno para o caso, sendo condição de entrada de velocidade na direção horizontal unitária, condição de não escorregamento para as superfícies dos tubos e pressão zero na saída do tubo. Para este caso não são definidas as condições de contorno nos demais contornos superior e inferior do domínio (que não pertencem aos tubos), ou seja, nesses pontos o escoamento pode se desenvolver livremente. A malha para este caso tem 4993 nós e 9984 elementos com comprimento e largura adimensionais assinalados na figura, exceto para os diâmetros dos tubos, estes têm diâmetro de 3 unidades. O tempo total de simulação para este caso foi de 182 segundos.

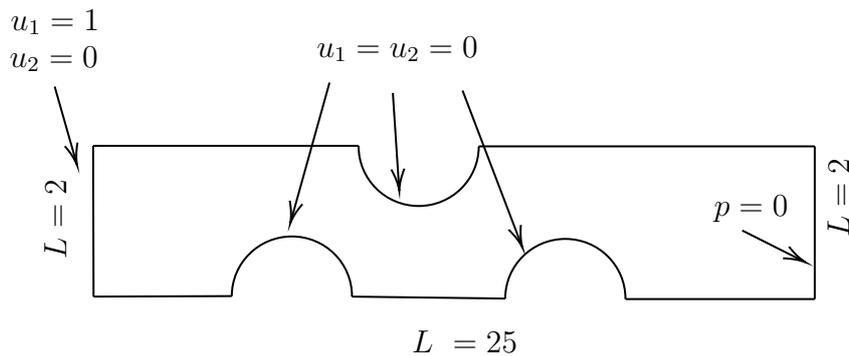


Figura 5.25: Representação das condições de contorno para o caso banco de tubos. Comprimentos L adimensionais.

O resultado do mapa de velocidades para Reynolds de 150 na direção horizontal apresentado na Figura 5.26 muito se assemelha com o padrão de linhas de fluxo obtidos por MASSAROTTI *et al.* [2] (Figura 2.3) com regiões de recirculação a jusante dos tubos. Outro aspecto que se observa um aumento do módulo das velocidades nas regiões próximas ao topo/base dos tubos, o que é esperado uma vez que há uma constricção da seção disponível para passagem do fluido e está coerente com o que se observa uma aproximação das linhas de fluxo da da referência (regiões de cor mais escura da figura). Nessas mesmas regiões de constricção se observa, a partir da Figura 5.28 que há também quedas de no valor da pressão justamente como esperado e coerente com o que é observado para os valores de velocidade horizontal. As velocidades verticais indicadas na Figura 5.27 marcam justamente os extremos da escala próximo a região dos tubos, principalmente o segundo e terceiro tubo, indicando a mudança brusca no sentido do escoamento que muda de direção para acompanhar o desvio forçado devido à geometria do problema.

O resultado obtido para este caso aumenta ainda mais o nível de evidência da correção dos resultados de simulação para casos em que há um domínio de geometria mais complexa com presença de obstáculos, o simulador foi capaz de capturar variações de velocidades em duas dimensões de forma coerente com o tipo de comportamento esperado, especialmente nas regiões entre os tubos e reproduziu os mesmos

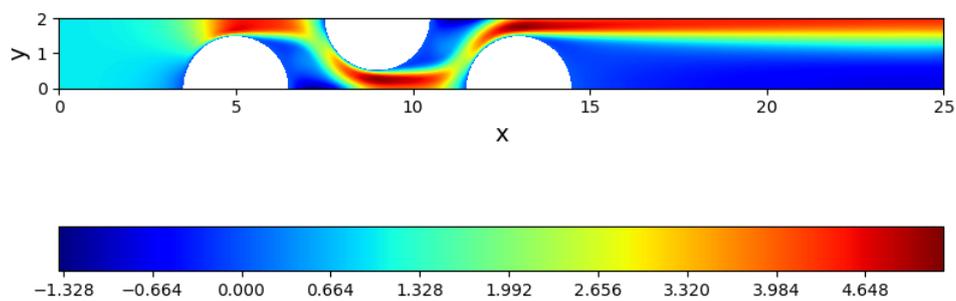


Figura 5.26: Mapas de velocidade horizontal para o caso do banco de tubos com $Re=150$.

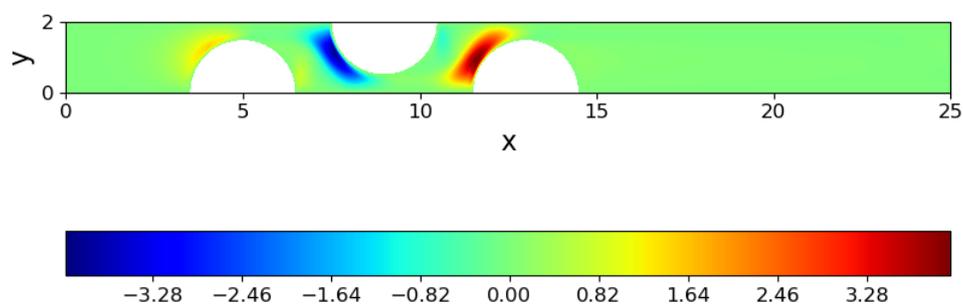


Figura 5.27: Mapas de velocidade vertical para o caso do banco de tubos com $Re=150$.

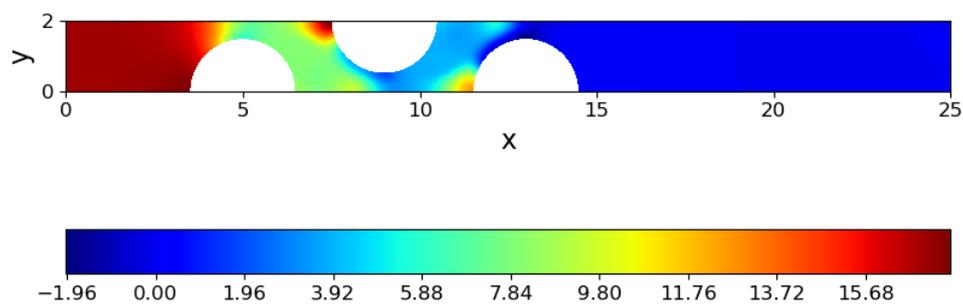


Figura 5.28: Mapas de pressão para o caso do banco de tubos com $Re=150$.

resultados de referência para este caso.

Capítulo 6

Considerações finais

6.1 Conclusões

Este trabalho apresentou os objetivos, modelagem numérica e implementação de um simulador feito em Julia para solução de problemas de escoamento de fluidos. Os resultados dos casos de testes selecionados e apresentados neste trabalho evidenciam o bom funcionamento esperado do programa, já que houve boa concordância com os dados de referência de simulações e experimentos.

Para além disso, o código resultante do simulador atende às diretrizes previamente estabelecidas para o projeto. A simplicidade do código advém da forma como foi planejada a implementação, modularizando para facilitar não somente o entendimento das unidades do programas e suas responsabilidades, como também facilitar significativamente a manutenção futura do código e adição de novas funcionalidades, sem precisar alterar o código pré-existente. Adiciona-se a isso o fato de o código estar amplamente documentado de modo a torná-lo didático à primeira vista.

A modularização do código tenderá a facilitar a extensão do simulador com novos modelos, métodos de solução, novos tipos de elementos, etc. Como todo o programa foi projetado para facilitar a leitura e novos desenvolvimentos, as partes que podem ser acrescentadas assim serão sem precisar alterar código já existente. Por exemplo, para adicionar novos elementos, deve-se adicionar as funções de montagem dos novos elementos que na prática consiste em adicionar novos arquivos e sem que seja necessário alterar nada no código de montagem global tampouco alterar código no núcleo do simulador.

Os tempos de execução do programa para os casos apresentados foram bastante satisfatórios; obter a solução em poucas dezenas de segundos de problemas da escala dos tratados aqui é um resultado bastante aceitável para uma primeira versão deste simulador. Futuramente podem ser feitas comparações dos tempos de execução desses mesmos casos em outros simuladores, para dar melhor suporte à conclusão

de que os tempos obtidos seriam igualmente aceitáveis para maioria dos usuários do simulador.

O alto número de testes unitários apoiaram o desenvolvimento do projeto no sentido de garantir o funcionamento individual de cada função, sendo testadas individualmente do restante do programa, além de fornecer vasta documentação de como são usadas as funções individualmente, que tipos de entrada se espera antes para suas chamadas e quais saídas esperadas. Isso tende a facilitar desenvolvimentos futuros com a garantia que as partes já implementadas não deixem de funcionar e, se deixarem, a execução automática dos testes indicará facilmente onde o novo código está causando problemas.

Os casos de simulação selecionados para validação do funcionamento do simulador mostraram que ele é capaz de capturar comportamento de escoamento de fluidos newtonianos em regime permanente de forma fiel aos resultados esperados fisicamente e pelos valores numéricos de experimentos físicos e computacionais. Além disso, os casos simulados com mudanças de geometria e obstáculos trazem evidência de que é possível simular geometrias mais complexas em tempo razoável e se obter o resultado correto.

Sendo assim, pode-se dizer que os objetivos e diretrizes estabelecidas para o projeto foram atendidas e o simulador apresenta resultados que indicam que ele pode ser usado no futuro como *framework* para se executar novos de experimentos computacionais com outros modelos, tipos de malha, variações do método CBS, etc.

6.2 Trabalhos futuros

Há várias possibilidade para extensão do simulador para que ele possa, por exemplo, solucionar problemas baseados em novos modelos, baseados em variações do método CBS ou utilizar outros tipos de malha. Várias dessas novas funcionalidades e melhorias podem ser adicionadas ao programa desenvolvido neste trabalho e estão listadas resumidamente nas próximas seções. Os detalhes com as instruções para a implementação dessas melhorias e extensões do simulador estão descritas no próprio repositório do código ¹ no diretório `/docs`.

6.2.1 Implementação de novos tipos de malha

Trabalhos futuros poderiam empenhar tempo na implementação de algumas novas possibilidades de elementos usados na malha, configurando funções auxiliares e de montagem para os novos tipos de elementos, outros formatos de arquivo de malha, etc. Algumas sugestões de implementação seriam:

¹<https://github.com/gabrielantao/WaveFE>

1. adição de outros tipos de elementos: segmentos, quadriláteros, tetraedros, hexaedros, etc.
2. adição de malha híbrida (e.g. elementos triangulares e quadrangulares)
3. adição de elementos de alta ordem de interpolação (quadrática, cúbica)
4. implementação de outros tipos de formatos de malhas além das geradas pelo Gmsh
5. implementação de *scripts* e ou interface para melhor integração com o gerador de malhas e demais parâmetros

A arquitetura do simulador permite a adição de novos elementos: basta criar os **structs** que descrevem as propriedades desses novos elementos, bem como adicionar o conteúdo dessas funções. O esqueleto das funções que implementam rotinas auxiliares e de montagem desses elementos já estão incluídas no simulador devendo apenas ser preenchidas com a implementação das rotinas de montagem e cálculos adicionais (cálculo de áreas e outras propriedades). O elemento triangular pode servir como exemplo para implementação dos novos elementos.

Além dos novos elementos, pode ser implementada a importação de outros formatos de malha por meio da conversão dos dados de entrada no formato do Gmsh, ou mesmo convertendo-os para um formato comum, interno do simulador. *Scripts* que façam melhor integração para, a partir da malha gerada, definir as condições de contorno e demais parâmetros com objetivo de criar os arquivos de simulação necessários.

6.2.2 Implementação de novos modelos

Por fim, trabalhos derivados podem acrescentar novos modelos ao modelo existente atualmente para resolver problemas de outros domínios físicos, tais como:

1. modelos que envolvam problemas com transferência de calor
2. modelos que envolvam transporte de espécies químicas
3. modelos que envolvam meios porosos

NITHIARASU *et al.* [19] dedicam um capítulo para cada tipo de problema desses domínios e descreve bem e a formulação dos modelos além de trazer alguns exemplos de casos de testes para validação/verificação das soluções. Os trabalhos apresentados na revisão de literatura deste trabalho podem ser usados como ponto de partida para implementação de novos métodos e modelos de solução. Os resultados mostrados naqueles trabalhos poderão ser reproduzidos dentro do simulador.

A implementação dos novos modelos devem envolver majoritariamente a inclusão das funções de montagem para os elementos disponíveis e a inclusão de um arquivo com uma função `run_iteration` (representa uma iteração do loop principal) e que eventualmente faça operações exclusivas daquele modelo.

6.2.3 Variantes do método CBS

Algumas variantes de solução do CBS poderiam ser implementadas e comparadas com o método semi-implícito atualmente usado pelo simulador. Essas novas variantes poderiam ser comparadas entre si para solução de um mesmo problema, por exemplo, e essa informação pode ser usada no futuro para orientar a escolha do método que se espera melhorar a solução de novos tipos de problemas. Algumas sugestões seriam:

1. implementação da solução sem usar matrizes amortecidas para matriz de massas
2. implementação de casos transientes com CBS, ou seja, sem utilizar o artifício de tempos locais dos elementos
3. implementação de solução de problemas com malhas móveis a partir da técnica de ALE-CBS
4. implementação de solução envolvendo a técnica de compressibilidade artificial

O simulador foi projetado de modo que a já estar preparado pra receber essas novas funcionalidades sem que necessite maiores mudanças senão a implementação da funcionalidade em si. Por exemplo, quanto as malhas móveis, já existe uma função `move!` que recebe os dados de posição, velocidades, e aceleração dos nós, mas atualmente não executa nenhuma operação de movimento, ou seja, será necessário realizar os cálculos pra se determinar novas posições por exemplo a partir das forças atuantes nesses nós em determinado tempo. Essas implementações por si podem ser mais laboriosas porque devem exigir um estudo mais aprofundado de convergência das solução e adição de uma certa quantidade de testes, além disso podem exigir vez que elas podem demandar algum nível de reorganização do código para garantir o máximo o reaproveitamento de código e o respeito à arquitetura existente.

6.2.4 Melhorias de performance do simulador

Outra vertente de trabalhos futuros pode contemplar melhorias pontuais ou gerais de performance do simulador, assim como inclusão de testes e comparações de performance através de testes de *benchmarks* e avaliar consumo de recursos para solução de problemas similares em outros simuladores.

1. adição de novos *solvers* e condicionadores para matrizes das equações resolvidas
2. implementação de operações com matrizes simétricas para reduzir o custo de memória para as operações *solver*
3. implementação de algumas operações em paralelo, por exemplo, em algumas partes da operação de montagem das matrizes
4. perfilar o código para identificar *hot spots* em regiões do código que possam ser aperfeiçoadas
5. realizar *benchmarks* comparando resultados e desempenhos com outros simuladores

Essas tarefas pressupõem a avaliação do desempenho dos casos simulação aqui apresentados, ou mesmo adicionando novos casos eventualmente com malhas maiores pra que melhor sobressaiam os gargalo nas rotinas atualmente implementadas no simulador.

Referências Bibliográficas

- [1] PEREIRA, V., ROMAO, E., SILVA, J., et al. “Numerical simulation of incompressible flows by the stabilized finite element method”, *Engenharia Térmica*, v. 12, n. 1, pp. 67–79, 2013.
- [2] MASSAROTTI, N., NITHIARASU, P., ZIENKIEWICZ, O. C. “Characteristic-based-split (CBS) algorithm for incompressible flow problems with heat transfer”, *International Journal of Numerical Methods for Heat & Fluid Flow*, v. 8, n. 8, pp. 969–990, 1998. doi: 10.1108/09615539810244067. Disponível em: <<https://doi.org/10.1108/09615539810244067>>.
- [3] LIU, C.-B. “The Characteristic Based Split (CBS) scheme for laminar and turbulent incompressible flow simulations”. 2005. Disponível em: <<https://api.semanticscholar.org/CorpusID:128320253>>.
- [4] WANG, Y.-T., ZHANG, J.-Z. “An improved ALE and CBS-based finite element algorithm for analyzing flows around forced oscillating bodies”, *Finite Elements in Analysis and Design*, v. 47, n. 9, pp. 1058–1065, 2011. ISSN: 0168-874X. doi: <https://doi.org/10.1016/j.finel.2011.03.021>. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0168874X11000710>>.
- [5] JIN, L., SHEN, H. “Projection- and characteristic-based operator-splitting simulation of mixed convection flow coupling heat transfer and fluid flow in a lid-driven square cavity”, *Numerical Heat Transfer, Part B: Fundamentals*, v. 70, n. 4, pp. 354–371, 2016. doi: 10.1080/10407790.2016.1215694. Disponível em: <<https://doi.org/10.1080/10407790.2016.1215694>>.
- [6] TAN, L., ZHU, B., WANG, Y., et al. “Turbulent flow simulation using large eddy simulation combined with characteristic-based split scheme”, *Computers & Fluids*, v. 94, pp. 161–172, 2014. ISSN: 0045-7930. doi: <https://doi.org/10.1016/j.compfluid.2014.01.037>. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0045793014000565>>.

- [7] BUACHART, C., KANOK-NUKULCHAI, W., ORTEGA, E., et al. “A SHALLOW WATER MODEL BY FINITE POINT METHOD”, *International Journal of Computational Methods*, v. 11, n. 01, pp. 1350047, 2014. doi: 10.1142/S0219876213500473. Disponível em: <<https://doi.org/10.1142/S0219876213500473>>.
- [8] MINH TUAN NGUYEN, A. M. A., LEE, S.-W. “Natural Convection in a Non-Darcy Porous Cavity Filled with Cu–Water Nanofluid Using the Characteristic-Based Split Procedure in Finite-Element Method”, *Numerical Heat Transfer, Part A: Applications*, v. 67, n. 2, pp. 224–247, 2015. doi: 10.1080/10407782.2014.923225. Disponível em: <<https://doi.org/10.1080/10407782.2014.923225>>.
- [9] KULKARNI, V., SEETHARMU, K., ASWATHA NARAYANA, P., et al. “Flow analysis for flip chip underfilling process using characteristic based split method”. In: *Proceedings of 6th Electronics Packaging Technology Conference (EPTC 2004) (IEEE Cat. No.04EX971)*, pp. 615–619, 2004. doi: 10.1109/EPTC.2004.1396681.
- [10] LI, X., DUAN, Q. “Meshfree iterative stabilized Taylor–Galerkin and characteristic-based split (CBS) algorithms for incompressible N–S equations”, *Computer Methods in Applied Mechanics and Engineering*, v. 195, n. 44, pp. 6125–6145, 2006. ISSN: 0045-7825. doi: <https://doi.org/10.1016/j.cma.2005.12.011>. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0045782506000247>>.
- [11] DUAN, Q., LI, X. “An ALE based iterative CBS algorithm for non-isothermal non-Newtonian flow with adaptive coupled finite element and meshfree method”, *Computer Methods in Applied Mechanics and Engineering - COMPUT METHOD APPL MECH ENG*, v. 196, pp. 4911–4933, 11 2007. doi: 10.1016/j.cma.2007.06.010.
- [12] PRESSMAN, R., MAXIM, B. *Engenharia de Software - 8ª Edição*. New York, NY, USA, 2016. ISBN: 9788580555349. Disponível em: <<https://books.google.com.br/books?id=wexzCwAAQBAJ>>.
- [13] BOYCE, W., DIPRIMA, R., MEADE, D. *Elementary Differential Equations and Boundary Value Problems*. New York, NY, USA, Wiley, 2017. ISBN: 9781119381679. Disponível em: <<https://books.google.com.br/books?id=CpBEDwAAQBAJ>>.
- [14] BROOKS, A. N., HUGHES, T. J. “Streamline upwind/Petrov-Galerkin formulations for convection dominated flows with particular emphasis on

the incompressible Navier-Stokes equations”, *Computer Methods in Applied Mechanics and Engineering*, v. 32, n. 1, pp. 199–259, 1982. ISSN: 0045-7825. doi: [https://doi.org/10.1016/0045-7825\(82\)90071-8](https://doi.org/10.1016/0045-7825(82)90071-8). Disponível em: <https://www.sciencedirect.com/science/article/pii/S0045782582900718>.

- [15] DONEA, J. “A Taylor-Galerkin method for convective transport problems”, v. 20, n. 1, pp. 101–119, jan. 1984. doi: 10.1002/nme.1620200108. Disponível em: <https://doi.org/10.1002/nme.1620200108>.
- [16] LÖHNER, R., MORGAN, K., ZIENKIEWICZ, O. C. “The solution of non-linear hyperbolic equation systems by the finite element method”, v. 4, n. 11, pp. 1043–1063, nov. 1984. doi: 10.1002/flid.1650041105. Disponível em: <https://doi.org/10.1002/flid.1650041105>.
- [17] ZIENKIEWICZ, O., TAYLOR, R. *The Finite Element Method, Vol. 3: Fluid Dynamics*, v. 3. New York, NY, USA, 01 2000.
- [18] ZIENKIEWICZ, O. C., CODINA, R. “A general algorithm for compressible and incompressible flow—Part I. the split, characteristic-based scheme”, *International Journal for Numerical Methods in Fluids*, v. 20, pp. 869–885, 1995. doi: 10.1002/FLD.1650200812. Disponível em: <https://doi.org/10.1002/flid.1650200812>.
- [19] NITHIARASU, P., LEWIS, R., SEETHARAMU, K. *Fundamentals of the finite element method for heat and mass transfer*. New York, NY, USA, 02 2015. ISBN: 978-0-470-75625-6.
- [20] ZIENKIEWICZ, O. C., MORGAN, K., SAI, B. V. K. S., et al. “A general algorithm for compressible and incompressible flow—Part II. tests on the explicit form”, *International Journal for Numerical Methods in Fluids*, v. 20, pp. 887–913, 1995.
- [21] CODINA, R., VÁZQUEZ, M., ZIENKIEWICZ, O. C. “A general algorithm for compressible and incompressible flows. Part III: The semi-implicit form”, *International Journal for Numerical Methods in Fluids*, v. 27, n. 1-4, pp. 13–32, 1998. doi: [https://doi.org/10.1002/\(SICI\)1097-0363\(199801\)27:1/4<13::AID-FLD647>3.0.CO;2-8](https://doi.org/10.1002/(SICI)1097-0363(199801)27:1/4<13::AID-FLD647>3.0.CO;2-8). Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291097-0363%28199801%2927%3A1/4%3C13%3A%3AAID-FLD647%3E3.0.CO%3B2-8>.
- [22] CHORIN, A. J. “A numerical method for solving incompressible viscous flow problems”, *Journal of Computational Physics*, v. 2, n. 1, pp. 12–26, 1967.

ISSN: 0021-9991. doi: [https://doi.org/10.1016/0021-9991\(67\)90037-X](https://doi.org/10.1016/0021-9991(67)90037-X).
Disponível em: <<https://www.sciencedirect.com/science/article/pii/002199916790037X>>.

- [23] F. ARPINO, N. MASSAROTTI, A. M., NITHIARASU, P. “Artificial Compressibility-Based CBS Scheme for the Solution of the Generalized Porous Medium Model”, *Numerical Heat Transfer, Part B: Fundamentals*, v. 55, n. 3, pp. 196–218, 2009. doi: 10.1080/10407790802628838. Disponível em: <<https://doi.org/10.1080/10407790802628838>>.
- [24] CORTELLESSA, G., ARPINO, F., DI FRAIA, S., et al. “Two-phase explicit CBS procedure for compressible viscous flow transport in porous materials”, *International Journal of Numerical Methods for Heat & Fluid Flow*, v. 28, n. 2, pp. 336–360, Jan 2018. ISSN: 0961-5539. doi: 10.1108/HFF-02-2017-0080. Disponível em: <<https://doi.org/10.1108/HFF-02-2017-0080>>.
- [25] SCHREFLER, B. A. “A CBS-type stabilizing algorithm for the consolidation of saturated porous media”, *International Journal for Numerical Methods in Engineering*, v. 122, n. 15, pp. 4027–4028, 2021. doi: <https://doi.org/10.1002/nme.6708>. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.6708>>.
- [26] NITHIARASU, P. “An arbitrary Lagrangian Eulerian (ALE) formulation for free surface flows using the characteristic-based split (CBS) scheme”, *International Journal for Numerical Methods in Fluids*, v. 48, pp. 1415–1428, 2005.
- [27] NOBARI, M., NADERAN, H. “A numerical study of flow past a cylinder with cross flow and inline oscillation”, *Computers & Fluids*, v. 35, n. 4, pp. 393–415, 2006. ISSN: 0045-7930. doi: <https://doi.org/10.1016/j.compfluid.2005.02.004>. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0045793005000411>>.
- [28] NOBARI, M., GHAZANFARIAN, J. “A numerical investigation of fluid flow over a rotating cylinder with cross flow oscillation”, *Computers & Fluids - COMPUT FLUIDS*, v. 38, pp. 2026–2036, 12 2009. doi: 10.1016/j.compfluid.2009.06.008.
- [29] BAO, Y., ZHOU, D., TU, J. “Flow interference between a stationary cylinder and an elastically mounted cylinder arranged in proximity”, *Journal of*

Fluids and Structures, v. 27, n. 8, pp. 1425–1446, 2011. ISSN: 0889-9746. doi: <https://doi.org/10.1016/j.jfluidstructs.2011.08.008>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0889974611001290>.

[30] SUN, X., ZHANG, J.-Z., REN, X.-L. “Characteristic-Based Split (CBS) Finite Element Method for Incompressible Viscous Flow with Moving Boundaries”, *Engineering Applications of Computational Fluid Mechanics*, v. 6, n. 3, pp. 461–474, 2012. doi: 10.1080/19942060.2012.11015435. Disponível em: <https://doi.org/10.1080/19942060.2012.11015435>.

[31] BLOM, F. J. “Considerations on the spring analogy”, *International Journal for Numerical Methods in Fluids*, v. 32, n. 6, pp. 647–668, 2000. doi: [https://doi.org/10.1002/\(SICI\)1097-0363\(20000330\)32:6<647::AID-FLD979>3.0.CO;2-K](https://doi.org/10.1002/(SICI)1097-0363(20000330)32:6<647::AID-FLD979>3.0.CO;2-K). Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291097-0363%2820000330%2932%3A6%3C647%3A%3AAID-FLD979%3E3.0.CO%3B2-K>.

[32] WANG, T., ZHOU, G., JIANG, C., et al. “A coupled cell-based smoothed finite element method and discrete phase model for incompressible laminar flow with dilute solid particles”, *Engineering Analysis with Boundary Elements*, v. 143, pp. 190–206, 2022. ISSN: 0955-7997. doi: <https://doi.org/10.1016/j.enganabound.2022.05.014>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0955799722001667>.

[33] CODINA, R., ZIENKIEWICZ, O. C. “CBS versus GLS stabilization of the incompressible Navier–Stokes equations and the role of the time step as stabilization parameter”, *Communications in Numerical Methods in Engineering*, v. 18, n. 2, pp. 99–112, 2002. doi: <https://doi.org/10.1002/cnm.470>. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/cnm.470>.

[34] YE, X., QIN, G., ZHUO, Y., et al. “Analysis/application of stabilization by the over-integration technique in CBS-SEM for incompressible flow”, *Computers & Mathematics with Applications*, v. 117, pp. 1–13, 2022. ISSN: 0898-1221. doi: <https://doi.org/10.1016/j.camwa.2022.04.005>. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0898122122001547>.

[35] CELIK, B., EDIS, F. O. “Computational Investigation of Micro Backward-Facing Step Duct Flow in Slip Regime”, *Nanoscale and Microscale Thermophysical Engineering*, v. 11, n. 3-4, pp. 319–331, 2007. doi: 10.

1080/15567260701715438. Disponível em: <<https://doi.org/10.1080/15567260701715438>>.

- [36] ABADI, S. M. A. N. R., JAFARI, A. “INVESTIGATING THE NATURAL CONVECTION HEAT TRANSFER FROM TWO ELLIPTIC CYLINDERS IN A CLOSED CAVITY AT DIFFERENT CYLINDER SPACINGS”, *Heat Transfer Research*, v. 43, n. 3, pp. 259–284, 2012. ISSN: 1064-2285.
- [37] ZHANG, X. H., OUYANG, J., ZHANG, L. “The characteristic-based split (CBS) meshfree method for free surface flow problems in ALE formulation”, *International Journal for Numerical Methods in Fluids*, v. 65, n. 7, pp. 798–811, 2011. doi: <https://doi.org/10.1002/flid.2213>. Disponível em: <<https://onlinelibrary.wiley.com/doi/abs/10.1002/flid.2213>>.
- [38] ZIENKIEWICZ, O., TAYLOR, R. *Finite Element Method, Vol 1: The Basis*. 5th ed. Oxford, Butterworth-Heinemann, 2000.
- [39] GEUZAINÉ, C., REMACLE, J.-F. “Gmsh: A 3-D Finite Element Mesh Generator with Built-in Pre- and Post-Processing Facilities”, *International Journal for Numerical Methods in Engineering*, v. 79, pp. 1309 – 1331, 09 2009. doi: 10.1002/nme.2579.
- [40] GHIA, U., GHIA, K., SHIN, C. “High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method”, *Journal of Computational Physics*, v. 48, n. 3, pp. 387–411, 1982. ISSN: 0021-9991. doi: [https://doi.org/10.1016/0021-9991\(82\)90058-4](https://doi.org/10.1016/0021-9991(82)90058-4). Disponível em: <<https://www.sciencedirect.com/science/article/pii/0021999182900584>>.
- [41] DENHAM, M., PATRICK, M. “Laminar flow over a downstream-facing step in a two-dimensional flow channel”, *Trans. Inst. Chem. Engrs*, v. 52, n. 4, pp. 361–367, 1974.