SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito: 08.08.2005

Assinatura: *Ana Paula Camparo Jurema*

Simulação de escoamentos multifásicos em malhas não estruturadas

Fabício Simeoni de Sousa

Orientador: *Prof. Dr. Norberto Mangiavacchi*

Tese apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Doutor em Ciências – Ciências de Computação e Matemática Computacional.

USP – São Carlos
Agosto de 2005

Aluna: *Fabício Simeoni de Sousa*

A Comissão Julgadora:

Prof. Dr. Norberto Mangiavacchi

Prof. Dr. Antonio Castelo Filho

Prof. Dr. Alvaro Luiz Gayoso de Azeredo Coutinho

Prof. Dr. João Luiz Filgueiras de Azevedo

Prof. Dr. Luís Manoel de Mexia Heitor de Medeiros Portela

À *Tatiane*,
com muito amor.

Agradecimentos

É difícil colocar em poucas palavras o sentimento de gratidão que tenho por várias pessoas que me ajudaram, seja diretamente ou indiretamente, em mais esta etapa da minha vida. É tão difícil quanto, é não ser injusto com alguém por causa de pequenos esquecimentos ao se fazer estes agradecimentos.

Não há como deixar de agradecer primeiramente a Deus por tudo que tem acontecido comigo nestes últimos anos, especialmente agora. A realização deste doutorado é sem dúvida um presente divino, que se revelou em muitas oportunidades que tive, e ainda tenho pela frente.

À Tatiane, minha esposa querida, tenho eterna gratidão por sua compreensão, apoio e incentivo, principalmente nas horas mais difíceis. Sua enorme paciência e otimismo foram com certeza os pilares que me sustentaram neste doutorado. A ela dedico este trabalho, com muito amor.

Um agradecimento especial ao prof. Norberto, pela excelente orientação durante todo o período de mestrado e doutorado. São quase seis anos que se concretizaram em vários trabalhos bem sucedidos, e que resultaram nas muitas oportunidades que tenho agora. Mais do que apenas orientador, se tornou um amigo, pelo qual tenho muito respeito e admiração. Obrigado pela oportunidade de trabalharmos juntos, e espero que continuemos a produzir bons resultados sempre.

Agradeço à FAPESP por ter me financiado por tanto tempo, desde meados de minha graduação em matemática, até o final do doutorado, apoio sem o qual não seria possível realizar este sonho. Agradeço também à CAPES por ter financiado meu estágio de doutorado na Holanda durante 2004, também imprescindível para realização desta etapa.

Agradeço ao prof. Luis, que deu contribuições valiosíssimas para este trabalho e pelas horas agradáveis nos corredores do Kramers Lab., onde discutimos desde física de fluidos, até política mundial e brasileira. Agradeço também ao prof. Rob Mudde, pelo apoio e dicas valiosas nos seminários que tivemos sobre o projeto desenvolvido por lá. Muito obrigado às pessoas que sempre me ajudaram, seja em assuntos profissionais ou pessoais. Em especial aos meus colegas de sala Marcos, Michal e HP, e aos bons amigos Sumit e Vânia, que nos acolheram sempre com muita hospitalidade. Aos amigos Fabiana e Oscar (hoje professor aqui na USP), pelo apoio no início deste período, sempre difícil para mar-inheiros de primeira viagem. Muito obrigado aos colegas Marteen, Phillip, Michael, Jos, Wouter, Gianandrea, Roberto e outros que não me lembro agora, pelas horas agradáveis de discussões produtivas nos corredores do Kramers Lab. e nos *pubs* em Delft. A eles, um sonoro *dank u wel*.

Ao prof. Sean McKee, que me apoiou e foi de grande ajuda em Oxford, quando participei do *ICFD Conference on Numerical Methods for Fluid Dynamics*, concorrendo ao prêmio *Bill Morton*, *thank you very much indeed*.

Minha gratidão aos professores e pesquisadores do ICMC, especialmente aos membros do LCAD, professores Castelo, Poti, Gustavo, Murilo, Valdemir e João Batista pelo enorme incentivo e pelas dicas e conselhos que ajudaram em minhas escolhas. Também aos professores da UNESP de Presidente Prudente, que sempre me incentivaram a seguir o caminho que estou trilhando hoje, principalmente aos professores José Roberto, Suctônio, Manuel, Piteri e ao professor Messias Meneguette, pela confiança depositada em mim e pela orientação durante minha iniciação científica.

Aos meus familiares pelo apoio e suporte em todas as fases de minha vida. Mesmo estando longe, quero que saibam que estarão sempre no meu coração e terão sempre meu agradecimento mais sincero e especial, principalmente meus pais Adélia e Luis e minha irmã Vivi.

Aos ex-companheiros do LCAD, Luciane, Ricardo, Maria Luíza, Juliana, Marcelo, Fernando Pio, e aos ainda companheiros de laboratório, Helton, Alex, Cássio, João Paulo e Kemelli, pelas dicas e pelos momentos agradáveis dentro e fora do LCAD. Aos amigos Luis Fernando & Dani, Billy & Andréia, Igor & Cíntia, pelos ótimos momentos de descontração.

Enfim, agradeço a todas as pessoas que, por um lapso de memória, não entraram na lista acima, mas que de forma direta ou indireta, deram importantes contribuições para a realização deste projeto.

Resumo

Esta monografia apresenta o desenvolvimento e os resultados obtidos da implementação de um método numérico para simular escoamentos multifásicos em malhas dinâmicas não estruturadas. As equações de *Navier-Stokes* são desenvolvidas em uma formulação *Lagrangeana-Euleriana* arbitrária e são aproximadas utilizando-se o método de elementos finitos. Um método de projeção baseado em decomposição *LU* é utilizado para desacoplar aceleração e pressão. A interface que define a fronteira livre entre os fluidos imiscíveis é representada explicitamente por vértices e arestas da triangulação, e a tensão interfacial é calculada através de uma distribuição baseada na discretização do gradiente de uma função *Heaviside*. O movimento da malha é computado através da composição entre a velocidade do escoamento e uma velocidade elástica, calculada utilizando-se um filtro *Laplaciano* a partir da posição dos vértices. O controle da malha dinâmica é feito através de inserção e remoção de pontos baseando-se em triangulações localmente *Delaunay*, para se manter a qualidade dos elementos. Adicionalmente, é proposto um esquema de distribuição de pontos através da estimativa do erro baseado na *Hessiana* das velocidades. São apresentadas validações para escoamentos monofásicos e multifásicos, com comparações teóricas e corroboração por outros métodos, ilustrando o bom desempenho do método proposto. Adicionalmente, é mostrada uma aplicação a problemas de escoamentos de bolhas, comparando-se os resultados obtidos com resultados de outras técnicas numéricas.

Abstract

This thesis presents the results obtained from the implementation of a numerical method to simulate multiphase flows using dynamic unstructured meshes. The *Navier-Stokes* equations are obtained using the arbitrary *Lagrangian-Eulerian* (ALE) formulation, are discretized using the *Galerkin* finite element method. A projection method based on approximated block *LU* decomposition is employed to decouple acceleration and pressure. The interface between the immiscible fluids is represented by edges and vertices belonging to the triangulation, and the interface tension is computed using a distribution technique based on the gradient of a *Heaviside* function. The mesh is moved using a mesh velocity, computed from a combination between the fluid velocity and an elastic velocity. The elastic velocity is computed based on a *Laplacian* filter over the position of the mesh points. To control the quality of the elements in the triangulation, a dynamic mesh control procedure is employed, in which points are inserted and deleted based on *Delaunay* triangulations. Additionally, a smooth distribution of the edge sizes in the mesh is computed using the error estimated by the *Hessian* of the velocities. We present validations for one and two-fluid simulations, comparing the results to analytical expressions, experimental data and numerical results from other methods available in the literature with good agreement. Additionally, an application to the simulation of bubbly flows is carried out, and the comparisons to another numerical technique are presented.

Sumário

1	Introdução	1
1.1	Técnicas de discretização	2
1.2	Métodos para escoamentos de fluidos	3
1.3	Malhas dinâmicas	3
1.4	Escoamentos multifásicos	4
1.4.1	Representação da interface	4
1.4.2	Escoamentos com bolhas e aplicações	6
1.5	Proposta deste trabalho	7
1.6	Organização da tese	8
2	Equações de conservação para escoamentos multifásicos	9
2.1	Introdução	9
2.2	Preliminares	10
2.2.1	Formulação das equações	10
2.2.2	Varição temporal de propriedades em um referencial arbitrário	11
2.3	Equações de conservação	13
2.3.1	Equação de conservação de massa	13
2.3.2	Equação de conservação da quantidade de movimento linear	15
2.4	Condições inicial e de contorno	17
2.4.1	Condições para contornos rígidos	17
2.4.2	Condições na interface	18
2.5	Adimensionalização	20
2.5.1	Parâmetros adimensionais	20
2.5.2	Adimensionalização das equações de conservação	22
3	Método de elementos finitos	25
3.1	Formulação variacional	25

3.1.1	Formulação variacional da equação de <i>Poisson</i>	25
3.1.2	Formulação variacional das equações de <i>Navier-Stokes</i>	27
3.2	Discretização espacial	30
3.2.1	Conceitos básicos de elementos finitos	30
3.2.2	Algumas funções de interpolação para quadriláteros e triângulos . .	34
3.2.3	Tipos de elementos para escoamentos de fluidos	38
3.2.4	Método de <i>Galerkin</i> para equação de <i>Poisson</i>	41
3.2.5	Método de <i>Taylor-Galerkin</i> para as equações de <i>Navier-Stokes</i> . . .	44
3.3	Discretização temporal	50
4	Métodos de projeção	53
4.1	Introdução	53
4.2	A idéia geral do método da projeção	54
4.3	Método de projeção contínuo	56
4.4	Método de passo fracionário	57
4.5	Método baseado em decomposição <i>LU</i>	58
4.5.1	Matriz de massa “ <i>lumped</i> ”	60
5	Representação da interface	61
5.1	Métodos de representação da interface	61
5.1.1	Representação <i>Lagrangeana</i> da interface no método <i>ALE</i>	64
5.2	Cálculo da tensão interfacial	65
5.2.1	Cálculo da curvatura e distribuição da força	66
5.2.2	Discretização do termo de tensão interfacial	67
5.3	Captura de mudanças topológicas na interface	68
6	Controle de malha	69
6.1	Velocidade da malha dinâmica	69
6.1.1	Suavização da malha	70
6.1.2	Movimentação da interface	72
6.2	Controle da malha dinâmica	74
6.2.1	Triangulação de <i>Delaunay</i>	74
6.2.2	<i>Flipping</i> 2D	76
6.2.3	Inserção e remoção de pontos	77
6.3	Refinamento adaptativo	83
6.3.1	Refinamento para malhas dinâmicas	84

6.3.2	Estimativa do espaçamento da malha	84
7	Validação e resultados numéricos	87
7.1	Validações para escoamentos monofásicos confinados	87
7.1.1	Escoamento em uma cavidade	87
7.1.2	Escoamento de <i>Hagen-Poiseuille</i>	90
7.1.3	Escoamento em um degrau	92
7.1.4	Desprendimento de vórtices	93
7.2	Validações para escoamentos multifásicos	94
7.2.1	Bolha estática	96
7.2.2	Gota oscilante	97
7.2.3	Ascensão de bolhas	100
7.3	Escoamento com mudança topológica na interface	105
7.4	Inversão da força de sustentação em escoamentos de bolhas	105
7.4.1	Soluções obtidas com método <i>front-tracking/front-capturing</i> 3D . .	107
7.4.2	Soluções obtidas com método <i>ALE</i> 2D	112
8	Considerações finais	121
A	Teoremas importantes	125
B	Produtos de vetores e tensores	127
B.1	Produto diádico ($\otimes$)	127
B.2	Produto escalar entre tensores ($:$)	127
	Referências bibliográficas	128

Lista de Figuras

2.1	Domínios utilizados na formulação ALE.	11
3.1	Exemplo de uma triangulação de um domínio Ω	31
3.2	Elemento arbitrário com s lados.	32
3.3	Elemento quadrilátero de tamanho $a \times b$	34
3.4	Determinação de funções de interpolação para um elemento triangular. . .	36
5.1	Exemplo de representação da interface.	64
5.2	Determinação direta da força concentrada, aplicada pontualmente nos vértices de interface da malha.	65
5.3	Curvatura interpretada como variação dos vetores normal e tangente no comprimento de arco infinitesimal ds , e sua discretização.	66
5.4	Mudança topológica na interface: os elementos inválidos são identificados (esquerda) e suas fasces trocadas (direita).	68
6.1	Exemplo do reposicionamento de um vértice pela suavização <i>Laplaciana</i> : caso convexo e não convexo.	71
6.2	Exemplo da movimentação de pontos na superfície de uma bolha ascendendo em um fluido, degradando a sua representação discreta.	72
6.3	Movimentação de um ponto na interface em um incremento de tempo diferencial $dt \rightarrow 0$, do tempo t para o tempo $t + dt$, utilizando a velocidade do escoamento $\mathbf{u}$ e a velocidade $\mathbf{u}_I$ calculada por (6.5).	73
6.4	Movimentação de um ponto na interface em um incremento de tempo diferencial $dt \rightarrow 0$, do tempo t para o tempo $t + dt$, utilizando a velocidade do escoamento $\mathbf{u}$ e a velocidade $\mathbf{u}_I$ calculada por (6.6).	74
6.5	(a) Exemplo de uma triangulação que falha no critério de <i>Delaunay</i> ; (b) Exemplo de uma triangulação de <i>Delaunay</i>	75
6.6	Exemplo de triangulação de <i>Delaunay</i> e diagrama de <i>Voronoi</i>	76

6.7	Exemplos de arestas que possui (esquerda) e que não possui (direita) <i>flipping</i> .	76
6.8	Definição de aresta localmente <i>Delaunay</i> .	77
6.9	Etapas do processo de refinamento: (a) Aresta a ser subdividida; (b) Inserção de vértice infringiu a propriedade de <i>Delaunay</i> de um triângulo; (c) A aresta que não é localmente <i>Delaunay</i> é trocada por seu <i>flipping</i> ; (d) Exemplo de malha resultante, se todas as arestas forem subdivididas.	79
6.10	Triangulação inicial T_n e triangulação resultante da remoção de T_{old} de T_n .	81
6.11	Triangulações T_{old} a partir de A_{old} e T_{new} a partir de A_{new} .	81
6.12	Nova triangulação obtida da inserção de T_{new} no lugar de T_{old} .	81
6.13	Exemplo das soluções obtidas para a equação de <i>Helmholtz</i> unidimensional, para diferentes valores de k . A linha contínua representa o termo fonte ℓ_D , a linha tracejada foi obtida com $k = 0.0028$ e a linha pontilhada com $k = 0.0443$.	85
7.1	Domínio para o escoamento em uma cavidade, com valores adimensionais.	88
7.2	Malhas estáticas utilizadas na validação, com número de elementos igual a (a) 236, (b) 932 e (c) 3690.	88
7.3	Malhas adaptativas finais obtidas com refinamento segundo a <i>Hessiana</i> das velocidades, com número de elementos igual a (a) 463, (b) 893, (c) 1702 e (d) 3374.	89
7.4	Escala de cores dos campos de velocidade nas direções x e y . O refinamento foi realizado nas áreas onde há maior gradiente.	89
7.5	Malha adaptativa resultante e campo de velocidade na direção x .	91
7.6	Comparação entre os perfis das velocidades aproximada e exata.	92
7.7	Domínio para o escoamento em um degrau, com valores adimensionais.	92
7.8	Malha utilizada na simulação do degrau.	93
7.9	Linhas de corrente após a simulação atingir o estado estacionário, mostrando a circulação abaixo da camada limite.	93
7.10	Esquema do escoamento em torno de um cilindro, com valores adimensionais.	93
7.11	Malha utilizada na simulação do escoamento em torno de um cilindro.	94
7.12	Evolução temporal da vorticidade no escoamento.	95
7.13	Oscilação da componente v da velocidade, em $\mathbf{x} = (3.1, 5)$.	96
7.14	Domínio para a simulação de uma bolha estática.	96
7.15	Gráfico da pressão na bolha estática, para $h = 0.05$.	98
7.16	Perfis do salto de pressão na bolha estática, para várias malhas diferentes.	98
7.17	Inversão no sentido da velocidade u no escoamento, caracterizando oscilação.	99

7.18	Evolução do diâmetro da gota no tempo, ilustrando o amortecimento devido a viscosidade.	100
7.19	Domínio para a análise de convergência para o problema de ascensão de uma bolha.	101
7.20	Malhas utilizadas para o teste de convergência.	101
7.21	Malha inicial (a) e comparação entre as malhas obtidas no tempo $t = 12.5$ para os diversos parâmetros: (b) $\beta_2 = 0$, (c) $\beta_2 = 0.1$, (d) $\beta_2 = 0.5$, (e) $\beta_2 = 1.0$	103
7.22	Número de <i>Reynolds</i> da bolha e conservação de massa para os casos simulados.	104
7.23	Comparação entre a evolução no tempo dos diâmetros horizontal (D_x) e vertical (D_y).	104
7.24	Comparação entre os perfis das bolhas obtidas com $\beta_2 = 0$ (preta); $\beta_2 = 0.1$ (verde); $\beta_2 = 0.5$ (azul); $\beta_2 = 1.0$ (vermelha).	104
7.25	Coalescência de duas bolhas ascendendo em um fluido.	106
7.26	Esquema de representação do domínio referencial: W_b é a componente vertical da velocidade da bolha, W_0 é a velocidade de referência no contorno, L é o tamanho do domínio, D_b é o diâmetro da bolha, e $\omega = \partial v / \partial x = W_0 / L$ é a intensidade do escoamento cisalhante.	107
7.27	Número de <i>Reynolds</i> relativo das bolhas pequena e grande, respectivamente.	108
7.28	Corte bidimensional do domínio, mostrando as linhas de corrente para as bolhas pequena e grande.	109
7.29	Comparação entre os formatos das bolhas: (a) pequena, experimental; (b) pequena, numérica; (c) grande, experimental; (d) grande, numérica.	109
7.30	Trajetórias para as bolhas pequena e grande, com e sem escoamento cisalhante. As linhas representam as soluções numéricas.	110
7.31	Evolução temporal do coeficiente de força de sustentação transversal para as bolhas pequena e grande.	111
7.32	Estudo de resolução para as bolhas pequena e grande, onde D é o diâmetro da bolha e h é o espaçamento da malha. A razão D/h representa a quantidade de células presente no diâmetro equivalente da bolha na direção x	111
7.33	Evolução temporal da velocidade vertical, vista no referencial da bolha, para $D = 2.91 \text{ mm}$. Os tempos são adimensionalizados por ω	113
7.34	Evolução temporal da velocidade vertical, vista no referencial da bolha, para $D = 3.52 \text{ mm}$. Os tempos são adimensionalizados por ω	114

7.35	Evolução temporal da velocidade vertical, vista no referencial da bolha, para $D = 5.53 \text{ mm}$. Os tempos são adimensionalizados por ω	114
7.36	Trajetórias e número de <i>Reynolds</i> das bolhas.	115
7.37	Domínio utilizado para o estudo da inversão da força de sustentação. . . .	115
7.38	Trajetórias e número de <i>Reynolds</i> para as bolhas.	116
7.39	Evolução temporal da interface e velocidade vertical vista no referencial da bolha, para $Ca = 0.01$. Os tempos mostrados são adimensionais ($t^* = \omega t$). .	117
7.40	Evolução temporal da interface e velocidade vertical vista no referencial da bolha, para $Ca = 0.08$. Os tempos mostrados são adimensionais ($t^* = \omega t$). .	117
7.41	Comparação entre os perfis e posição das bolhas inicial (linha serrilhada), $Ca = 0.01$ (direita) e $Ca = 0.08$ (esquerda), no tempo $t^* = 3.0$	119
7.42	Configuração das malhas em tempos diferentes, para $Ca = 0.01$ (a)-(b) e para $Ca = 0.08$ (c)-(d).	119

Capítulo 1

Introdução

Com o recente crescimento do poder computacional disponível, novas técnicas de aproximação numérica vem se desenvolvendo para as diversas áreas de pesquisa existentes. Simulações de processos físicos podem ser feitas com o auxílio do computador, permitindo aos cientistas a investigação de fenômenos não compreendidos, seja pela falta de ferramentas para se estudar tais fenômenos, ou pela impossibilidade de se reproduzir o processo físico em condições controladas.

A área de dinâmica de fluidos, que estuda a física de processos que envolvam escoamentos de fluidos, vem se beneficiando do desenvolvimento de novas e variadas técnicas de aproximação numérica. A dinâmica de fluidos computacional busca o desenvolvimento destas técnicas numéricas, sendo uma área altamente interdisciplinar, envolvendo conceitos de computação científica, engenharia, física, química, entre outras.

A dinâmica de fluidos computacional é cada vez mais utilizada como ferramenta para se explicar, paralelamente aos estudos analíticos e experimentais, vários fenômenos relevantes, como por exemplo escoamentos em reatores, transporte de óleos pesados, escoamento de lubrificantes, escoamentos de fluidos biológicos e aplicações na área médica, entre muitos outros.

Devido a variedade de aplicações relacionadas com a dinâmica de fluidos computacional, muitas técnicas diferentes para se aproximar os fenômenos físicos foram desenvolvidas. Estas técnicas, na sua grande maioria, buscam a representação discreta destes processos, tanto na representação geométrica do problema, quanto na discretização dos modelos teóricos existentes. A seguir são discutidas algumas destas técnicas.

1.1 Técnicas de discretização

As técnicas de discretização, como visto acima, buscam a representação discreta do problema de forma a tornar possível sua simulação no computador. Para tanto, considera-se dois tipos de aproximação: a discretização geométrica e a discretização dos modelos matemáticos para o problema.

As técnicas de discretização geométrica consistem na subdivisão do domínio contínuo em elementos computacionais representáveis no computador. Várias técnicas foram desenvolvidas para estes fins, sendo três as principais classes de métodos: discretização por malhas estruturadas, por malhas não estruturadas, e sem a utilização de malhas. A primeira busca a subdivisão do domínio preservando-se alguma estrutura de indexação, facilitando o acesso aos elementos através de seus índices. Em contrapartida, malhas não estruturadas não preservam tal característica, sendo que o acesso aos elementos é feito através de relações de vizinhança. A vantagem desta técnica é a facilidade de representação de domínios complexos, normalmente difíceis de serem representados por malhas estruturadas, além de propiciar a implementação de técnicas de refinamento adaptativo e malhas dinâmicas com maior controle de qualidade. Adicionalmente, novas técnicas de geração de malhas simpliciais através de triangulações de *Delaunay* proporcionam a obtenção de malhas com critérios de qualidade garantidos teoricamente [Shewchuk, 1997, Edelsbrunner et al., 2000]. Há ainda vários métodos de aproximação que buscam a solução numérica das equações dos modelos sem a utilização de malhas, os chamados métodos “*meshless*” [Mangiavacchi et al., 2002].

As técnicas de discretização de modelos matemáticos podem ser classificadas em três classes: métodos de diferenças finitas, métodos de volumes finitos e métodos de elementos finitos. Métodos de diferenças finitas são baseados em aproximações por séries de *Taylor* dos operadores diferenciais presentes nos modelos matemáticos, truncados de acordo com a precisão desejada. São métodos desenvolvidos especialmente para utilização em malhas estruturadas, mas novas técnicas permitem a sua utilização também para malhas não estruturadas e para discretizações sem a utilização de malhas. Os métodos de volumes finitos facilitam a discretização em malhas não estruturadas, avaliando-se as equações diferenciais integradas em volumes de controle na malha computacional. Os métodos de elementos finitos também podem facilmente ser utilizados sobre uma discretização de malhas não estruturadas. Estes buscam a solução do problema variacional associado ao problema original, pesando-se as equações e integrando-as no domínio computacional. Adicionalmente, funções de interpolação são associadas aos elementos para representação das grandezas relevantes do problema no interior dos mesmos.

1.2 Métodos para escoamentos de fluidos

Em se tratando de escoamentos de fluidos, há dois tipos a se considerar: escoamentos onde a massa específica dos fluidos muda com relação ao tempo e espaço, chamados de escoamentos compressíveis, e escoamentos onde mudança na massa específica do fluido é insignificante e pode ser desprezada, os chamados escoamentos incompressíveis.

Os métodos discutidos acima podem ser aplicados para a discretização das equações que modelam escoamentos de fluidos, as equações de *Navier-Stokes*. Adicionalmente, em escoamentos incompressíveis, que são o objeto de estudo deste trabalho, devido ao forte acoplamento entre as grandezas velocidade e pressão, a resolução destas equações exige um tratamento adequado. Há duas classes de métodos para solução de tais equações: os métodos acoplados e os métodos segregados. Métodos acoplados buscam uma aproximação resolvendo sistemas de equações não lineares multidimensionais de maneira acoplada [Fortuna, 2000, Zienkiewicz and Taylor, 2000]. Apesar de ser a maneira mais imediata de se resolver o problema, é muito difícil e computacionalmente muito cara. Neste sentido, os métodos segregados buscam a aproximação do sistema não linear pela solução de vários sistemas lineares, resolvendo-se as grandezas como velocidade e pressão de maneira segregada e sequencial.

Dentre os métodos segregados, os que mais se destacaram foram os denominados métodos de projeção. Tal família foi primeiramente introduzida por Chorin [Chorin, 1968], seguido por muitos outros autores, como [Amsden and Harlow, 1970] com o método SMAC, [Patankar and Spalding, 1972] com o método SIMPLE e [Tomé and McKee, 1994] com método GENSMAC. Nestes métodos, o campo de velocidades é inicialmente aproximado por um campo intermediário e então obtém-se uma correção, que é utilizada para a determinação da velocidade final naquele passo no tempo. Os métodos de projeção podem ainda ser classificados em métodos *contínuos*, que fazem o desacoplamento antes de qualquer discretização, métodos de *passo fracionário* [Gresho, 1990, Gresho and Chan, 1990], que fazem o desacoplamento após a discretização temporal e antes da discretização espacial, e métodos baseados em decomposição *LU* [Perot, 1993, Ni et al., 2003, Lee et al., 2001], que fazem o desacoplamento após as discretizações espacial e temporal das equações.

1.3 Malhas dinâmicas

Atualmente vem se tornando possível, e computacionalmente interessante, calcular aproximações para problemas em escoamentos de fluidos utilizando-se malhas dinâmicas, ou seja, malhas que se movem no espaço, a cada passo no tempo. Em uma estratégia *La-*

grangeana, por exemplo [Hirt et al., 1970, Sousa and Mangiavacchi, 2004], os pontos utilizados na discretização são movidos com a própria velocidade do fluido, tornando-se uma representação mais realista, uma vez que erros provenientes da convecção são totalmente eliminados, e a malha se adapta à geometria produzida pelo escoamento.

A movimentação da malha computacional pode ser feita também de maneira arbitrária, dando origem aos métodos do tipo *Lagrangeano-Euleriano* arbitrário [Hirt et al., 1997, Uchiyama, 2001, Jaeger and Carin, 2002, Luo et al., 2004]. Neste tipo de método, a velocidade da malha computacional é calculada segundo algum critério de qualidade ou suavização de malha, tal que o reposicionamento dos pontos proporcione uma melhoria na qualidade da aproximação. Métodos deste tipo são extremamente bem sucedidos em simulações com contornos que se movem, tais como paredes móveis, superfícies livres e, eventualmente, problemas envolvendo uma interface entre diferentes fluidos.

Técnicas de refinamento adaptativo podem ainda ser incorporadas aos métodos de malha dinâmica, aproveitando-se a movimentação dos pontos para reposicioná-los segundo algum critério de qualidade, baseado puramente em restrições geométricas ou ainda nas grandezas físicas resolvidas pelo escoamento.

Malhas dinâmicas apresentam um grande potencial para resolver problemas geometricamente complexos, tais como os que ocorrem em escoamentos multifásicos e com superfícies livres.

1.4 Escoamentos multifásicos

Escoamentos multifásicos são aqueles que envolvem a interação entre vários fluidos ao mesmo tempo. Para o caso onde os fluidos não se misturam, chamados de imiscíveis, a região que separa os diversos fluidos é uma área crítica do escoamento, cuja representação deve ser feita de maneira precisa para não degradar os resultados.

1.4.1 Representação da interface

Existem várias formas de se representar a interface em escoamentos multifásicos. Unverdi & Tryggvason classificam os métodos de representação em dois grandes grupos: métodos de captura de interface (*interface capturing*) e métodos de acompanhamento de interface (*interface tracking*) [Unverdi and Tryggvason, 1992]. Segundo estes autores, o método de captura consiste em tratar a interface como sendo uma região de variação acentuada, não sendo necessária a implementação de elementos computacionais explícitos para marcar a interface. Uma vantagem deste tipo de método é que mudanças topológicas, como

rupturas ou junções, podem ser tratadas automaticamente. Uma das desvantagens deste método é que a localização da interface não é bem definida, diminuindo sua precisão. Alguns exemplos de métodos de captura de interface são os métodos *Volume-of-Fluid*, ou VOF [Hirt and Nichols, 1981] e *level-set* [Sussman et al., 1994].

No método VOF, primeiramente introduzido por [Hirt and Nichols, 1981], são utilizadas funções marcadoras para reconstrução da interface, que assumem valores entre 0 e 1, dependendo da quantidade de fluido em cada célula. A cada passo no tempo, a interface é reconstruída a partir destas funções marcadoras, gerando interfaces discretas no interior de cada célula. Esta interface é então transportada com a velocidade do fluido, e as funções marcadoras são atualizadas. Uma extensão deste método para escoamentos bifásicos pode ser vista em [Ginzburg and Wittum, 2001].

O método *level-set* foi primeiramente introduzido em [Sussman et al., 1994], e consiste em representar a interface pelo conjunto de nível zero de uma função de pseudo-concentração ϕ , a qual é transportada pelo domínio computacional juntamente com o fluido. Esta abordagem introduz fórmulas convenientes para a aproximação da curvatura e dos vetores normais à interface, necessários para o cálculo da tensão interfacial. Este fato, juntamente com a simplicidade de implementação do método, constituem atrativos que devem ser levados em consideração na hora de se escolher a forma de representação da interface. No entanto a difusão de ϕ durante a simulação pode causar perda de precisão, levando à necessidade de se implementar esquemas especiais de convexão e formas de reinicialização para manter as propriedades desejáveis de ϕ .

Vários autores buscam o aprimoramento do método *level-set*. Por exemplo, van der Pijl [van der Pijl et al., 2003] apresenta um método híbrido resultado da combinação do método VOF com o método *level-set*, para eliminar problemas na conservação de massa. Já em [Sousa and Mangiavacchi, 2004] é apresentado um método *level-set Lagrangeano*, onde não há necessidade da solução de uma equação de transporte para ϕ , também reduzindo erros na conservação de massa.

Os métodos de acompanhamento introduzem elementos computacionais para se representar a interface, sendo estes movidos com a velocidade dos fluidos. Dentre os métodos de acompanhamento de interface, podem-se destacar *volume-tracking* e *front-tracking*.

O método de *volume-tracking* utiliza partículas marcadoras para reconstrução da interface, combinando uma precisão razoável com facilidade de implementação. No popular método MAC [Harlow and Welch, 1965], partículas marcadoras são inseridas para identificar a região espacial ocupada por um único fluido com superfície livre. Uma extensão deste método pode ser encontrada por exemplo em [Amsden and Harlow, 1970]. Avanços

significativos destes métodos foram feitos ao longo dos anos [McKee et al., 2004], dando origem a vários outros métodos, como o método GENSMAC [Tomé and McKee, 1994] para a solução de escoamentos com superfícies livres, por exemplo.

Contudo, o método de *front-tracking*, implementado pela primeira vez por Glimm e seus colaboradores [Glimm et al., 1988] possui uma precisão maior a um custo computacional relativamente menor. Tal método consiste em representar a interface por um conjunto interligado de pontos, formando uma malha *Lagrangeana*, que é movida sobre o domínio computacional. Esta técnica permite uma representação mais precisa da interface, porém o tratamento de mudanças topológicas na interface é extremamente difícil, principalmente no caso tridimensional. Apesar desta dificuldade, diversos pesquisadores produziram trabalhos relevantes, como [Esmaceli and Tryggvason, 1998] e [Esmaceli and Tryggvason, 1999] que aprimoraram o método para escoamentos tridimensionais com várias bolhas.

Há também métodos híbridos, como o método de projeção de segmentos introduzido em [Tornberg, 2000]. Trata-se de um híbrido entre o método *level-set* e o método *front-tracking*, e consiste em dividir a interface em segmentos, e representá-los por funções diferentes nas direções x e y . Este método facilita a representação de trocas topológicas na interface, com precisão comparável ao método de *front-tracking*.

Mais recentemente, em um artigo devido a Sousa et. al [Sousa et al., 2004], foi desenvolvido um método de *front-tracking/front-capturing* onde a representação da interface é feita através de uma malha Lagrangeana 2D, tal como o método *front-tracking*, porém, adicionalmente, representa a interface na malha Euleriana 3D através da variação do gradiente de uma função *Heaviside* que, discretizada, determina um esquema de distribuição dos efeitos de tensão superficial sobre os pontos próximos à interface.

1.4.2 Escoamentos com bolhas e aplicações

Em escoamentos bifásicos com bolhas, a interação entre os dois fluidos ainda não é bem entendida, apesar da importância deste tipo de escoamento para vários problemas industriais. Um problema comum na indústria petrolífera, por exemplo, é a extração de petróleo de grandes profundidades. Para diminuir o peso da coluna de petróleo, é utilizada uma técnica chamada de *gas-lift*, onde é injetado um gás na coluna de petróleo criando um escoamento com bolhas. No entanto, sob certas circunstâncias, as bolhas tendem a se concentrar no centro do escoamento, formando grandes golfadas de gás (*slugs*), o que prejudica o bombeamento. É conhecido que este tipo de fenômeno ocorre devido a uma inversão da força de sustentação (*lift*) agindo nas bolhas, que faz com que certas bolhas

se concentrem no centro do escoamento vertical. Apesar da relevância do problema, há poucos trabalhos na literatura que apresentem resultados de soluções numéricas para este tipo de escoamento.

Vários autores buscam o entendimento deste fenômeno. Um dos trabalhos mais significativos é o de Tomiyama [Tomiyama, 2002], onde foram conduzidos vários experimentos com bolhas ascendendo em um escoamento cisalhante vertical. Ele conclui que bolhas pequenas têm sua força de sustentação direcionada para a parede, enquanto que bolhas grandes tendem a se concentrar no centro do escoamento. Outros autores também reportaram este tipo de comportamento [Grossetete, 1995, Liu, 1993]. Tomiyama também derivou uma correlação para o coeficiente de força de sustentação transversal, variando com o número de *Reynolds* para bolhas pequenas (diâmetro $D \leq 4.4 \text{ mm}$), e com o número de *Eötvös* para bolhas grandes ($D > 4.4 \text{ mm}$).

Dentre os poucos trabalhos que trazem resultados numéricos sobre o assunto, o de Ervin & Tryggvason [Ervin and Tryggvason, 1997] implementa um método de *front-tracking* sobre uma malha estruturada uniforme para simular a ascensão de bolhas em um escoamento cisalhante induzido por paredes se movendo em direções opostas. Neste trabalho foram feitas simulações 2D e 3D, e os resultados comparados com dados experimentais obtidos em [Shridar and Katz, 1995].

1.5 Proposta deste trabalho

Este trabalho propõe o desenvolvimento de um método numérico para a solução das equações de conservação que modelam o escoamento de vários fluidos imiscíveis, sobre uma malha de elementos finitos que se move com velocidade arbitrária no domínio. A interface será discretizada pelos próprios elementos computacionais da malha, e a aplicação da tensão interfacial feita através da distribuição da força nos nós livres vizinhos à interface.

Para a solução das equações de *Navier-Stokes*, é proposta aqui uma extensão de um método de projeção baseado em decomposição *LU* para escoamentos multifásicos. Também são propostos esquemas para movimentação dos pontos utilizando técnicas de suavização de malha, e para o controle da malha dinâmica utilizando triangulações de *Delaunay*.

1.6 Organização da tese

Esta tese está organizada da seguinte forma:

- O capítulo 2 apresenta o desenvolvimento das equações de conservação para escoamentos de fluidos baseando-se em um referencial se movendo com uma velocidade arbitrária;
- O capítulo 3 descreve a aproximação das equações através de discretizações pelo método de elementos finitos;
- O capítulo 4 apresenta o método da projeção, partindo da decomposição de *Helmholtz-Hodge*, até o desenvolvimento de métodos baseados em decomposição *LU* aproximada das equações discretizadas;
- O capítulo 5 apresenta o método utilizado para a representação da interface entre fluidos em uma malha dinâmica, bem como o cálculo distribuído da tensão interfacial;
- O capítulo 6 apresenta os métodos utilizados para a movimentação da malha, controle de qualidade da triangulação e as técnicas de refinamento adaptativo empregadas neste trabalho;
- O capítulo 7 apresenta as validações para o método proposto, assim como os resultados de sua aplicação a problemas de escoamentos multifásicos de interesse;
- O capítulo 8 apresenta as conclusões e considerações finais.

Capítulo 2

Equações de conservação para escoamentos multifásicos

Este capítulo descreve as equações de conservação, que são modelos matemáticos de princípios físicos utilizados na simulação de escoamentos de fluidos. As equações são desenvolvidas na formulação *Lagrangeana-Euleriana arbitrária*. São apresentadas as condições de contorno e em seguida é feita uma discussão sobre o termo de tensão interfacial incorporado às equações. Posteriormente, estas equações são apresentadas na forma adimensional.

2.1 Introdução

As equações que modelam o escoamento de fluidos compressíveis e incompressíveis, turbulentos e laminares, representam as expressões matemáticas de princípios físicos bem familiares, como

- Conservação de massa;
- Conservação de quantidade de movimento (segunda lei de Newton);
- Conservação de energia (primeira lei da termodinâmica).

Um fluido aqui será considerado como um contínuo de matéria, onde as propriedades do escoamento são descritas em termos de grandezas macroscópicas, como pressão, velocidade e massa específica.

Na simulação de escoamentos multifásicos (multifluidos), cada fluido envolvido determina uma fase. Assim, para este tipo de escoamento, pode-se proceder utilizando as seguintes estratégias com relação às fases:

- Considerar o fluido em todas as fases como um contínuo, e considerar que as propriedades mudam de acordo com a posição de partículas marcadoras da interface, em cada ponto do escoamento como um todo;
- Considerar o escoamento em cada uma das fases como sendo um domínio separado, com propriedades e composição constantes, e determinar a condição de acoplamento entre os diversos domínios;
- Considerar uma abordagem híbrida, onde no interior de cada fase são usadas as equações para propriedades constantes, e o acoplamento entre os domínios é obtido utilizando-se uma formulação de propriedades variáveis apenas para a interface.

Preende-se neste trabalho seguir a primeira linha de abordagem, por ser a que se mostrou mais apropriada para os casos bidimensional e tridimensional já abordados em outros trabalhos (veja em [Santos et al., 2001] e [Sousa and Mangiavacchi, 2002]), e principalmente devido ao tipo de método escolhido para desenvolvimento deste trabalho.

Desta maneira, as equações de conservação de quantidade de movimento e conservação de massa serão desenvolvidas com propriedades como massa específica e viscosidade variáveis para o escoamento como um todo, mas constantes no interior de cada uma das fases separadamente.

2.2 Preliminares

2.2.1 Formulação das equações

Na grande maioria das referências encontradas na literatura, há dois tipos de abordagem para se especificar o movimento de um fluido em uma dada região do espaço: a descrição *Euleriana* e a descrição *Lagrangeana* (veja por exemplo em [Nachbin, 2001]).

Na formulação *Euleriana*, define-se uma região fixa no espaço, que não se deforma com relação ao tempo, onde o comportamento do fluido será estudado. Neste caso, há fluxo de massa pelas faces desse volume de controle, e as equações para o escoamento são determinadas a partir do balanço de fluxos neste volume de controle.

Na formulação *Lagrangeana*, define-se uma região material formada por um conjunto de partículas de fluido, denominada de volume de controle *Lagrangeano*, ou ainda região

Figura 2.1: Domínios utilizados na formulação ALE.

material. Conforme as partículas se movimentam no escoamento, esta região se deforma, e não há fluxo de massa através de suas fronteiras. Nesta formulação, as grandezas do escoamento são especificadas como função do tempo e da partícula de fluido.

Existe porém uma forma mais geral de se descrever o movimento de um fluido, chamada de descrição *Lagrangeana-Euleriana* arbitrária (*Arbitrary Lagrangian-Eulerian*, ou simplesmente *ALE*). Nesta formulação, as equações são desenvolvidas em um referencial se movendo no espaço com velocidade arbitrária. Quando este referencial se move com a mesma velocidade do fluido, obtém-se as equações na descrição *Lagrangeana*, e quando o referencial está parado, obtém-se a descrição *Euleriana*. Essa descrição é muito útil para o caso de métodos com malhas dinâmicas, onde as equações são escritas no referencial da malha que se move com velocidade arbitrária.

2.2.2 Variação temporal de propriedades em um referencial arbitrário

Antes de se determinar as equações de conservação utilizando-se a descrição *Lagrangeana-Euleriana* arbitrária, é necessário descrever a variação temporal das propriedades macroscópicas do fluido em um referencial se movendo com velocidade arbitrária. Analogamente ao que foi feito em [Hughes, 1981], considere os domínios material (S_ξ), referencial ou computacional (S_η) e espacial (S_x) mostrados na figura 2.1. Na descrição *Lagrangeana-Euleriana* arbitrária, os domínios S_ξ e S_η estão se movendo enquanto S_x está parado.

Considere que uma partícula ξ de fluido ocupando o domínio S_ξ se move para um ponto $\mathbf{x}$ em um dado instante t no domínio espacial. Segundo [Aris, 1962], o movimento de um fluido pode ser descrito por uma transformação. Seja ϕ a transformação $\mathbf{x} = \phi(\xi, t)$ que descreve esse movimento, então define-se *velocidade* da partícula ξ no domínio espacial como sendo

$$\mathbf{u} = \left. \frac{\partial \mathbf{x}}{\partial t} \right|_{\xi} = \frac{\partial}{\partial t} \phi(\xi, t) \quad (2.1)$$

Da mesma forma, seja $\hat{\phi}$ a transformação que descreve o movimento de um ponto η no domínio S_η para um ponto $\mathbf{x}$ no domínio S_x , então define-se velocidade do ponto η no domínio espacial como

$$\hat{\mathbf{u}} = \left. \frac{\partial \mathbf{x}}{\partial t} \right|_{\eta} = \frac{\partial}{\partial t} \hat{\phi}(\eta, t) \quad (2.2)$$

Se uma propriedade f é espessa como função de (ξ, t) ou $(\mathbf{x}, t)$, a expressão corresponde a uma descrição material (*Lagrangeana*) ou espacial (*Euleriana*) respectivamente. Em uma formulação *Lagrangeana-Euleriana* arbitrária, a expressão de f como função de (η, t) é chamada de descrição referencial. Considere as seguintes derivadas temporais

$$\left. \frac{\partial}{\partial t} f(\mathbf{x}, t) \right|_{\xi} = \left. \frac{\partial}{\partial t} f(\phi(\xi, t), t) \right|_{\xi} = \left. \frac{\partial f}{\partial t} \right|_{\mathbf{x}} + \frac{\partial}{\partial t} \phi(\xi, t) \cdot \nabla f = \left. \frac{\partial f}{\partial t} \right|_{\mathbf{x}} + (\mathbf{u} \cdot \nabla) f \quad (2.3)$$

$$\left. \frac{\partial}{\partial t} f(\mathbf{x}, t) \right|_{\eta} = \left. \frac{\partial}{\partial t} f(\hat{\phi}(\eta, t), t) \right|_{\eta} = \left. \frac{\partial f}{\partial t} \right|_{\mathbf{x}} + \frac{\partial}{\partial t} \hat{\phi}(\eta, t) \cdot \nabla f = \left. \frac{\partial f}{\partial t} \right|_{\mathbf{x}} + (\hat{\mathbf{u}} \cdot \nabla) f \quad (2.4)$$

O termo do lado esquerdo da equação (2.3) é denominado *derivada material* ou *substantiva* da propriedade f , e denota-se como

$$\frac{Df}{Dt} = \left. \frac{\partial f}{\partial t} \right|_{\xi} \quad (2.5)$$

Portanto, subtraindo (2.4) de (2.3), a derivada material ou substantiva da propriedade f em um referencial η que se move com velocidade $\hat{\mathbf{u}}$ é dada por

$$\frac{Df}{Dt} - \left. \frac{\partial f}{\partial t} \right|_{\xi} = \left. \frac{\partial f}{\partial t} \right|_{\eta} + ((\mathbf{u} - \hat{\mathbf{u}}) \cdot \nabla) f \quad (2.6)$$

Note que em uma descrição *Lagrangeana*, o domínio referencial se desloca com a mesma velocidade do domínio material, ou seja, $\hat{\mathbf{u}} = \mathbf{u}$, e portanto a variação temporal de uma propriedade f nos domínios material e referencial é a mesma. Em uma descrição *Euleriana*, o domínio referencial está parado em relação ao domínio espacial, ou seja,

$\hat{\mathbf{u}} = \mathbf{0}$, e portanto a expressão da variação temporal de f no domínio material é dada por

$$\left. \frac{Df}{Dt} \right|_{\xi} = \left. \frac{\partial f}{\partial t} \right|_{\eta} + (\mathbf{u} \cdot \nabla) f \quad (2.7)$$

que corresponde à derivada material ou substantiva de f utilizando-se uma descrição *Euleriana*.

2.3 Equações de conservação

Nesta seção as equações de conservação serão desenvolvidas utilizando-se uma descrição *Lagrangeana-Euleriana* arbitrária, a partir dos balanços das propriedades em regiões materiais.

2.3.1 Equação de conservação de massa

Seja $V(t) \subset \mathbb{R}^m$ uma região material no espaço num instante t , e seja $S(t) = \partial V(t)$ o bordo de $V(t)$. A massa total nesta região material é dada por

$$m(V(t), t) = \int_{V(t)} \rho(\mathbf{x}, t) dV, \quad (2.8)$$

onde $\rho(\mathbf{x}, t)$ é a massa por unidade de volume, também denominada massa específica. O princípio da conservação de massa diz que, na ausência de fontes ou sorvedouros, a variação temporal de massa nessa região material é nula, ou seja,

$$\frac{d}{dt} m(V(t), t) = \frac{d}{dt} \int_{V(t)} \rho(\mathbf{x}, t) dV = 0. \quad (2.9)$$

Considerando-se $\mathbf{u}$ a velocidade da região material $V(t)$, e utilizando-se o teorema de *Leibnitz* (A.7) em (2.9), tem-se

$$\int_V \left. \frac{\partial}{\partial t} \rho(\mathbf{x}, t) \right|_{\mathbf{x}} dV + \int_S (\mathbf{n} \cdot \mathbf{u}) \rho dS = 0, \quad (2.10)$$

e aplicando o teorema da divergência (A.6) vem

$$\int_V \left. \frac{\partial}{\partial t} \rho(\mathbf{x}, t) \right|_{\mathbf{x}} dV + \int_V \nabla \cdot (\rho \mathbf{u}) dV = 0, \quad (2.11)$$

e portanto

$$\int_V \left(\frac{\partial}{\partial t} \rho(\mathbf{x}, t) \Big|_{\mathbf{x}} + \nabla \cdot (\rho \mathbf{u}) \right) dV = 0. \quad (2.12)$$

Note que (2.12) é nula para uma região material V arbitrária, e assim

$$\frac{\partial}{\partial t} \rho(\mathbf{x}, t) \Big|_{\mathbf{x}} + \nabla \cdot (\rho \mathbf{u}) = 0. \quad (2.13)$$

Substituindo (2.4) em (2.13) vem

$$\frac{\partial \rho}{\partial t} \Big|_{\eta} - (\hat{\mathbf{u}} \cdot \nabla) \rho + \nabla \cdot (\rho \mathbf{u}) = 0, \quad (2.14)$$

e utilizando a identidade

$$\nabla \cdot (\rho \mathbf{u}) = \rho \nabla \cdot \mathbf{u} + (\mathbf{u} \cdot \nabla) \rho \quad (2.15)$$

obtem-se finalmente a equação de conservação de massa

$$\frac{\partial \rho}{\partial t} \Big|_{\eta} + ((\mathbf{u} - \hat{\mathbf{u}}) \cdot \nabla) \rho + \rho \nabla \cdot \mathbf{u} = 0, \quad (2.16)$$

ou de forma mais compacta, utilizando a notação dada em (2.6), tem-se

$$\frac{D\rho}{Dt} + \rho \nabla \cdot \mathbf{u} = 0. \quad (2.17)$$

Como já foi dito, a massa específica ρ e a viscosidade μ são variáveis no sentido que elas tomarão valores diferentes para cada fase, sendo constantes no interior das mesmas. Desta forma, sendo $\Omega = \cup_i \Omega_i$, onde Ω_i é a região ocupada pelo fluido i , tem-se

$$\rho(\mathbf{x}) = \rho_i, \text{ se } \mathbf{x} \in \Omega_i, \quad (2.18)$$

$$\mu(\mathbf{x}) = \mu_i, \text{ se } \mathbf{x} \in \Omega_i. \quad (2.19)$$

Então, o que ocorre é um “salto” dessas propriedades na interface. Desta forma, como a massa específica de uma partícula material não se altera no tempo, tem-se

$$\frac{D\rho}{Dt} = 0 \quad (2.20)$$

e substituindo em (2.17), vem

$$\rho \nabla \cdot \mathbf{u} = 0 \quad (2.21)$$

e como $\rho \neq 0$ para todas as fases, tem-se

$$\nabla \cdot \mathbf{u} = 0 \quad (2.22)$$

A equação (2.22) é conhecida como condição de incompressibilidade, e será mantida em todo o domínio. Os fluidos que obedecem esta condição são chamados de fluidos incompressíveis. Esta condição diz que o campo de velocidades resultante do escoamento dos fluidos é *solenoidal*.

2.3.2 Equação de conservação da quantidade de movimento linear

A equação de conservação da quantidade de movimento linear é obtida aplicando-se a segunda lei de Newton, a qual afirma que a taxa de variação temporal da quantidade de movimento de uma partícula é igual a resultante das forças que agem sobre essa partícula. Considere novamente uma região material arbitrária $V(t) \in \Omega$ movendo-se no espaço, limitada por uma superfície $S(t) = \partial V(t)$.

O balanço integral da quantidade de movimento pode ser expresso pela equação

$$\frac{d}{dt} \int_{V(t)} \rho \mathbf{u}(\mathbf{x}, t) dV = \int_S \boldsymbol{\sigma} \cdot \mathbf{n} dS + \int_{V(t)} \rho \mathbf{b}(\mathbf{x}, t) dV \quad (2.23)$$

Na equação acima, o lado esquerdo representa a taxa de variação de quantidade de movimento linear (produto da massa pela aceleração), a primeira integral do lado direito representa a resultante das tensões agindo sobre a superfície S , e a segunda integral do lado direito representa a resultante das forças de campo.

Aplicando o teorema de *Leibnitz* (A.7) em (2.23) vem

$$\int_V \left. \frac{\partial(\rho \mathbf{u})}{\partial t} \right|_{\mathbf{x}} dV + \int_S (\mathbf{n} \cdot \mathbf{u}) \rho \mathbf{u} dS = \int_S \boldsymbol{\sigma} \cdot \mathbf{n} dS + \int_V \rho \mathbf{b} dV \quad (2.24)$$

e utilizando o teorema da divergência de *Gauss* (A.6) para transformar as integrais de superfície em integrais de volume, tem-se

$$\int_V \left. \frac{\partial(\rho \mathbf{u})}{\partial t} \right|_{\mathbf{x}} dV + \int_V \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) dV = \int_V \nabla \cdot \boldsymbol{\sigma} dV + \int_V \rho \mathbf{b} dV, \quad (2.25)$$

onde $\otimes$ representa o produto diádico entre dois vetores (ver B.1). Assim

$$\int_V \left(\left. \frac{\partial(\rho \mathbf{u})}{\partial t} \right|_{\mathbf{x}} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) - \nabla \cdot \boldsymbol{\sigma} - \rho \mathbf{b} \right) dV = 0, \quad (2.26)$$

de onde vem

$$\left. \frac{\partial(\rho \mathbf{u})}{\partial t} \right|_{\mathbf{x}} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) - \nabla \cdot \boldsymbol{\sigma} - \rho \mathbf{b} = 0 . \quad (2.27)$$

Utilizando (2.4) para substituir a derivada temporal, e distribuindo o operador divergente do segundo termo, obtém-se

$$\left. \frac{\partial(\rho \mathbf{u})}{\partial t} \right|_{\boldsymbol{\eta}} - (\hat{\mathbf{u}} \cdot \nabla)(\rho \mathbf{u}) + (\mathbf{u} \cdot \nabla)(\rho \mathbf{u}) + \rho \mathbf{u}(\nabla \cdot \mathbf{u}) - \nabla \cdot \boldsymbol{\sigma} - \rho \mathbf{b} = 0 , \quad (2.28)$$

que pela equação (2.22) resulta

$$\left. \frac{\partial(\rho \mathbf{u})}{\partial t} \right|_{\boldsymbol{\eta}} + ((\mathbf{u} - \hat{\mathbf{u}}) \cdot \nabla) \rho \mathbf{u} - \nabla \cdot \boldsymbol{\sigma} - \rho \mathbf{b} = 0 , \quad (2.29)$$

e finalmente de (2.6) vem

$$\frac{D(\rho \mathbf{u})}{Dt} = \nabla \cdot \boldsymbol{\sigma} + \rho \mathbf{b} . \quad (2.30)$$

Na equação (2.30), $\boldsymbol{\sigma}$ representa o tensor de tensões totais do escoamento. Utilizando um modelo *Newtoniano* isotrópico, este tensor pode ser escrito como

$$\boldsymbol{\sigma} = -p \mathbf{I} + \boldsymbol{\tau} \quad (2.31)$$

onde $p = p(\mathbf{x}, t)$ representa o campo de pressão do escoamento, e $\boldsymbol{\tau}$ é o tensor de tensões viscosas dado por

$$\boldsymbol{\tau} = 2\mu \mathbf{D} + \left(\lambda - \frac{2}{3}\mu \right) (\nabla \cdot \mathbf{u}) \mathbf{I} , \quad (2.32)$$

sendo μ o coeficiente de viscosidade dinâmica e λ o coeficiente de viscosidade volumétrica do fluido. Devido a condição de incompressibilidade (2.22), pode-se escrever simplesmente

$$\boldsymbol{\tau} = 2\mu \mathbf{D} . \quad (2.33)$$

O tensor $\mathbf{D}$ que aparece nas equações (2.32) e (2.33), é denominado *tensor deformação*, e é dado por

$$\mathbf{D} = \frac{1}{2} \dot{\boldsymbol{\gamma}}(\mathbf{u}) = \frac{1}{2} (\nabla \mathbf{u} + \nabla \mathbf{u}^T) , \quad (2.34)$$

onde $\dot{\boldsymbol{\gamma}}(\mathbf{u})$ é denominado de *tensor taxa de deformação*. Deste modo, utilizando o tensor de tensões *Newtoniano* isotrópico, pode-se calcular o divergente do tensor de tensões

totais:

$$\nabla \cdot \boldsymbol{\sigma} = \nabla \cdot [-p\mathbf{I} + \mu(\nabla \mathbf{u} + \nabla \mathbf{u}^T)] = -\nabla p + \nabla \cdot [\mu(\nabla \mathbf{u} + \nabla \mathbf{u}^T)] . \quad (2.35)$$

O vetor $\mathbf{b}$ é uma força por unidade de massa que age sobre o fluido na região Ω . Para o tipo de escoamento que é objeto de interesse neste trabalho, somente o campo gravitacional é considerado, ou seja, $\mathbf{b} = \mathbf{g}$. Finalmente, a equação de conservação de quantidade de movimento pode ser expressa como

$$\frac{D(\rho \mathbf{u})}{Dt} = -\nabla p + \nabla \cdot [\mu(\nabla \mathbf{u} + \nabla \mathbf{u}^T)] + \rho \mathbf{g} . \quad (2.36)$$

As equações da continuidade (2.22) e da conservação de quantidade de movimento linear (2.36) são as equações de conservação que modelam escoamentos incompressíveis, e são também chamadas de equações de *Navier-Stokes*.

2.4 Condições inicial e de contorno

A escolha da condição inicial e das condições de contorno apropriadas é fundamental para a formulação de qualquer problema modelado por equações diferenciais, uma vez que o comportamento físico da solução depende de tais condições. A condição inicial apropriada para as equações (2.22) e (2.36) é que o campo de velocidades seja especificado em todo o domínio de modo a respeitar as condições de contorno e que seja solenoidal.

A seguir serão detalhadas as condições de contorno apropriadas para alguns tipos de fronteira mais utilizados.

2.4.1 Condições para contornos rígidos

Para os contornos rígidos, pode-se aplicar as condições:

- *Sem escorregamento (no-slip)*: Para escoamentos viscosos, nas paredes sólidas, define-se a componente normal (u_n) e as componentes tangenciais (u_t) da velocidade na parede como sendo nulas. Esta condição reflete o fato de o fluido imediatamente adjacente a parede estar em repouso em relação a mesma.
- *Condição de simetria ou com escorregamento (free-slip)*: É usada quando há fronteiras de simetria, ou quando os efeitos da condição *no-slip* não são desejáveis. Neste caso, define-se $u_n = 0$ para a componente normal da velocidade, e $\partial u_t / \partial n = 0$ para as componentes tangenciais, onde n é a direção normal à fronteira rígida.

- *Condição de entrada de fluido (inflow)*: É usada em fronteiras onde há entrada de fluido no sistema (fonte de massa). Para tal condição, define-se $u_n = u_{\text{inflow}}$ para a componente normal da velocidade, e normalmente $u_t = 0$ para as componentes tangenciais.
- *Condição de saída de fluido (outflow)*: É usada em fronteiras onde há saída de fluido do sistema (sorvedouro de massa). Existem várias formas de se prescrever condições de contorno para saída de fluido, porém, a especificação correta de tais condições ainda é um problema difícil. Uma condição muito utilizada para esta fronteira é especificar um escoamento paralelo, dada por

$$u_t = 0 \quad \text{e} \quad \sigma^{nn} = 0 \quad (2.37)$$

onde σ^{nn} representa a tensão normal à fronteira de saída de fluido, que é dada por

$$\sigma^{nn} = \mathbf{n} \cdot \left[-p\mathbf{I} + \mu (\nabla \mathbf{u} + \nabla \mathbf{u}^T) \right] \cdot \mathbf{n} = -p + 2\mu \frac{\partial u_n}{\partial n}. \quad (2.38)$$

Assim, em uma fronteira de saída onde as componentes tangenciais da velocidade são nulas, pela equação da continuidade (2.22) tem-se $\partial u_n / \partial n = 0$, e de (2.37) obtém-se $p = 0$ na fronteira de saída de fluido.

2.4.2 Condições na interface

Para o caso de escoamentos multifásicos, incluindo os efeitos de tensão interfacial, algumas condições de contorno na superfície que separa os fluidos envolvidos (interface) devem ser impostas [Batchelor, 1994, Dufour and Pelletier, 1998]. Sendo I a interface entre dois fluidos A e B , tais condições são:

- Continuidade das velocidades normais: $\mathbf{u}_A \cdot \mathbf{n}_A = \mathbf{u}_B \cdot \mathbf{n}_B = \mathbf{u}_I \cdot \mathbf{n}_I$
- Continuidade das velocidades tangenciais: $\mathbf{u}_A \cdot \mathbf{t}_I = \mathbf{u}_B \cdot \mathbf{t}_I$
- Equilíbrio das forças na interface: $(\sigma_B - \sigma_A) \cdot \mathbf{n}_I = \sigma \kappa \mathbf{n}_I$

Nas condições acima, $\mathbf{u}_A$, $\mathbf{u}_B$ e $\mathbf{u}_I$ representam as velocidades nos fluidos A e B , e na interface I . Analogamente, $\mathbf{n}$ e $\mathbf{t}$ representam os vetores normais e tangenciais respectivamente, sendo suas posições identificadas através de seus índices. Ainda, σ_A e σ_B são os tensores de tensão total relativos a cada fluido. No modelo adotado neste trabalho, o tensor de tensões totais é o mesmo para todos os fluidos.

Geralmente estas condições são impostas como condições de contorno na interface. Neste trabalho, tais condições serão impostas através da adição de um termo fonte na equação (2.36), ou seja,

$$\frac{D(\rho \mathbf{u})}{Dt} = -\nabla p + \nabla \cdot [\mu (\nabla \mathbf{u} + \nabla \mathbf{u}^T)] + \rho \mathbf{g} + \mathbf{f}. \quad (2.39)$$

A força $\mathbf{f}$ representa, no caso de escoamentos multifásicos, uma força campo que está ligada à tensão interfacial, e portanto, deve agir diretamente na interface entre dois fluidos.

Segundo o modelo proposto por [Esmarelli and Tryggvason, 1998], tal força é dada por

$$\mathbf{f} = \int_{\Gamma} \sigma \kappa \delta^m(\mathbf{x} - \mathbf{x}') \mathbf{n} \, dI, \quad (2.40)$$

onde $\sigma = \sigma(\mathbf{x}, t) \in \mathbb{R}$ é o coeficiente de tensão interfacial entre os fluidos separados pela interface, $\kappa = \kappa(\mathbf{x}, t) \in \mathbb{R}$ denota a curvatura local da superfície que define a interface e $\mathbf{n} = \mathbf{n}(\mathbf{x}, t) \in \mathbb{R}^m$ é o vetor normal à interface em $(\mathbf{x}, t)$. A função $\delta^m : \mathbb{R}^m \rightarrow \mathbb{R}$ é uma função salto m -dimensional construída pela multiplicação de funções δ unidimensionais, ou seja,

$$\delta^m(\mathbf{x} - \mathbf{x}') = \delta(x_1 - x'_1) \delta(x_2 - x'_2) \cdots \delta(x_m - x'_m), \quad (2.41)$$

onde as variáveis com apóstrofo são avaliadas na interface. Segundo [Tryggvason et al., 1998], a adição deste modelo de tensão interfacial como termo fonte na equação de conservação de quantidade de movimento linear garante a satisfação das condições de contorno descritas no começo desta seção.

Um modelo equivalente proposto por [Unverdi and Tryggvason, 1992] é dado por

$$\mathbf{f} = \sigma \kappa \delta_D \mathbf{n} \quad (2.42)$$

e foi implementado por vários autores, como por exemplo [Ginzburg and Wittum, 2001], utilizando um método VOF¹, por [Tornberg and Engquist, 2000], utilizando o método *segment-projection*, e por [Sousa and Mangiavacchi, 2004] utilizando um método *level-set Lagrangeano*. Aqui, δ_D é uma função delta de *Dirac*, e sua ação sobre qualquer função suave $\varphi : \mathbb{R}^m \rightarrow \mathbb{R}$ é dada por

$$\int_{\Omega} \delta_D \varphi \, d\Omega = \int_I \varphi \, dI, \quad (2.43)$$

onde a última integral denota uma integral de superfície sobre a interface I .

¹ *Volume Of Fluid*

Um terceiro modelo equivalente aos dois anteriores é dado por

$$\mathbf{f} = -\sigma\kappa\nabla H \quad (2.44)$$

onde $H : \mathbb{R}^m \rightarrow \{0, 1\}$ é uma função de *Heaviside*, a qual pode ser definida como sendo 1 onde existe um certo fluido e 0 caso contrário. Neste caso, a interface será identificada como a região onde o gradiente da função salto possui um valor não-nulo. Ainda, a direção e o sentido da força são dados pela direção e sentido (contrário) do vetor gradiente. Este modelo é muito utilizado em métodos do tipo *level-set* [Sussman et al., 1994], onde H é definida em termos da função de *level-set*, e foi também utilizado em [Sousa et al., 2004] com sucesso para um método do tipo *front-tracking/front-capturing*. Por este motivo, este será também o modelo adotado neste trabalho, e seu tratamento descrito no capítulo 5.

2.5 Adimensionalização

Os problemas em mecânica dos fluidos envolvem grandezas que os caracterizam, como velocidade, pressão, massa específica, etc. Tais grandezas na forma dimensional são diferenciadas por suas magnitudes, dadas através de um sistema métrico escolhido previamente.

O processo de adimensionalização consiste em normalizar as dimensões das grandezas envolvidas no escoamento e identificar os parâmetros que o governam. Como resultado deste processo, aparecem as *parâmetros adimensionais*, que são relações únicas sobre as forças que atuam no escoamento. Deste modo, problemas em escalas de dimensão diferentes, mas que tenham as mesmas parâmetros adimensionais, terão resultados equivalentes, e são denominados similares.

A seguir são apresentadas alguns parâmetros que normalmente aparecem como resultado do processo de adimensionalização de problemas de escoamentos de fluidos.

2.5.1 Parâmetros adimensionais

- **Número de Reynolds (Re):** É a razão entre as forças inerciais e as forças viscosas do escoamento, dada por

$$Re = \frac{\rho_0 LU}{\mu_0} = \frac{LU}{\nu_0}, \quad (2.45)$$

onde ρ_0 , L , U e μ_0 são os valores de referência para a massa específica, comprimento, velocidade e viscosidade, respectivamente. O valor ν_0 representa a viscosidade cinemática, e é a razão entre a viscosidade aparente ou dinâmica μ_0 e a massa específica

ρ_0 . Desta forma, se $Re < 1$, então as forças viscosas são predominantes no escoamento, enquanto que $Re > 1$ significa uma predominância das forças inerciais. Quando $Re \gg 1$ as forças viscosas são importantes apenas em regiões adjacentes às superfícies sólidas, denominadas de camada limite. O número de *Reynolds* indica também se o escoamento é laminar ($Re < Re_{crítico}$) ou turbulento (caso contrário).

- **Número de Froude (Fr):** É a razão entre as forças inerciais e as forças gravitacionais, isto é

$$Fr = \frac{U}{\sqrt{gL}}. \quad (2.46)$$

O número de Froude caracteriza escoamentos nos quais gravidade tem papel importante como é o caso de escoamentos com superfícies livres. Se $Fr < 1$, há predominância das forças gravitacionais sobre as inerciais no escoamento.

- **Número de Weber (We):** É a razão entre as forças inerciais e as forças de tensão superficial, dada por

$$We = \frac{\rho_0 LU^2}{\sigma_0} \quad (2.47)$$

onde σ_0 é o valor de referência para o coeficiente de tensão superficial. Quando $We \gg 1$, as forças de tensão superficial não são importantes no escoamento.

- **Número de Eötvös (EO):** É a razão entre as forças de empuxo e forças de tensão superficial, dada por

$$EO = \frac{\rho_0 g D^2}{\sigma_0}. \quad (2.48)$$

Este parâmetro adimensional aparece em escoamentos que envolvam bolhas ou gotas, onde é conhecido um diâmetro de referência (D). É igual ao número de Weber, considerando-se que a velocidade é adimensionalizada por $U = \sqrt{gD}$. Alguns autores costumam usar $\rho_0 = |\rho_f - \rho_b|$, onde ρ_f e ρ_b são as densidades dos fluidos externo e da bolha ou gota respectivamente.

- **Número de Galileo ou Archimedes (N):** É a razão entre as forças de empuxo e as forças viscosas, dada por

$$N = \frac{\rho_0^2 D^3 g}{\mu_0^2} \quad (2.49)$$

onde D é o diâmetro de referência, como acontece com o número de Eötvös. Este número nada mais é do que o número de *Reynolds* ao quadrado, utilizando-se $U = \sqrt{gD}$ para adimensionalizar a velocidade.

- **Número de Morton (M):** É a razão entre o número de *Eötvös* ao cubo e o número de *Galileo* ou *Archimedes* ao quadrado, e é dado por

$$M = \frac{(\rho_f - \rho_b)\mu_0^4 g}{\rho_f^2 \sigma_0^3} \quad (2.50)$$

onde ρ_f e ρ_b são as densidades dos fluidos externo e da bolha respectivamente.

2.5.2 Adimensionalização das equações de conservação

Para cada grandeza relevante no problema, o seu valor dimensional pode ser expresso como o produto de uma grandeza adimensional por um valor dimensional de referência. Considere para esta adimensionalização que sejam conhecidos os valores de referência para comprimentos (L), velocidades (U), massa específica (ρ_0), viscosidade (μ_0), gravidade (g_0) e tensão superficial (σ_0), de modo que

$$\begin{aligned} \rho &= \rho_0 \rho^* & \mu &= \mu_0 \mu^* & p &= \rho_0 U^2 p^* \\ \mathbf{x} &= L \mathbf{x}^* & \mathbf{u} &= U \mathbf{u}^* & t &= \frac{L}{U} t^* \\ \mathbf{g} &= g_0 \mathbf{g}^* & \kappa &= \frac{1}{L} \kappa^* & \sigma &= \sigma_0 \sigma^* \end{aligned}$$

e para os operadores diferenciais,

$$\frac{\partial}{\partial t} = \frac{U}{L} \frac{\partial}{\partial t^*} \quad \text{e} \quad \nabla = \frac{1}{L} \nabla^* .$$

Considere ainda que o termo de tensão interfacial dado em (2.44) é escrito como

$$\mathbf{f} = -\frac{\sigma_0}{L^2} \sigma^* \kappa^* \nabla^* H = \frac{\sigma_0}{L^2} \mathbf{f}^* . \quad (2.51)$$

A adimensionalização da equação da continuidade (2.22) é imediata:

$$\nabla \cdot \mathbf{u} = 0 \quad \Rightarrow \quad \frac{U}{L} \nabla^* \cdot \mathbf{u}^* = 0 \quad \Rightarrow \quad \nabla^* \cdot \mathbf{u}^* = 0 . \quad (2.52)$$

Considerando a equação de quantidade de movimento (2.39), expandindo-se o termo de derivada material e substituindo as variáveis e operadores dimensionais pelos equivalentes

adimensionais, obtém-se

$$\begin{aligned} \frac{\rho_0 U^2}{L} \frac{\partial(\rho^* \mathbf{u}^*)}{\partial t^*} + \frac{\rho_0 U^2}{L} ((\mathbf{u}^* - \hat{\mathbf{u}}^*) \cdot \nabla^*) \rho^* \mathbf{u}^* = -\frac{\rho_0 U^2}{L} \nabla^* p^* \\ + \frac{\mu_0 U}{L^2} \nabla^* \cdot [\mu^* (\nabla^* \mathbf{u}^* + \nabla^* \mathbf{u}^{*T})] + \rho_0 g_0 \rho^* \mathbf{g}^* + \frac{\sigma_0}{L^2} \mathbf{f}^* . \end{aligned} \quad (2.53)$$

Multiplicando (2.53) por $L/\rho_0 U^2$ vem

$$\begin{aligned} \frac{\partial(\rho^* \mathbf{u}^*)}{\partial t^*} + ((\mathbf{u}^* - \hat{\mathbf{u}}^*) \cdot \nabla^*) \rho^* \mathbf{u}^* = -\nabla^* p^* \\ + \frac{\mu_0}{\rho_0 L U} \nabla^* \cdot [\mu^* (\nabla^* \mathbf{u}^* + \nabla^* \mathbf{u}^{*T})] + \frac{g_0 L}{U^2} \rho^* \mathbf{g}^* + \frac{\sigma_0}{\rho_0 L U^2} \mathbf{f}^* . \end{aligned} \quad (2.54)$$

Utilizando as definições de número de *Reynolds* (Re), *Froude* (Fr) e *Weber* (We), e eliminando-se os asteriscos para simplificar a notação, obtém-se a equação

$$\frac{\partial(\rho \mathbf{u})}{\partial t} + ((\mathbf{u} - \hat{\mathbf{u}}) \cdot \nabla) \rho \mathbf{u} = -\nabla p + \frac{1}{Re} \nabla \cdot [\mu (\nabla \mathbf{u} + \nabla \mathbf{u}^T)] + \frac{1}{Fr^2} \rho \mathbf{g} + \frac{1}{We} \mathbf{f} , \quad (2.55)$$

que é a forma adimensional da equação de conservação de quantidade de movimento linear.

Para o caso específico de escoamentos multifásicos de ascensão de bolhas, uma situação onde não há *a priori* informação sobre a velocidade de referência, os comprimentos são normalmente adimensionalizados pelo diâmetro de referência D e as velocidades são adimensionalizadas por $\sqrt{gD}$. Para este caso, o número de *Reynolds* é uma resposta, e não um parâmetro do problema. Assim, seguindo um procedimento análogo ao que foi colocado acima, a forma adimensional da equação de conservação de quantidade de movimento linear fica

$$\frac{\partial(\rho \mathbf{u})}{\partial t} + ((\mathbf{u} - \hat{\mathbf{u}}) \cdot \nabla) \rho \mathbf{u} = -\nabla p + \frac{1}{N^2} \nabla \cdot [\mu (\nabla \mathbf{u} + \nabla \mathbf{u}^T)] + \rho \mathbf{g} + \frac{1}{Eo} \mathbf{f} , \quad (2.56)$$

ou seja, os únicos parâmetros adimensionais que definem o problema neste caso são os números de *Galileo* (N) e de *Eötvös* (Eo).

Capítulo 3

Método de elementos finitos

Este capítulo descreve a formulação variacional e discretização das equações de *Navier-Stokes* no contexto do método de elementos finitos. Primeiramente, as equações serão escritas na formulação variacional, em seguida é feita uma breve apresentação do método de elementos finitos, e as equações são discretizadas no espaço pelo método de *Galerkin*. Finalmente é apresentada a discretização temporal com aproximações explícita e semi-implícita, e sua discussão.

3.1 Formulação variacional

Nesta seção, as equações de *Navier-Stokes* serão escritas na formulação variacional. Tal processo é o primeiro passo na aproximação por elementos finitos, e consiste em projetar as equações em um espaço cuja base é conhecida e integrá-las sobre todo o domínio. O resultado final é chamado de forma variacional ou forma fraca do problema inicialmente proposto. A fim de ilustrar este processo, primeiramente será obtida a forma variacional para uma equação de *Poisson*, e em seguida será obtida a forma variacional das equações de *Navier-Stokes*.

3.1.1 Formulação variacional da equação de *Poisson*

Considere a equação de *Poisson*

$$-\nabla \cdot (\alpha \nabla u) = f , \tag{3.1}$$

válida num domínio Ω de dimensão m , sujeita às condições de contorno

$$u = g, \quad \text{em } \Gamma_1 \quad (3.2)$$

$$\alpha \nabla u \cdot \mathbf{n} + \sigma u = h, \quad \text{em } \Gamma_2 \quad (3.3)$$

onde $\Gamma = \partial\Omega$ é o bordo de Ω , com $\Gamma = \Gamma_1 \oplus \Gamma_2$, isto é, $\Gamma = \Gamma_1 \cup \Gamma_2$ e $\Gamma_1 \cap \Gamma_2 = \emptyset$, e $\mathbf{n}$ é o vetor normal a Γ_2 . Note que (3.2) representa a condição de *Dirichlet* em Γ_1 e (3.3) representa uma condição mista em Γ_2 . Considere o espaço de dimensão infinita

$$L^2(\Omega) = \left\{ v : \Omega \rightarrow \mathbb{R}, \int_{\Omega} v^2 d\Omega < \infty \right\} \quad (3.4)$$

e o espaço de *Sobolev*

$$H^1(\Omega) = \left\{ v \in L^2(\Omega) : \frac{\partial v}{\partial x_i} \in L^2(\Omega), i = 1, \dots, m \right\}. \quad (3.5)$$

Pode-se definir os subespaços

$$\mathbb{V}_{u_\Gamma} = \{ v \in H^1(\Omega) : v = u_\Gamma \text{ em } \Gamma_1 \} \quad (3.6)$$

e em particular, para $u_\Gamma = 0$,

$$\mathbb{V}_0 = \{ v \in H^1(\Omega) : v = 0 \text{ em } \Gamma_1 \}. \quad (3.7)$$

Para obter a formulação variacional do problema dado pelas equações (3.1)-(3.3), basta multiplicar (3.1) por uma função teste $w \in \mathbb{V}_0$ arbitrária e integrar o resultado no domínio. Assim, o problema agora consiste em obter uma solução $u \in \mathbb{V}_{u_\Gamma}$ de forma que

$$\int_{\Omega} [-\nabla \cdot (\alpha \nabla u) - f] w d\Omega = 0 \quad (3.8)$$

para todo $w \in \mathbb{V}_0$. Aplicando a primeira forma de *Green* (A.3) na integral acima, tem-se

$$\int_{\Omega} \alpha \nabla u \cdot \nabla w d\Omega - \int_{\Gamma} w(\alpha \nabla u) \cdot \mathbf{n} d\Gamma - \int_{\Omega} f w d\Omega = 0 \quad (3.9)$$

e como

$$\int_{\Gamma} w(\alpha \nabla u) \cdot \mathbf{n} d\Gamma = \int_{\Gamma_1} w(\alpha \nabla u) \cdot \mathbf{n} d\Gamma + \int_{\Gamma_2} w(\alpha \nabla u) \cdot \mathbf{n} d\Gamma, \quad (3.10)$$

utilizando a condição de contorno (3.3) tem-se

$$\int_{\Omega} \alpha \nabla u \cdot \nabla w \, d\Omega - \int_{\Gamma_1} w (\alpha \nabla u) \cdot \mathbf{n} \, d\Gamma + \int_{\Gamma_2} (\sigma u - h) w \, d\Gamma - \int_{\Omega} f w \, d\Omega = 0 \quad (3.11)$$

Como $w \in \mathbb{V}_0 \Rightarrow w = 0$ em Γ_1 , vem

$$\int_{\Omega} \alpha \nabla u \cdot \nabla w \, d\Omega + \int_{\Gamma_2} (\sigma u - h) w \, d\Gamma - \int_{\Omega} f w \, d\Omega = 0 . \quad (3.12)$$

Portanto, definindo as seguintes formas

$$k(\alpha, u, w) = \int_{\Omega} \alpha \nabla u \cdot \nabla w \, d\Omega \quad (3.13)$$

$$\tilde{k}(u, w) = \int_{\Gamma_2} (\sigma u - h) w \, d\Gamma \quad (3.14)$$

$$m(f, w) = \int_{\Omega} f w \, d\Omega , \quad (3.15)$$

a formulação variacional do problema de Poisson dado por (3.1)-(3.3) pode ser reescrito da seguinte maneira: Encontrar $u \in \mathbb{V}_{u\Gamma}$ tal que

$$k(\alpha, u, w) + \tilde{k}(u, w) = m(f, w), \quad \forall w \in \mathbb{V}_0 . \quad (3.16)$$

Esta formulação também é conhecida como forma fraca do problema de *Poisson*. Note que as funções peso w satisfazem a condição de *Dirichlet* homogênea em Γ_1 , onde é imposta a condição de *Dirichlet* do problema. No entanto, não é necessário impor nenhuma restrição adicional vinda da condição de *Neumann* às funções peso w , sendo que esta condição aparece naturalmente como resultado da integração por partes. Por este motivo, neste tipo de formulação, as condições de *Dirichlet* são chamadas de condições de contorno essenciais, e as condições de *Neumann* são chamadas de condições de contorno naturais.

3.1.2 Formulação variacional das equações de *Navier-Stokes*

Considere as equações de conservação para escoamentos multifásicos incompressíveis na formulação *ALE*, as equações de *Navier-Stokes*, dadas na forma adimensional por

$$\frac{\partial(\rho \mathbf{u})}{\partial t} + ((\mathbf{u} - \hat{\mathbf{u}}) \cdot \nabla) \rho \mathbf{u} + \nabla p - \frac{1}{Re} \nabla \cdot [\mu (\nabla \mathbf{u} + \nabla \mathbf{u}^T)] - \frac{1}{Fr^2} \rho \mathbf{g} - \frac{1}{We} \mathbf{f} = 0 \quad (3.17)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (3.18)$$

válidas em um domínio $\Omega \subset \mathbb{R}^m$ sujeita às condições de contorno

$$\mathbf{u} = \mathbf{u}_\Gamma, \quad \text{em } \Gamma_1 \quad (3.19)$$

$$u_t = 0 \quad \text{e} \quad \boldsymbol{\sigma}^n = 0, \quad \text{em } \Gamma_2. \quad (3.20)$$

Note que (3.19) representa uma condição de velocidade prescrita (paredes sólidas ou entrada de fluido), e (3.20) representa uma condição de tensão normal e velocidade tangencial à fronteira Γ_2 nulas, normalmente utilizada em fronteiras de saída de fluido, como visto na seção 2.4. Considere o subespaço

$$\mathbb{V} = H^1(\Omega)^m = \{ \mathbf{v} = (v_1, \dots, v_m) : v_i \in H^1(\Omega), \forall i = 1, \dots, m \} \quad (3.21)$$

onde $H^1(\Omega)$ é o espaço de *Sobolev* dado por (3.5). Note ainda que $\mathbb{V} = H^1(\Omega)^m$ é o produto cartesiano de m espaços $H^1(\Omega)$. Definindo-se

$$\mathbb{V}_{\mathbf{u}_\Gamma} = \{ \mathbf{v} \in \mathbb{V} : \mathbf{v} = \mathbf{u}_\Gamma \text{ em } \Gamma_1 \} \quad (3.22)$$

$$\mathbb{P}_{p_\Gamma} = \{ q \in L^2(\Omega) : q = p_\Gamma \text{ em } \Gamma_2 \} \quad (3.23)$$

a formulação variacional do problema acima consiste em encontrar soluções $\mathbf{u}(\mathbf{x}, t) \in \mathbb{V}_{\mathbf{u}_\Gamma}$ e $p(\mathbf{x}, t) \in \mathbb{P}_0$ tais que

$$\int_{\Omega} \left\{ \frac{\partial(\rho \mathbf{u})}{\partial t} + ((\mathbf{u} - \hat{\mathbf{u}}) \cdot \nabla) \rho \mathbf{u} + \nabla p - \frac{1}{Re} \nabla \cdot [\mu (\nabla \mathbf{u} + \nabla \mathbf{u}^T)] - \frac{1}{Fr^2} \rho \mathbf{g} - \frac{1}{We} \mathbf{f} \right\} \cdot \mathbf{w} \, d\Omega = 0 \quad (3.24)$$

para todo $\mathbf{w} \in \mathbb{V}_0$, e também

$$\int_{\Omega} (\nabla \cdot \mathbf{u}) q \, d\Omega = 0 \quad (3.25)$$

para todo $q \in \mathbb{P}_0$. Desenvolvendo (3.24) vem

$$\begin{aligned} & \int_{\Omega} \frac{\partial(\rho \mathbf{u})}{\partial t} \cdot \mathbf{w} \, d\Omega + \int_{\Omega} ((\mathbf{u} - \hat{\mathbf{u}}) \cdot \nabla) \rho \mathbf{u} \cdot \mathbf{w} \, d\Omega + \int_{\Omega} \nabla p \cdot \mathbf{w} \, d\Omega \\ & - \int_{\Omega} \frac{1}{Re} \nabla \cdot [\mu (\nabla \mathbf{u} + \nabla \mathbf{u}^T)] \cdot \mathbf{w} \, d\Omega - \int_{\Omega} \frac{1}{Fr^2} \rho \mathbf{g} \cdot \mathbf{w} \, d\Omega - \int_{\Omega} \frac{1}{We} \mathbf{f} \cdot \mathbf{w} \, d\Omega = 0. \end{aligned} \quad (3.26)$$

Aplicando o teorema de integração por partes de *Green* (A.5) na integral do termo viscoso, tem-se

$$\begin{aligned} \int_{\Omega} \nabla \cdot [\mu (\nabla \mathbf{u} + \nabla \mathbf{u}^T)] \cdot \mathbf{w} \, d\Omega = & - \int_{\Omega} \mu [(\nabla \mathbf{u} + \nabla \mathbf{u}^T) : \nabla \mathbf{w}^T] \, d\Omega \\ & + \int_{\Gamma} \mathbf{n} \cdot [\mu (\nabla \mathbf{u} + \nabla \mathbf{u}^T) \cdot \mathbf{w}] \, d\Gamma, \end{aligned} \quad (3.27)$$

onde o operador $(:)$ representa o produto escalar entre dois tensores (ver B.2). A integral no contorno Γ que aparece na equação acima pode ser separada em duas integrais, em Γ_1 e Γ_2 . A integral em Γ_1 é nula devido ao fato que $\mathbf{w} = \mathbf{0}$ em Γ_1 . A integral em Γ_2 também é nula, fato que decorre diretamente da condição de contorno (3.20), e portanto, a integral em Γ é nula. Aplicando-se novamente integração por partes (teorema A.2) no termo de gradiente de pressão, vem

$$\int_{\Omega} \nabla p \cdot \mathbf{w} \, d\Omega = - \int_{\Omega} p \nabla \cdot \mathbf{w} \, d\Omega + \int_{\Gamma} p \mathbf{w} \cdot \mathbf{n} \, d\Gamma \quad (3.28)$$

onde a integral de contorno que aparece acima é nula pois $\mathbf{w} = \mathbf{0}$ em Γ_1 e $p = 0$ em Γ_2 , como visto na seção 2.4. Assim, (3.26) pode ser escrita como

$$\begin{aligned} \int_{\Omega} \frac{\partial(\rho \mathbf{u})}{\partial t} \cdot \mathbf{w} \, d\Omega + \int_{\Omega} ((\mathbf{u} - \hat{\mathbf{u}}) \cdot \nabla) \rho \mathbf{u} \cdot \mathbf{w} \, d\Omega \\ - \int_{\Omega} p \nabla \cdot \mathbf{w} \, d\Omega + \frac{1}{Re} \int_{\Omega} \mu [(\nabla \mathbf{u} + \nabla \mathbf{u}^T) : \nabla \mathbf{w}^T] \, d\Omega \\ - \frac{1}{Fr^2} \int_{\Omega} \rho \mathbf{g} \cdot \mathbf{w} \, d\Omega - \frac{1}{We} \int_{\Omega} \mathbf{f} \cdot \mathbf{w} \, d\Omega = 0. \end{aligned} \quad (3.29)$$

Definindo-se as formas integrais

$$m(\mathbf{v}, \mathbf{w}) = \int_{\Omega} \mathbf{v} \cdot \mathbf{w} \, d\Omega \quad (3.30)$$

$$a(\mathbf{u}, \mathbf{v}, \mathbf{w}) = \int_{\Omega} (\mathbf{u} \cdot \nabla) \mathbf{v} \cdot \mathbf{w} \, d\Omega \quad (3.31)$$

$$k(\mu, \mathbf{v}, \mathbf{w}) = \int_{\Omega} \mu [(\nabla \mathbf{v} + \nabla \mathbf{v}^T) : \nabla \mathbf{w}^T] \, d\Omega \quad (3.32)$$

$$g(p, \mathbf{w}) = \int_{\Omega} \nabla p \cdot \mathbf{w} \, d\Omega \quad (3.33)$$

$$d(p, \mathbf{w}) = \int_{\Omega} (\nabla \cdot \mathbf{w}) p \, d\Omega \quad (3.34)$$

pode-se escrever o problema na forma fraca como segue: Encontrar soluções $\mathbf{u}(\mathbf{x}, t) \in \mathbb{V}_{\mathbf{u}}$, e $p(\mathbf{x}, t) \in \mathbb{P}$ tais que

$$\begin{aligned} m\left(\frac{\partial(\rho\mathbf{u})}{\partial t}, \mathbf{w}\right) + a(\mathbf{u} - \hat{\mathbf{u}}, \rho\mathbf{u}, \mathbf{w}) - g(p, \mathbf{w}) \\ + \frac{1}{Re} k(\mu, \mathbf{u}, \mathbf{w}) - \frac{1}{Fr^2} m(\rho\mathbf{g}, \mathbf{w}) - \frac{1}{We} m(\mathbf{f}, \mathbf{w}) = 0 \end{aligned} \quad (3.35)$$

$$d(q, \mathbf{u}) = 0, \quad (3.36)$$

para todo $\mathbf{w} \in \mathbb{V}_0$ e $q \in \mathbb{P}_0$.

3.2 Discretização espacial

Até agora as equações de *Poisson* e de *Navier-Stokes* foram escritas na formulação variacional, porém na forma contínua. O objetivo desta seção é utilizar o método de elementos finitos para discretizá-las no espaço, de forma a obter um sistema de equações algébricas no caso da equação de *Poisson*, ou diferenciais ordinárias no caso das equações de *Navier-Stokes*.

3.2.1 Conceitos básicos de elementos finitos

O primeiro passo na discretização de qualquer problema a ser resolvido numericamente é a discretização do domínio. No caso do método de elementos finitos, o domínio Ω é discretizado por um número finito de elementos Ω^e de forma que

$$\Omega = \bigcup_{e=1}^E \Omega^e, \quad (3.37)$$

com $\Omega^e \cap \Omega^f = \emptyset$, $\forall e, f = 1, \dots, E$, $e \neq f$. Devido à natureza do método de elementos finitos, vários tipos de elementos podem ser utilizados na mesma malha, o que é conhecido na literatura por malhas mistas. Elementos curvos também podem ser usados com o devido tratamento numérico, porém a utilização de elementos poligonais é mais comum, devido à facilidade de implementação. Neste trabalho, o domínio será discretizado por simplexos (triângulos em 2D), formando uma triangulação do domínio Ω (um exemplo de uma triangulação de um domínio Ω pode ser visto na figura 3.1). A utilização de simplexos é justificada aqui pois um dos principais objetivos é a implementação de um método de malha dinâmica. A qualidade de malhas simplexiais em 2D, como por exemplo malhas baseadas em triangulações de *Delaunay*, é garantida por várias propriedades geométricas

Figura 3.1: Exemplo de uma triangulação de um domínio Ω .

e relações vindas de uma teoria bem estabelecida.

Para ilustrar os conceitos básicos do método de elementos finitos, considere uma função $\phi : \mathbb{R}^m \rightarrow \mathbb{R}$, como sendo a solução exata do problema de campo

$$\mathcal{L}\phi = f, \quad (3.38)$$

válida num domínio Ω com condições de contorno estabelecidas, e onde $\mathcal{L}$ é um operador diferencial. O método de elementos finitos busca uma aproximação $\Phi(\mathbf{x})$ da solução exata $\phi(\mathbf{x})$ de maneira particionada em cada elemento Ω^e do domínio, ou seja, a aproximação de $\phi(\mathbf{x})$ restrita a cada elemento Ω^e é tal que $\phi^e(\mathbf{x}) = 0$ para qualquer ponto $\mathbf{x}$ fora de Ω^e . Assim, a aproximação de $\phi(\mathbf{x})$ pode ser escrita como

$$\Phi(\mathbf{x}) = \sum_e \phi^e(\mathbf{x}). \quad (3.39)$$

O valor de ϕ avaliado nos vértices da malha que discretiza Ω é chamado de valor nodal de ϕ , denotado por ϕ_i , onde i é o número do nó correspondente, isto é,

$$\phi_i = \phi(\mathbf{x}_i). \quad (3.40)$$

Define-se graus de liberdade de um vértice (ou nó) como sendo a quantidade de variáveis associadas com este mesmo nó, e graus de liberdade de um elemento como sendo a soma dos graus de liberdade dos nós que definem tal elemento. Considere um elemento poligonal com s lados, como mostrado na figura 3.2, e considere que, associado a cada vértice do elemento está o valor de ϕ avaliado naquele nó, digamos ϕ_i , $i = 1, \dots, s$. A cada elemento pode ser associada uma família de funções de interpolação, também chamadas

Figura 3.2: Elemento arbitrário com s lados.

de funções de forma, representada genericamente pela matriz

$$\mathbf{N}^e(\mathbf{x}) = \begin{bmatrix} N_{1,1}^e(\mathbf{x}) & \cdots & N_{1,s}^e(\mathbf{x}) \\ \vdots & \ddots & \vdots \\ N_{r,1}^e(\mathbf{x}) & \cdots & N_{r,s}^e(\mathbf{x}) \end{bmatrix} \quad (3.41)$$

onde r representa o grau de liberdade de cada nó, e s representa a quantidade de nós em cada elemento. Portanto, o número de graus de liberdade no elemento é dado por rs . Assumindo que cada nó tenha somente um grau de liberdade, tem-se

$$\mathbf{N}^e(\mathbf{x}) = [N_1^e(\mathbf{x}), \dots, N_s^e(\mathbf{x})] \quad (3.42)$$

Assim, sendo

$$\boldsymbol{\delta}^e = [\phi_1, \dots, \phi_s]^T, \quad (3.43)$$

o que é chamado de vetor deslocamento na literatura de elementos finitos, tem-se que a aproximação para ϕ em cada elemento é dada por

$$\phi^e(\mathbf{x}) = \mathbf{N}^e \boldsymbol{\delta}^e \quad (3.44)$$

e a solução aproximada de (3.38) pode ser escrita como

$$\Phi(\mathbf{x}) = \sum_{e=1}^{NE} \phi^e(\mathbf{x}) = \sum_{e=1}^{NE} \mathbf{N}^e \boldsymbol{\delta}^e, \quad (3.45)$$

onde NE é o número total de elementos.

Tendo sido decidido o tipo de discretização do domínio, o próximo passo é escolher como são as funções de interpolação. A escolha mais comum é optar por funções polinomi-

ais, pois além de serem mais fáceis de manipular, tanto algebricamente quanto computacionalmente, aproximações polinomiais possuem propriedades garantidas pelo teorema da aproximação de *Weierstrass* (teorema A.1), o qual especifica que qualquer função contínua pode ser aproximada por um polinômio, tão próximo quanto se deseje.

A escolha de tais aproximações polinomiais é feita seguindo-se algumas observações:

1. O número total de termos no polinômio deve ser igual ao número total de graus de liberdade do elemento. Isto garante a unicidade da aproximação;
2. A aproximação não deve ter uma direção preferencial, ou seja, deve ter invariância geométrica;
3. Para garantir convergência, as incógnitas devem ser contínuas, e a aproximação polinomial tal que permita com que elas assumam qualquer forma linear arbitrária.

Apesar disso, é difícil estabelecer regras gerais que possam ser aplicadas a todos os casos. Geralmente não é ideal acrescentar na aproximação polinomial termos de alta ordem ao custo da exclusão de outros de baixa ordem, o que pode causar oscilações na interpolação. Assim, o uso de polinômios completos é mais recomendável.

O problema de se determinar as funções de interpolação através do uso de aproximações polinomiais leva à inversão de matrizes mal-condicionadas. Por isso, a melhor forma de se obter tais funções é através de interpolação direta no elemento. Isto é feito escolhendo-se convenientemente um conjunto de polinômios de interpolação $N_i^e(\mathbf{x})$ de maneira que, se $\mathbf{x}_j$ representa as coordenadas do nó j , então

$$N_i^e(\mathbf{x}_j) = \delta_{ij}, \quad i, j = 1, \dots, s \quad (3.46)$$

onde δ_{ij} representa o delta de *Kroneck*, dado por

$$\delta_{ij} = \begin{cases} 1, & \text{se } i = j, \\ 0, & \text{se } i \neq j. \end{cases} \quad (3.47)$$

É claro que as funções de forma obtidas desta maneira devem satisfazer as condições 1, 2 e 3 vistas anteriormente. Em particular, se as funções de forma satisfazem a condição 3, então

$$\sum_{i \in e} N_i^e = 1, \quad \sum_{i \in e} N_i^e \mathbf{x}_i = \mathbf{x}. \quad (3.48)$$

Figura 3.3: Elemento quadrilátero de tamanho $a \times b$.

3.2.2 Algumas funções de interpolação para quadriláteros e triângulos

Nesta seção são ilustradas algumas funções de interpolação para elementos quadriláteros e triangulares em 2D, no intuito de dar uma breve introdução ao cálculo de funções de interpolação para alguns tipos de elementos. A extensão desta família de funções para dimensões maiores pode ser feita seguindo-se desenvolvimento análogo.

Funções de interpolação de *Lagrange* para elementos quadriláteros

Considere o elemento quadrilátero e formado pelos nós A , B , C e D , ilustrado na figura 3.3, e considere a seguinte transformação de coordenadas

$$\xi(x, y) = \frac{2}{a}(x - x_m), \quad \eta(x, y) = \frac{2}{b}(y - y_m), \quad (3.49)$$

onde (x_m, y_m) são as coordenadas do centro do elemento e . Daqui para frente, o sistema de coordenadas (x, y) será denotado por sistema de coordenadas globais, e o sistema (ξ, η) será denotado por sistema de coordenadas locais do elemento. Note que no elemento e , $\xi \in [-1, 1]$ e $\eta \in [-1, 1]$.

Considerando-se que cada nó do elemento e tenha somente um grau de liberdade, sua família de funções de forma é dada pela matriz

$$\mathbf{N}^e(\xi, \eta) = \begin{bmatrix} N_A^e(\xi, \eta) & N_B^e(\xi, \eta) & N_C^e(\xi, \eta) & N_D^e(\xi, \eta) \end{bmatrix}, \quad (3.50)$$

onde, de acordo com (3.46), tem-se

$$N_A^e(\xi_A, \eta_A) - N_B^e(\xi_B, \eta_B) = N_C^e(\xi_C, \eta_C) - N_D^e(\xi_D, \eta_D) = 1 \quad (3.51)$$

e ainda,

$$N_i^e(\xi_j, \eta_j) = 0, \quad \forall i, j \in e, \quad i \neq j \quad (3.52)$$

com $\xi_i = \xi(x_i, y_i)$ e $\eta_i = \eta(x_i, y_i)$.

Tais funções são facilmente identificadas como polinômios interpoladores de *Lagrange* nas coordenadas locais do elemento, ou seja,

$$N_A^e(\xi, \eta) = \frac{1}{4}(1 - \xi)(1 - \eta) \quad (3.53)$$

$$N_B^e(\xi, \eta) = \frac{1}{4}(1 + \xi)(1 - \eta) \quad (3.54)$$

$$N_C^e(\xi, \eta) = \frac{1}{4}(1 + \xi)(1 + \eta) \quad (3.55)$$

$$N_D^e(\xi, \eta) = \frac{1}{4}(1 - \xi)(1 + \eta) \quad (3.56)$$

Como o elemento e possui quatro nós com um grau de liberdade em cada nó, a aproximação da variável ϕ no elemento segue a seguinte forma bilinear

$$\phi^e(x, y) = \alpha_0 + \alpha_1 x + \alpha_2 y + \alpha_3 xy, \quad (3.57)$$

e segundo a equação (3.44), pode-se escrever

$$\begin{aligned} \phi^e(x, y) &= \mathbf{N}^e(\xi(x, y), \eta(x, y)) \boldsymbol{\delta}^e \\ &= \begin{bmatrix} N_A^e(\xi, \eta) & N_B^e(\xi, \eta) & N_C^e(\xi, \eta) & N_D^e(\xi, \eta) \end{bmatrix} \begin{bmatrix} \phi_A \\ \phi_B \\ \phi_C \\ \phi_D \end{bmatrix}. \end{aligned} \quad (3.58)$$

Elementos quadriláteros com funções de interpolação de ordem mais alta podem ser determinados, utilizando-se mais pontos na interpolação de *Lagrange*, ou utilizando polinômios do tipo *Serendipity* [Zienkiewicz and Taylor, 2000].

Funções de interpolação para elementos triangulares

Apesar de ser possível a determinação de funções de interpolação em coordenadas globais para o triângulo, o uso de coordenadas locais simplifica e padroniza os cálculos na hora de se resolver as integrais do modelo variacional. Considere então um sistema de coordenadas locais no triângulo (L_1, L_2, L_3) , que na figura 3.4 é dado por

$$L_1 = \frac{A_1}{A}, \quad L_2 = \frac{A_2}{A}, \quad L_3 = \frac{A_3}{A}, \quad (3.59)$$

Figura 3.4: Determinação de funções de interpolação para um elemento triangular.

onde $A = A_1 + A_2 + A_3$ é a área do triângulo e A_1 , A_2 e A_3 são as áreas mostradas na figura 3.4. Note que, por exemplo, qualquer ponto sobre a aresta entre os vértices 1 e 2 tem coordenada $L_3 = 0$, e se o ponto coincide com o vértice 3, então $L_3 = 1$, uma vez que $A_3 = A$. Desta forma, a posição de qualquer ponto dentro do triângulo pode ser representada pelas coordenadas (L_1, L_2, L_3) , também chamadas de coordenadas baricênticas. Note ainda que

$$L_1 + L_2 + L_3 = 1. \quad (3.60)$$

A relação entre o sistema de coordenadas global (x, y) e o sistema de coordenadas local (L_1, L_2, L_3) é dada pelas equações

$$x = L_1x_1 + L_2x_2 + L_3x_3, \quad (3.61)$$

$$y = L_1y_1 + L_2y_2 + L_3y_3, \quad (3.62)$$

que podem ser resolvidas para obter L_i em termos de x e y como

$$L_i(x, y) = \frac{a_i + b_ix + c_iy}{2A}, \quad (3.63)$$

onde a área A é dada por

$$A = \frac{1}{2} \begin{vmatrix} 1 & x_1 & y_1 \\ 1 & x_2 & y_2 \\ 1 & x_3 & y_3 \end{vmatrix}. \quad (3.64)$$

As constantes a_i , b_i e c_i são dadas em termos das coordenadas nodais

$$a_i = x_j y_k - x_k y_j, \quad (3.65)$$

$$b_i = y_j - y_k, \quad (3.66)$$

$$c_i = x_k - x_j, \quad (3.67)$$

onde j e k são permutações cíclicas de i . De (3.63) as seguintes relações para as derivadas podem ser obtidas

$$\frac{\partial L_i}{\partial x} = \frac{b_i}{2A}, \quad \frac{\partial L_i}{\partial y} = \frac{c_i}{2A}. \quad (3.68)$$

Finalmente, um resultado sobre integração em coordenadas de área, importante para a discretização por elementos triangulares, é dado por

$$\int_A L_1^m L_2^n L_3^p dx dy = \frac{2Am!n!p!}{(m+n+p+2)!}, \quad (3.69)$$

cujas provas podem ser encontradas em [Davies, 1980].

Considerando um elemento triangular com um grau de liberdade em cada nó, a aproximação da variável ϕ no elemento segue a seguinte forma

$$\phi^e(x, y) = \alpha_0 + \alpha_1 x + \alpha_2 y, \quad (3.70)$$

e utilizando as coordenadas baricêntricas do triângulo e como funções de interpolação, isto é,

$$\mathbf{N}^e = \begin{bmatrix} L_1 & L_2 & L_3 \end{bmatrix}, \quad (3.71)$$

pode-se escrever

$$\phi^e(x, y) = \mathbf{N}^e(x, y) \boldsymbol{\delta}^e = \begin{bmatrix} L_1(x, y) & L_2(x, y) & L_3(x, y) \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix}. \quad (3.72)$$

Funções de interpolação para elementos de mais alta ordem podem ser determinadas, mas são facilmente encontradas em qualquer livro de elementos finitos, como [Davies, 1980] ou para mais detalhes [Zienkiewicz and Taylor, 2000].

3.2.3 Tipos de elementos para escoamentos de fluidos

Na seção anterior, foram apresentados dois tipos de elementos para o caso onde há apenas um grau de liberdade para cada nó do elemento e apenas para uma variável. Para o caso das equações de *Navier-Stokes*, é necessário calcular funções de interpolação para mais de uma variável ao mesmo tempo, como velocidade e pressão. Devido ao acoplamento entre estas duas grandezas nas equações, a combinação entre ambas não pode ser arbitrária.

Existe uma vasta literatura sobre esse assunto, principalmente no que diz respeito aos testes que devem ser feitos para identificar elementos que são ou não válidos para o problema. Por exemplo, em [Cuvelier et al., 1986] pode ser encontrada a derivação de uma condição de admissibilidade, conhecida como condição de *Babuška-Brezzi*, e que pode também ser vista em detalhes em [Zienkiewicz and Taylor, 2000]. No entanto, esta condição é um tanto abstrata e difícil de se verificar na prática. Outros autores fornecem métodos mais simplificados de se verificar a condição de *Babuška-Brezzi* (veja por exemplo em [Fortin, 1981]).

Na literatura são encontrados casos de uso de elementos que não satisfazem a condição de *Babuška-Brezzi*, porém, tais elementos não podem ser utilizados com o método padrão de *Galerkin*. Alguns outros métodos, como por exemplo o método de penalidades, permite o uso de alguns elementos desse tipo.

Nesta seção serão apresentados brevemente alguns elementos triangulares que são admissíveis segundo a condição *Babuška-Brezzi* e de suas funções de interpolação, bem como uma breve discussão sobre a praticidade de sua implementação para o problema abordado neste trabalho. O objetivo aqui é apresentar alguns elementos disponíveis na literatura e justificar a escolha feita para este trabalho.

Com relação à velocidade, é necessário que as aproximações sobre os lados do elemento sejam contínuas, enquanto que para a pressão, estas podem ser contínuas ou descontínuas. A seguir são apresentados alguns elementos, separados em duas famílias: elementos do tipo *Taylor-Hood*, caracterizados por aproximação contínua para a pressão, e elementos do tipo *Crouzeix-Raviart*, caracterizados por uma aproximação descontínua para a pressão [Cuvelier et al., 1986, Fortin, 1981].

Família *Taylor-Hood*

Elemento $P_1^1 - P_1$: Também chamado de *mini-elemento*, consiste de um triângulo em 2D (tetraedro em 3D) juntamente com uma aproximação chamada de “linear estendida” para a velocidade e uma aproximação linear para a pressão. A aproximação linear estendida para a velocidade é definida como uma aproximação linear nos vértices mais uma

função “bolha”, definida como sendo 1 no centróide e 0 nas fronteiras do triângulo. Para a velocidade, as funções de interpolação podem ser escritas em termos das funções de interpolação lineares dadas na seção anterior, ou seja,

$$\begin{aligned} N_i &= L_i - 9L_1L_2L_3, \quad i = 1, 2, 3 \\ N_4 &= 27L_1L_2L_3 \end{aligned} \quad (3.73)$$

enquanto que para a pressão, são utilizadas as funções de interpolação lineares dadas na seção 3.2.2.

Elemento $P_2 - P_1$: Neste elemento é feita uma aproximação quadrática para a velocidade e linear contínua para a pressão. Para a aproximação quadrática, são introduzidos graus de liberdade nos centros das arestas do triângulo. As funções de interpolação para a velocidade neste elemento são dadas em termos das funções lineares por

$$\begin{aligned} N_i &= L_i(2L_i - 1), \quad i = 1, 2, 3 \\ N_4 &= 4L_1L_3 \\ N_5 &= 4L_1L_2 \\ N_6 &= 4L_2L_3 \end{aligned} \quad (3.74)$$

Família Crouzeix-Raviart

Elemento $P_1 - P_0$: Possui graus de liberdade somente nos centros das arestas do triângulo, o que pode acarretar em valores diferentes nos vértices para cada elemento,

o que caracteriza uma aproximação não-conforme. A pressão é descontínua e constante no elemento, e sua função de interpolação é dada por

$$N_e = \begin{cases} 1, & \text{no elemento } e, \\ 0, & \text{nos outros elementos.} \end{cases} \quad (3.75)$$

Para a velocidade é possível construir uma base de divergência nula, o que permite a construção de um método de divergência nula para a solução das equações de *Navier-Stokes*. Mas detalhes podem ser encontrados em [Cuvelier et al., 1986].

Elemento $P_2^+ - P_1$: Elemento com aproximação quadrática extendida para a velocidade e linear descontínua para a pressão. Também utiliza-se de uma função “bolha” adicional no centróide do elemento. É muito utilizado para problemas em escoamentos de fluidos devido a sua precisão. As funções de interpolação para a velocidade, escrita em termos das funções lineares, são dadas por

$$\begin{aligned} N_i &= L_i(2L_i - 1) + 3L_1L_2L_3, \quad i = 1, 2, 3 \\ N_4 &= 4L_1L_3 - 12L_1L_2L_3 \\ N_5 &= 4L_1L_2 - 12L_1L_2L_3 \\ N_6 &= 4L_2L_3 - 12L_1L_2L_3 \\ N_7 &= 27L_1L_2L_3 \end{aligned} \quad (3.76)$$

enquanto que para a pressão são utilizadas as funções de interpolação lineares, mas com valores nos vértices diferentes para cada elemento.

Escolha de um elemento para malhas dinâmicas

Como o objetivo deste trabalho é implementar um método para solução das equações de *Navier-Stokes* utilizando uma formulação *Lagrangiana-Euleriana* arbitrária, a discretização por elementos finitos será feita sobre uma malha que se move em relação ao espaço. A malha será movida através do transporte de seus vértices, e portanto, para tornar viável o processo de transporte e controle de malha, é desejável que o elemento tenha graus de liberdade nos vértices do triângulo. Uma rápida análise nos elementos apresentados acima descarta o elemento $P_1 - P_0$, uma vez que este possui graus de liberdade somente nas arestas, o que torna o processo de transporte e remalhamento muito difícil.

Para os elementos de segunda ordem, como o $P_2 - P_1$ e o $P_2^+ - P_1$, a inserção de graus de liberdade nas arestas faz com que o problema seja resolvido com mais precisão, mas com um custo computacional elevado. Como o transporte dos pontos da malha é feito somente nos vértices do triângulo (para não deformar o elemento), a utilização de um elemento linear é mais apropriada. Para casos onde é necessário um nível maior de precisão, é possível definir localmente um grau de refinamento maior, aumentando a precisão com um sacrifício moderado de tempo de computação.

Assim, a escolha do mini-elemento ($P_1^+ - P_1$) é mais apropriada para este problema, uma vez que este é linear para a velocidade e pressão, e possui graus de liberdade nos vértices mais um grau de liberdade no centróide do triângulo, reduzindo o número de incógnitas do problema.

3.2.4 Método de *Galerkin* para equação de *Poisson*

O objetivo desta seção é dar uma breve introdução ao método de *Galerkin*, utilizado na discretização por elementos finitos, para o problema de *Poisson* cuja formulação variacional foi derivada na seção 3.1.1.

Considere o problema de *Poisson* dado por (3.1)-(3.3) e a sua respectiva formulação variacional (3.13)-(3.16). Suponha ainda que Ω seja discretizado por uma malha de elementos Ω^e aos quais estão associadas famílias de funções de interpolação lineares.

Considere ainda que em cada elemento e da malha está definida uma família de funções

$$\mathbf{N}^e = \begin{bmatrix} N_1^e & \dots & N_s^e \end{bmatrix} \quad (3.77)$$

onde s é o número de nós que definem o elemento ($s = 3$ no caso de elemento triangular). Para representar a numeração dos nós localmente em um elemento e qualquer, serão

utilizadas as letras i, j , com $i, j = 1, \dots, s$, e para representar a numeração global dos nós serão utilizadas as letras m, n com $m, n = 1, \dots, M$, onde M é o número total de nós na malha. Deste modo, se uma função de interpolação é referenciada como N_m^e , isto significa que $N_m^e = N_i^e$, onde i é a numeração local do nó m no elemento e .

Define-se função de interpolação nodal no nó m como sendo a soma de todas as funções de interpolação definidas em m , isto é,

$$N_m = \sum_{e \in S(m)} N_m^e, \quad (3.78)$$

onde $S(m)$ é o conjunto de todos os elementos que contém o nó m . Assim, se u_n é o valor nodal da variável u avaliada no nó n , a aproximação da variável u pode ser expressa em função de uma combinação linear de funções de interpolação nodais, como segue

$$U(x, y) = \sum_n N_n(x, y) u_n. \quad (3.79)$$

Tal aproximação pode ser escrita em função da numeração local dos nós, isto é,

$$U(x, y) = \sum_e \sum_{j \in e} N_j^e(x, y) u_j \quad (3.80)$$

e em cada elemento e da malha, $U(x, y)$ restrito a um elemento é dada por

$$u^e(x, y) = \sum_{j \in e} N_j^e(x, y) u_j. \quad (3.81)$$

Segundo o método de *Galerkin*, a discretização da forma fraca do problema de Poisson é feita substituindo-se o espaço V_{u_F} por um espaço de dimensão finita $V_{u_F}^h$, gerado pelas funções de interpolação N_m . Na prática, tal procedimento se resume na substituição da função peso w pelas funções de interpolação nodais N_m , e a incógnita u pela aproximação dada em (3.79), nas formas integrais (3.13)-(3.15). Considere a formulação variacional (3.16) substituindo-se as formas integrais (3.13)-(3.15), ou seja,

$$\int_{\Omega} \alpha \nabla u \cdot \nabla w \, d\Omega + \int_{\Gamma_2} (\sigma u - h) w \, d\Gamma = \int_{\Omega} f w \, d\Omega. \quad (3.82)$$

Substituindo w por N_m e u por U , tem-se

$$\int_{\Omega} \alpha \nabla \left(\sum_n N_n u_n \right) \cdot \nabla N_m \, d\Omega + \int_{\Gamma_2} \left[\sigma \left(\sum_n N_n u_n \right) - h \right] N_m \, d\Gamma = \int_{\Omega} f N_m \, d\Omega, \quad (3.83)$$

que define uma equação para cada nó livre da malha, isto é $m = 1, \dots, M$. Como Ω é discretizado por uma malha de elementos Ω^e , e como a integral de um elemento e não tem contribuição nas integrais dos outros elementos da malha, pode-se escrever

$$\sum_e \int_{\Omega^e} \sum_n \alpha \nabla (N_n u_n) \cdot \nabla N_m \, d\Omega + \sum_e \int_{\Gamma_2^e} \sum_n \sigma N_n u_n N_m \, d\Gamma = \sum_e \int_{\Omega^e} f N_m \, d\Omega + \sum_e \int_{\Gamma_2^e} h N_m \, d\Gamma, \quad (3.84)$$

onde as integrais são agora avaliadas em cada elemento Ω^e . Note que N_m restrito ao elemento e é exatamente N_m^e , e que $\sum_n N_n u_n$ restrito ao elemento e é $\sum_{j \in e} N_j^e u_j$, resultando

$$\sum_e \int_{\Omega^e} \sum_{j \in e} \alpha \nabla N_j^e \cdot \nabla N_m^e u_j \, d\Omega + \sum_e \int_{\Gamma_2^e} \sum_{j \in e} \sigma N_j^e N_m^e u_j \, d\Gamma = \sum_e \int_{\Omega^e} f N_m^e \, d\Omega + \sum_e \int_{\Gamma_2^e} h N_m^e \, d\Gamma. \quad (3.85)$$

Note que em (3.85) o índice j agora varia localmente dentro do elemento ($j = 1, 2, 3$, no caso de um elemento triangular linear), e o índice m ainda varia globalmente ($m = 1, \dots, M$). Mantendo-se uma correspondência biunívoca entre a numeração local (N_i^e) e a numeração global (N_m^e), a equação (3.85) pode ainda ser reescrita como

$$\sum_e \int_{\Omega^e} \sum_{i,j \in e} \alpha \nabla N_i^e \cdot \nabla N_j^e u_j \, d\Omega + \sum_e \int_{\Gamma_2^e} \sum_{i,j \in e} \sigma N_i^e N_j^e u_j \, d\Gamma = \sum_e \int_{\Omega^e} \sum_{i \in e} N_i^e f \, d\Omega + \sum_e \int_{\Gamma_2^e} \sum_{i \in e} N_i^e h \, d\Gamma. \quad (3.86)$$

A equação (3.86) define um sistema linear

$$\mathbf{K} \mathbf{u} = \mathbf{f} \quad (3.87)$$

onde $\mathbf{u}$ tem como componentes os valores nodais u_n , a matriz $\mathbf{K}$ e o vetor $\mathbf{f}$ são dados por

$$\mathbf{K} = \mathcal{A}(\mathbf{k}^e + \bar{\mathbf{k}}^e); \quad \mathbf{f} = \mathcal{A}(\mathbf{f}^e + \bar{\mathbf{f}}^e) \quad (3.88)$$

e $\mathcal{A}$ é o operador de montagem que segue a regra estabelecida pelos índices dados em (3.85). As matrizes $\mathbf{k}^e$ e $\bar{\mathbf{k}}^e$, e os vetores $\mathbf{f}^e$ e $\bar{\mathbf{f}}^e$ são definidos por

$$k_{ij}^e = \int_{\Omega^e} \alpha \nabla N_i^e \cdot \nabla N_j^e d\Omega \quad (3.89)$$

$$\bar{k}_{ij}^e = \int_{\Gamma_2^e} \sigma N_i^e N_j^e d\Gamma \quad (3.90)$$

$$f_i^e = \int_{\Omega^e} f N_i^e d\Omega \quad (3.91)$$

$$f_i^e = \int_{\Gamma_2^e} h N_i^e d\Gamma. \quad (3.92)$$

Note que nas expressões acima, os índices i e j são referentes à numeração interna de cada elemento, com $i, j = 1, 2, 3$ no caso de elementos triangulares lineares. O operador de montagem $\mathcal{A}$ nada mais é do que uma regra para se montar o sistema linear resultante, colocando as várias submatrizes $\mathbf{k}^e$ e $\bar{\mathbf{k}}^e$ e subvetores $\mathbf{f}^e$ e $\bar{\mathbf{f}}^e$ nas suas devidas posições, de forma a preservar a numeração nodal inicial.

As integrais acima podem ser calculadas exatamente para o caso de elementos triangulares com funções de forma lineares (veja seção 3.2.2), utilizando os resultados mostrados nas equações (3.68) e (3.69). Por exemplo, fazendo-se $N_i^e = L_i$ em (3.89), vem

$$k_{ij}^e = \int_{\Omega^e} \alpha \left(\frac{\partial L_i}{\partial x} \frac{\partial L_j}{\partial x} + \frac{\partial L_i}{\partial y} \frac{\partial L_j}{\partial y} \right) d\Omega = \frac{\alpha}{4A^e} (b_i b_j + c_i c_j), \quad (3.93)$$

onde b_i , b_j , c_i e c_j são dados por (3.66) e (3.67), e A^e é a área do elemento Ω^e onde está sendo calculada a integral. Desta forma, resolvendo-se o sistema dado em (3.87), encontram-se aproximações para a solução da equação de *Poisson* (3.1).

3.2.5 Método de *Taylor-Galerkin* para as equações de *Navier-Stokes*

Considere as equações de *Navier-Stokes* na sua forma adimensional, dada em (2.52) e (2.55), e sua forma variacional dada em (3.35)-(3.36). Para facilitar os cálculos, considere estas equações desenvolvidas em um sistema cartesiano bidimensional. Sejam

$$\mathbf{u} = (u, v), \quad \hat{\mathbf{u}} = (\hat{u}, \hat{v}), \quad \mathbf{g} = (g_x, g_y), \quad \mathbf{f} = (f_x, f_y), \quad \mathbf{w} = (w_x, w_y), \quad (3.94)$$

a equação (3.35) fica

$$\begin{aligned}
 & \int_{\Omega} \left(\frac{\partial(\rho u)}{\partial t} w_x + \frac{\partial(\rho v)}{\partial t} w_y \right) d\Omega \\
 & + \int_{\Omega} \left\{ \left((u - \hat{u}) \frac{\partial(\rho u)}{\partial x} + (v - \hat{v}) \frac{\partial(\rho u)}{\partial y} \right) w_x + \left((u - \hat{u}) \frac{\partial(\rho v)}{\partial x} + (v - \hat{v}) \frac{\partial(\rho v)}{\partial y} \right) w_y \right\} d\Omega \\
 & - \int_{\Omega} \left(p \frac{\partial w_x}{\partial x} + p \frac{\partial w_y}{\partial y} \right) d\Omega + \frac{1}{Re} \int_{\Omega} \mu \left\{ \left(\frac{\partial u}{\partial x} \frac{\partial w_x}{\partial x} + \frac{\partial v}{\partial x} \frac{\partial w_y}{\partial x} + \frac{\partial u}{\partial y} \frac{\partial w_x}{\partial y} + \frac{\partial v}{\partial y} \frac{\partial w_y}{\partial y} \right) \right. \\
 & \quad \left. + \left(\frac{\partial u}{\partial x} \frac{\partial w_x}{\partial x} + \frac{\partial u}{\partial y} \frac{\partial w_x}{\partial y} + \frac{\partial v}{\partial x} \frac{\partial w_y}{\partial x} + \frac{\partial v}{\partial y} \frac{\partial w_y}{\partial y} \right) \right\} d\Omega - \frac{1}{Fr^2} \int_{\Omega} \rho (g_x w_x + g_y w_y) d\Omega \\
 & \quad - \frac{1}{We} \int_{\Omega} (f_x w_x + f_y w_y) d\Omega = 0 . \quad (3.95)
 \end{aligned}$$

Segundo a formulação variacional do problema, precisa-se determinar soluções $\mathbf{u} = (u, v) \in \mathbb{V}_{\mathbf{u}_T}$ e $p \in \mathbb{P}$ de modo que (3.95) seja verdadeira para todo $\mathbf{w} = (w_x, w_y) \in \mathbb{V}_0$. Mas note que, sendo satisfeitas as seguintes expressões

$$\begin{aligned}
 & \int_{\Omega} \frac{\partial(\rho u)}{\partial t} w_x d\Omega + \int_{\Omega} \left((u - \hat{u}) \frac{\partial(\rho u)}{\partial x} + (v - \hat{v}) \frac{\partial(\rho u)}{\partial y} \right) w_x d\Omega \\
 & - \int_{\Omega} p \frac{\partial w_x}{\partial x} d\Omega + \frac{1}{Re} \int_{\Omega} \mu \left(\frac{\partial u}{\partial x} \frac{\partial w_x}{\partial x} + \frac{\partial u}{\partial y} \frac{\partial w_x}{\partial y} + \frac{\partial v}{\partial x} \frac{\partial w_x}{\partial x} + \frac{\partial v}{\partial y} \frac{\partial w_x}{\partial y} \right) d\Omega \\
 & \quad - \frac{1}{Fr^2} \int_{\Omega} \rho g_x w_x d\Omega - \frac{1}{We} \int_{\Omega} f_x w_x d\Omega = 0 \quad (3.96)
 \end{aligned}$$

$$\begin{aligned}
 & \int_{\Omega} \frac{\partial(\rho v)}{\partial t} w_y d\Omega + \int_{\Omega} \left((u - \hat{u}) \frac{\partial(\rho v)}{\partial x} + (v - \hat{v}) \frac{\partial(\rho v)}{\partial y} \right) w_y d\Omega \\
 & - \int_{\Omega} p \frac{\partial w_y}{\partial y} d\Omega + \frac{1}{Re} \int_{\Omega} \mu \left(\frac{\partial v}{\partial x} \frac{\partial w_y}{\partial x} + \frac{\partial v}{\partial y} \frac{\partial w_y}{\partial y} + \frac{\partial u}{\partial y} \frac{\partial w_y}{\partial x} + \frac{\partial u}{\partial x} \frac{\partial w_y}{\partial y} \right) d\Omega \\
 & \quad - \frac{1}{Fr^2} \int_{\Omega} \rho g_y w_y d\Omega - \frac{1}{We} \int_{\Omega} f_y w_y d\Omega = 0 , \quad (3.97)
 \end{aligned}$$

para quaisquer $w_x \in \mathbb{V}_0$ e $w_y \in \mathbb{V}_0$ respectivamente, então (3.95) é satisfeita automaticamente. Deste modo, pode-se trabalhar com as equações na direção x (3.96) e na direção y (3.97) separadamente, sem que haja alguma perda de generalidade. Para a equação (3.96), tem-se

$$\int_{\Omega} \left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} \right) q d\Omega = 0 . \quad (3.98)$$

Considere NV o número de pontos de velocidade, NP o número de pontos de pressão

e NE o número de elementos na malha de elementos finitos que discretiza o domínio Ω . O método de *Taylor-Galerkin* consiste fazer as seguintes substituições em (3.95)

$$u(\mathbf{x}, t) \approx \sum_{n=1}^{NV} N_n(\mathbf{x}) u_n(t) \quad (3.99)$$

$$v(\mathbf{x}, t) \approx \sum_{n=1}^{NV} N_n(\mathbf{x}) v_n(t) \quad (3.100)$$

$$p(\mathbf{x}, t) \approx \sum_{r=1}^{NP} P_r(\mathbf{x}) p_r(t) , \quad (3.101)$$

que são aproximações semi-contínuas, isto é, contínuas no tempo (t) e discretas no espaço ($\mathbf{x}$). Aqui, $N_n(\mathbf{x})$ representam as funções de interpolação utilizadas para a velocidade e $P_n(\mathbf{x})$ as funções de interpolação para a pressão. Considere ainda

$$g_x(\mathbf{x}, t) \approx \sum_{n=1}^{NV} N_n(\mathbf{x}) g_{x,n}(t) \quad g_y(\mathbf{x}, t) \approx \sum_{n=1}^{NV} N_n(\mathbf{x}) g_{y,n}(t) \quad (3.102)$$

$$f_x(\mathbf{x}, t) \approx \sum_{n=1}^{NV} N_n(\mathbf{x}) f_{x,n}(t) \quad f_y(\mathbf{x}, t) \approx \sum_{n=1}^{NV} N_n(\mathbf{x}) f_{y,n}(t) \quad (3.103)$$

como sendo as aproximações para os termos de gravidade e de tensão interfacial,

$$\rho(\mathbf{x}, t) \approx \sum_{e=1}^{NE} \rho^e(t) \quad (3.104)$$

$$\mu(\mathbf{x}, t) \approx \sum_{e=1}^{NE} \mu^e(t) \quad (3.105)$$

como sendo as aproximações para as propriedades de cada fluido, e considere ainda a linearização do termo convectivo dada pela aproximação das velocidades convectivas em cada elemento

$$(u - \hat{u})(\mathbf{x}, t) \approx \sum_{e=1}^{NE} (u^e - \hat{u}^e)(t) \quad (3.106)$$

$$(v - \hat{v})(\mathbf{x}, t) \approx \sum_{e=1}^{NE} (v^e - \hat{v}^e)(t) . \quad (3.107)$$

A equação de conservação de quantidade de movimento é normalmente avaliada em todos os nós livres de velocidade, e portanto, as funções peso w_x e w_y são substituídas por

funções de interpolação $N_m = N_m(\mathbf{x})$, $m = 1, \dots, NV$. Aplicando este procedimento na equação (3.96), vem

$$\begin{aligned} & \sum_e \int_{\Omega^e} \sum_n \frac{\partial(\rho^e u_n)}{\partial t} N_m N_n \, d\Omega \\ & + \sum_e \int_{\Omega^e} \sum_n \rho^e N_m \left((u^e - \hat{u}^e) \frac{\partial N_n}{\partial x} + (v^e - \hat{v}^e) \frac{\partial N_n}{\partial y} \right) u_n \, d\Omega - \sum_e \int_{\Omega^e} \sum_r \frac{\partial N_m}{\partial x} P_r p_r \, d\Omega \\ & + \frac{1}{Re} \sum_e \int_{\Omega^e} \sum_n \mu^e \left(\frac{\partial N_m}{\partial x} u_n \frac{\partial N_n}{\partial x} + \frac{\partial N_m}{\partial y} \frac{\partial N_n}{\partial y} u_n + \frac{\partial N_m}{\partial x} \frac{\partial N_n}{\partial x} u_n + \frac{\partial N_m}{\partial y} \frac{\partial N_n}{\partial x} v_n \right) d\Omega \\ & - \frac{1}{Fr^2} \sum_e \int_{\Omega^e} \sum_n \rho^e N_m N_n g_{x,n} \, d\Omega - \frac{1}{We} \sum_e \int_{\Omega^e} \sum_n N_m N_n f_{x,n} \, d\Omega = 0 \quad (3.108) \end{aligned}$$

para $m = 1, \dots, NV$. Na equação acima, os limites dos somatórios foram omitidos para simplificar a notação. Note que ρ^e é constante no interior de cada elemento, e adicionalmente, não varia com relação ao tempo neste elemento. Restringindo-se as funções de interpolação nodais a cada elemento e , tem-se

$$\begin{aligned} & \sum_e \int_{\Omega^e} \sum_{i,j \in e} \rho^e \frac{\partial u_j}{\partial t} N_i^e N_j^e \, d\Omega \\ & + \sum_e \int_{\Omega^e} \sum_{i,j \in e} \rho^e N_i^e \left((u^e - \hat{u}^e) \frac{\partial N_j^e}{\partial x} + (v^e - \hat{v}^e) \frac{\partial N_j^e}{\partial y} \right) u_j \, d\Omega - \sum_e \int_{\Omega^e} \sum_{i,k \in e} \frac{\partial N_i^e}{\partial x} P_k^e p_k \, d\Omega \\ & + \frac{1}{Re} \sum_e \int_{\Omega^e} \sum_{i,j \in e} \mu^e \left(\frac{\partial N_i^e}{\partial x} \frac{\partial N_j^e}{\partial x} u_j + \frac{\partial N_i^e}{\partial y} \frac{\partial N_j^e}{\partial y} u_j + \frac{\partial N_i^e}{\partial x} \frac{\partial N_j^e}{\partial x} u_j + \frac{\partial N_i^e}{\partial y} \frac{\partial N_j^e}{\partial x} v_j \right) d\Omega \\ & - \frac{1}{Fr^2} \sum_e \int_{\Omega^e} \sum_{i,j \in e} \rho^e N_i^e N_j^e g_{x,j} \, d\Omega - \frac{1}{We} \sum_e \int_{\Omega^e} \sum_{i,j \in e} N_i^e N_j^e f_{x,j} \, d\Omega = 0. \quad (3.109) \end{aligned}$$

Um procedimento análogo pode ser realizado a partir da equação (3.97) para obter

$$\begin{aligned} & \sum_e \int_{\Omega^e} \sum_{i,j \in e} \rho^e \frac{\partial v_j}{\partial t} N_i^e N_j^e \, d\Omega \\ & + \sum_e \int_{\Omega^e} \sum_{i,j \in e} \rho^e N_i^e \left((u^e - \hat{u}^e) \frac{\partial N_j^e}{\partial x} + (v^e - \hat{v}^e) \frac{\partial N_j^e}{\partial y} \right) v_j \, d\Omega - \sum_e \int_{\Omega^e} \sum_{i,k \in e} \frac{\partial N_i^e}{\partial y} P_k^e p_k \, d\Omega \\ & + \frac{1}{Re} \sum_e \int_{\Omega^e} \sum_{i,j \in e} \mu^e \left(\frac{\partial N_i^e}{\partial x} \frac{\partial N_j^e}{\partial x} v_j + \frac{\partial N_i^e}{\partial y} \frac{\partial N_j^e}{\partial y} v_j + \frac{\partial N_i^e}{\partial x} \frac{\partial N_j^e}{\partial y} u_j + \frac{\partial N_i^e}{\partial y} \frac{\partial N_j^e}{\partial y} v_j \right) d\Omega \\ & - \frac{1}{Fr^2} \sum_e \int_{\Omega^e} \sum_{i,j \in e} \rho^e N_i^e N_j^e g_{y,j} \, d\Omega - \frac{1}{We} \sum_e \int_{\Omega^e} \sum_{i,j \in e} N_i^e N_j^e f_{y,j} \, d\Omega = 0. \quad (3.110) \end{aligned}$$

A equação da continuidade (3.98), é avaliada nos nós livres de pressão, e portanto, a função peso q é aproximada pelas funções de interpolação associadas à pressão $P_r(\mathbf{x})$, resultando

$$\sum_e \int_{\Omega^e} \sum_n \left(\frac{\partial N_n}{\partial x} u_n + \frac{\partial N_n}{\partial y} v_n \right) P_r d\Omega = 0. \quad (3.111)$$

para $r = 1, \dots, NP$. Restringindo as funções de interpolação a cada elemento e , vem

$$\sum_e \int_{\Omega^e} \sum_{j,k \in e} \left(\frac{\partial N_j^e}{\partial x} u_j + \frac{\partial N_j^e}{\partial y} v_j \right) P_k^e d\Omega = 0. \quad (3.112)$$

As equações (3.109), (3.110) e (3.112) podem ser representadas na forma de um sistema de equações diferenciais ordinárias

$$\begin{aligned} \mathbf{M}_{\rho,x} \dot{\mathbf{u}} + \mathbf{A}_x \mathbf{u} + \frac{1}{Re} \{ (2\mathbf{K}_{xx} + \mathbf{K}_{yy}) \mathbf{u} + \mathbf{K}_{xy} \mathbf{v} \} - \mathbf{G}_x \mathbf{p} - \frac{1}{Fr^2} \mathbf{M}_{\rho,x} \mathbf{g}_x - \frac{1}{We} \mathbf{M}_x \mathbf{f}_x &= \mathbf{0} \\ \mathbf{M}_{\rho,y} \dot{\mathbf{v}} + \mathbf{A}_y \mathbf{v} + \frac{1}{Re} \{ \mathbf{K}_{yx} \mathbf{u} + (\mathbf{K}_{xx} + 2\mathbf{K}_{yy}) \mathbf{v} \} - \mathbf{G}_y \mathbf{p} - \frac{1}{Fr^2} \mathbf{M}_{\rho,y} \mathbf{g}_y - \frac{1}{We} \mathbf{M}_y \mathbf{f}_y &= \mathbf{0} \\ \mathbf{D}_x \mathbf{u} + \mathbf{D}_y \mathbf{v} &= \mathbf{0} \end{aligned} \quad (3.113)$$

onde $\dot{\mathbf{u}} = [\frac{\partial u_1}{\partial t}, \dots, \frac{\partial u_{NU}}{\partial t}]^T$, $\dot{\mathbf{v}} = [\frac{\partial v_1}{\partial t}, \dots, \frac{\partial v_{NV}}{\partial t}]^T$, $\mathbf{u} = [u_1, \dots, u_{NU}]^T$, $\mathbf{v} = [v_1, \dots, v_{NV}]^T$, $\mathbf{p} = [p_1, \dots, p_{NP}]^T$, $\mathbf{g}_x = [g_1^x, \dots, g_{NU}^x]^T$, $\mathbf{g}_y = [g_1^y, \dots, g_{NV}^y]^T$, $\mathbf{f}_x = [f_1^x, \dots, f_{NU}^x]^T$ e $\mathbf{f}_y = [f_1^y, \dots, f_{NV}^y]^T$ são os vetores dos valores nodais para as variáveis de velocidade e pressão, e para as forças de gravidade e interfacial. As matrizes deste sistema de EDO's são dadas por

$$\begin{aligned} \mathbf{M}_{\rho,x} &= \mathcal{A}_x(\mathbf{m}_\rho^e), & \mathbf{M}_{\rho,y} &= \mathcal{A}_y(\mathbf{m}_\rho^e), & \mathbf{M}_x &= \mathcal{A}_x(\mathbf{m}^e), & \mathbf{M}_y &= \mathcal{A}_y(\mathbf{m}^e), \\ \mathbf{K}_{xx} &= \mathcal{A}_x(\mathbf{k}_{xx}^e), & \mathbf{K}_{xy} &= \mathcal{A}_x(\mathbf{k}_{xy}^e), & \mathbf{K}_{yx} &= \mathcal{A}_y(\mathbf{k}_{yx}^e), & \mathbf{K}_{yy} &= \mathcal{A}_y(\mathbf{k}_{yy}^e), \\ \mathbf{A}_x &= \mathcal{A}_x(\mathbf{a}^e), & \mathbf{A}_y &= \mathcal{A}_y(\mathbf{a}^e), & \mathbf{G}_x &= \mathcal{A}_x(\mathbf{g}_x^e), & \mathbf{G}_y &= \mathcal{A}_y(\mathbf{g}_y^e), \\ \mathbf{D}_x &= \mathcal{A}_x(\mathbf{d}_x^e), & \mathbf{D}_y &= \mathcal{A}_y(\mathbf{d}_y^e), \end{aligned}$$

tal que as submatrizes $\mathbf{m}_\rho^e$, $\mathbf{m}^e$, $\mathbf{a}^e$, $\mathbf{k}_{xx}^e$, $\mathbf{k}_{xy}^e$, $\mathbf{k}_{yx}^e$, $\mathbf{k}_{yy}^e$, $\mathbf{g}_x^e$, $\mathbf{g}_y^e$, $\mathbf{d}_x^e$ e $\mathbf{d}_y^e$, que são matrizes definidas localmente para cada elemento, são dadas por

$$m_{\rho,ij}^e = \int_{\Omega^e} \rho^e N_i^e N_j^e d\Omega \quad (3.114)$$

$$m_{ij}^e = \int_{\Omega^e} N_i^e N_j^e d\Omega \quad (3.115)$$

$$a_{ij}^e = \int_{\Omega^e} \rho^e N_i^e \left((u^e - \hat{u}^e) \frac{\partial N_j^e}{\partial x} + (v^e - \hat{v}^e) \frac{\partial N_j^e}{\partial y} \right) d\Omega \quad (3.116)$$

$$k_{xx,ij}^e = \int_{\Omega^e} \mu^e \left(\frac{\partial N_i^e}{\partial x} \frac{\partial N_j^e}{\partial x} \right) d\Omega \quad (3.117)$$

$$k_{xy,ij}^e = \int_{\Omega^e} \mu^e \left(\frac{\partial N_i^e}{\partial y} \frac{\partial N_j^e}{\partial x} \right) d\Omega \quad (3.118)$$

$$k_{yx,ij}^e = \int_{\Omega^e} \mu^e \left(\frac{\partial N_i^e}{\partial x} \frac{\partial N_j^e}{\partial y} \right) d\Omega \quad (3.119)$$

$$k_{yy,ij}^e = \int_{\Omega^e} \mu^e \left(\frac{\partial N_i^e}{\partial y} \frac{\partial N_j^e}{\partial y} \right) d\Omega \quad (3.120)$$

$$g_{x,ik}^e = \int_{\Omega^e} \frac{\partial N_i^e}{\partial x} P_k^e d\Omega \quad (3.121)$$

$$g_{y,ik}^e = \int_{\Omega^e} \frac{\partial N_i^e}{\partial y} P_k^e d\Omega \quad (3.122)$$

$$d_{x,kj}^e = \int_{\Omega^e} \frac{\partial N_j^e}{\partial x} P_k^e d\Omega \quad (3.123)$$

$$d_{y,kj}^e = \int_{\Omega^e} \frac{\partial N_j^e}{\partial y} P_k^e d\Omega \quad (3.124)$$

O operador $\mathcal{A}$ que aparece acima é um operador que monta as submatrizes de elemento nas matrizes do sistema de EDO's (3.113), respeitando a correspondência entre índices globais e locais dados nas equações (3.109) e (3.110).

As dimensões das matrizes que aparecem no sistema (3.113) são $NV \times NP$ para $\mathbf{G}_x$ e $\mathbf{G}_y$, $NP \times NV$ para $\mathbf{D}_x$ e $\mathbf{D}_y$ e $NV \times NV$ para todas as outras. A partir de (3.113) é possível escrever o sistema de EDO's de forma mais compacta, acoplando as velocidades nas direções x e y , o que resulta

$$\begin{aligned} \mathbf{M}_\rho \dot{\mathbf{u}} + \mathbf{A} \mathbf{u} + \frac{1}{Re} \mathbf{K} \mathbf{u} - \mathbf{G} \mathbf{p} - \frac{1}{Fr^3} \mathbf{M}_\rho \mathbf{g} - \frac{1}{We} \mathbf{M} \mathbf{f} &= \mathbf{0} \\ \mathbf{D} \mathbf{u} &= \mathbf{0}, \end{aligned} \quad (3.125)$$

onde agora as variáveis são definidas como $\dot{\mathbf{u}} = [\frac{\partial u_1}{\partial t}, \dots, \frac{\partial u_{NV}}{\partial t}, v_1, \dots, v_{NV}]^T$, $\mathbf{u} = [u_1, \dots, u_{NV}, v_1, \dots, v_{NV}]^T$, $\mathbf{g} = [g_1^x, \dots, g_{NV}^x, g_1^y, \dots, g_{NV}^y]^T$, $\mathbf{f} = [f_1^x, \dots, f_{NV}^x, f_1^y, \dots, f_{NV}^y]^T$ e $\mathbf{p} =$

$[p_1, \dots, p_{NP}]^T$, e as matrizes dadas por

$$\begin{aligned} \mathbf{M}_\rho &= \begin{bmatrix} \mathbf{M}_{\rho,x} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_{\rho,y} \end{bmatrix}_{2NV \times 2NV} & \mathbf{M} &= \begin{bmatrix} \mathbf{M}_x & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_y \end{bmatrix}_{2NV \times 2NV} \\ \mathbf{A} &= \begin{bmatrix} \mathbf{A}_x & \mathbf{0} \\ \mathbf{0} & \mathbf{A}_y \end{bmatrix}_{2NV \times 2NV} & \mathbf{K} &= \begin{bmatrix} \mathbf{K}_{xx} & \mathbf{K}_{xy} \\ \mathbf{K}_{yx} & \mathbf{K}_{yy} \end{bmatrix}_{2NV \times 2NV} \\ \mathbf{G} &= \begin{bmatrix} \mathbf{G}_x \\ \mathbf{G}_y \end{bmatrix}_{2NV \times NP} & \mathbf{D} &= \begin{bmatrix} \mathbf{D}_x & \mathbf{D}_y \end{bmatrix}_{NP \times 2NV}. \end{aligned}$$

3.3 Discretização temporal

Na seção anterior foi visto como discretizar no espaço as equações de *Navier-Stokes* utilizando o método *Taylor-Galerkin*. O resultado deste processo é um conjunto de equações diferenciais ordinárias de primeira ordem, cujas derivadas podem ser discretizadas no tempo. Existem vários métodos disponíveis em uma vasta literatura sobre o assunto. O objetivo desta seção é apenas de apresentar algumas formas de discretizar a derivada temporal que aparece no sistema de EDO's resultante da discretização espacial pelo método de elementos finitos. São apresentadas a seguir discretizações de primeira e segunda ordem no tempo, utilizando estratégias explícita e semi-implícita.

Considere a equação de conservação de momento discretizada (3.125) e sejam $\mathbf{u}^n = \mathbf{u}(t)$ e $\mathbf{u}^{n+1} = \mathbf{u}(t + \Delta t)$ os vetores formados pelos valores nodais do campo de velocidades $\mathbf{u}(\mathbf{x}, t)$ avaliados nos tempos t e $t + \Delta t$ respectivamente.

Como o objetivo desta seção é apenas o de apresentar as discretizações no tempo utilizadas neste trabalho, considere que o termo de gradiente de pressão seja conhecido em um certo tempo $t \leq \tau < t + \Delta t$. Um método para obter soluções para velocidade e pressão é discutido no próximo capítulo.

O primeiro esquema a ser considerado aqui consiste em aproximar a derivada temporal por diferença avançada de primeira ordem, e avaliar o resto da expressão no tempo t . Seja

$$\mathbf{b} = \mathbf{G}\mathbf{p}^* + \frac{1}{Fr^2}\mathbf{M}_\rho\mathbf{g} + \frac{1}{We}\mathbf{M}\mathbf{f}, \quad (3.126)$$

a discretização temporal de $\dot{\mathbf{u}}$ resulta em

$$\mathbf{M}_\rho \left(\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} \right) + \mathbf{A}\mathbf{u}^n + \frac{1}{Re}\mathbf{K}\mathbf{u}^n - \mathbf{b} = \mathbf{0} \quad (3.127)$$

ou ainda,

$$\mathbf{M}_\rho \mathbf{u}^{n+1} = -\Delta t \left(\mathbf{A} \mathbf{u}^n + \frac{1}{Re} \mathbf{K} \mathbf{u}^n - \mathbf{b} \right) + \mathbf{M}_\rho \mathbf{u}^n \quad (3.128)$$

Note que, neste caso, a solução de (3.128) pode ser obtida resolvendo-se o sistema linear algébrico representado por de $\mathbf{M}_\rho$, já que todo lado direito de (3.128) é conhecido. Para a solução de tal sistema, pode-se aplicar, por exemplo, o método dos gradientes conjugados, uma vez que $\mathbf{M}_\rho$ é simétrica definida positiva.

É fato conhecido que, apesar de fácil de se manipular, o tratamento explícito do termo viscoso traz severas restrições ao incremento de tempo Δt . Por este motivo, discretizações implícitas são mais desejáveis. Um esquema totalmente implícito pode ser derivado avaliando os demais termos no tempo $t + \Delta t$, assim

$$\mathbf{M}_\rho \left(\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} \right) + \mathbf{A} \mathbf{u}^{n+1} + \frac{1}{Re} \mathbf{K} \mathbf{u}^{n+1} - \mathbf{b} = \mathbf{0} \quad (3.129)$$

e portanto,

$$\left(\mathbf{M}_\rho + \Delta t \mathbf{A} + \frac{\Delta t}{Re} \mathbf{K} \right) \mathbf{u}^{n+1} = \Delta t \mathbf{b} + \mathbf{M}_\rho \mathbf{u}^n \quad (3.130)$$

Apesar de ser um esquema robusto, a solução do sistema acima é difícil de se conseguir, pois a matriz $\mathbf{A}$, que vem dos termos não lineares da equação de quantidade de movimento não é simétrica. Isso exclui a possibilidade de se utilizar algoritmos rápidos, como o método de gradientes conjugados, para solução de (3.130). Uma alternativa que pode ser empregada, e que pode ser classificada como estratégia semi-implícita, consiste em deixar o termo não linear no tempo t e avaliar os outros termos no tempo $t + \Delta t$. Tal procedimento resulta no seguinte sistema

$$\mathbf{M}_\rho \left(\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} \right) + \mathbf{A} \mathbf{u}^n + \frac{1}{Re} \mathbf{K} \mathbf{u}^{n+1} - \mathbf{b} = \mathbf{0} \quad (3.131)$$

de onde vem

$$\left(\mathbf{M}_\rho + \frac{\Delta t}{Re} \mathbf{K} \right) \mathbf{u}^{n+1} = -\Delta t (\mathbf{A} \mathbf{u}^n - \mathbf{b}) + \mathbf{M}_\rho \mathbf{u}^n \quad (3.132)$$

Desta forma, como $(\mathbf{M}_\rho + \frac{\Delta t}{Re} \mathbf{K})$ é uma matriz simétrica definida positiva (pois $\mathbf{M}_\rho$ e $\mathbf{K}$ o são), pode-se utilizar o método dos gradientes conjugados para a solução de (3.132). Apesar do tratamento explícito do termo convectivo, tal procedimento permite o uso de um Δt maior pois retira a contribuição do termo viscoso sobre esta restrição.

É possível ainda generalizar estas discretizações temporais, avaliando-se os termos em um tempo $t + \alpha \Delta t$, com $0 \leq \alpha \leq 1$. Considerando que $\mathbf{u}^{n+\alpha} = \alpha \mathbf{u}^{n+1} + (1 - \alpha) \mathbf{u}^n$ e

mantendo-se o termo convectivo no tempo t , pode-se escrever

$$\mathbf{M}_\rho \left(\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} \right) + \mathbf{A} \mathbf{u}^n + \frac{1}{Re} \mathbf{K} \mathbf{u}^{n+\alpha} - \mathbf{b} = \mathbf{0} \quad (3.133)$$

e substituindo $\mathbf{u}^{n+\alpha}$ vem

$$\mathbf{M}_\rho \left(\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} \right) + \mathbf{A} \mathbf{u}^n + \frac{1}{Re} \mathbf{K} (\alpha \mathbf{u}^{n+1} + (1 - \alpha) \mathbf{u}^n) - \mathbf{b} = \mathbf{0} \quad (3.134)$$

o que resulta

$$\left(\mathbf{M}_\rho + \frac{\alpha \Delta t}{Re} \mathbf{K} \right) \mathbf{u}^{n+1} = -\Delta t \left(\mathbf{A} \mathbf{u}^n + \frac{(1 - \alpha)}{Re} \mathbf{K} \mathbf{u}^n - \mathbf{b} \right) + \mathbf{M}_\rho \mathbf{u}^n \quad (3.135)$$

Assim, pode-se variar o parâmetro α de forma a obter uma família de métodos. Em particular, $\alpha = 0$ resulta em uma discretização explícita, $\alpha = 1$ resulta em discretização semi-implícita e $\alpha = \frac{1}{2}$ resulta no método *Crank-Nicholson* para avanço no tempo. É conhecido na literatura que este método tem um erro de truncamento $\mathcal{O}(\Delta t^2)$, contudo, o tratamento explícito do termo convectivo pode degradar esta precisão. Para melhorar a aproximação explícita do termo convectivo, é possível ainda utilizar um esquema *Adams-Bashforth*, onde este termo é avaliado como uma extrapolação dos tempos t e $t - \Delta t$, ou seja, sendo $\alpha = \frac{1}{2}$

$$\mathbf{A} \mathbf{u}^{n+\frac{1}{2}} = \mathbf{A} \left[\frac{3}{2} \mathbf{u}^n - \frac{1}{2} \mathbf{u}^{n-1} \right] = \frac{3}{2} \mathbf{A} \mathbf{u}^n - \frac{1}{2} \mathbf{A} \mathbf{u}^{n-1} \quad (3.136)$$

e o esquema (3.135) fica

$$\left(\mathbf{M}_\rho + \frac{\Delta t}{2Re} \mathbf{K} \right) \mathbf{u}^{n+1} = -\Delta t \left(\frac{3}{2} \mathbf{A} \mathbf{u}^n - \frac{1}{2} \mathbf{A} \mathbf{u}^{n-1} + \frac{1}{2Re} \mathbf{K} \mathbf{u}^n - \mathbf{b} \right) + \mathbf{M}_\rho \mathbf{u}^n \quad (3.137)$$

Capítulo 4

Métodos de projeção

Este capítulo apresenta uma breve discussão sobre métodos de projeção, desde a decomposição de *Helmholtz-Hodge* até a derivação do método de projeção baseado em decomposição *LU* aproximada.

4.1 Introdução

Os métodos para solução das equações de *Navier-Stokes* para escoamentos de fluidos podem ser, de maneira geral, classificados em métodos *acoplados* e métodos *segregados*.

Métodos acoplados [Zienkiewicz and Taylor, 2000, Fortuna, 2000] buscam resolver o sistema completo de equações a cada ciclo computacional. Devido ao forte acoplamento entre velocidade e pressão, esta é a maneira mais imediata de se resolver estas equações. Tal abordagem, juntamente com as fortes não-linearidades vindas do termo convectivo, torna o problema acoplado computacionalmente difícil e muito caro de se resolver. No caso tridimensional, por exemplo, o sistema não-linear acoplado é composto de quatro equações (três de quantidade de movimento e uma de continuidade) para cada ponto computacional na malha que discretiza o domínio de solução.

Os métodos segregados buscam o desacoplamento entre as equações, separando o sistema não-linear multidimensional em problemas mais simples, que podem ser resolvidos sequencialmente. Dentre os métodos segregados, os que mais se destacaram foram os denominados métodos de projeção. Tal família de métodos foi primeiramente introduzida por Chorin [Chorin, 1968], seguido por muitos outros autores, como por exemplo

[Harlow and Welch, 1965] com o método MAC, [Tomé and McKee, 1994] com método GENSMAC e [Patankar, 1980] com o método SIMPLE. Os métodos de projeção podem ainda ser classificados em métodos contínuos, métodos semi-discretos, ou de passo fracionário [Gresho, 1990, Gresho and Chan, 1990], e métodos discretos, baseados em decomposição LU [Perot, 1993, Ni et al., 2003, Lee et al., 2001].

A seguir são apresentados alguns métodos de projeção disponíveis na literatura, seguido do método utilizado para resolver as equações desenvolvidas nos capítulos anteriores.

4.2 A idéia geral do método da projeção

A família de métodos de projeção é baseada no fato de que qualquer vetor $\mathbf{v} \in \Omega$, onde Ω é um domínio com contorno $\partial\Omega$ suave, pode ser unicamente decomposto da seguinte forma

$$\mathbf{v} = \mathbf{v}_d + \nabla\varphi \quad (4.1)$$

onde $\mathbf{v}_d$ é solenoidal e paralelo ao contorno $\partial\Omega$, isto é,

$$\nabla \cdot \mathbf{v}_d = 0 \quad \text{em } \Omega \quad (4.2)$$

$$\mathbf{v}_d \cdot \mathbf{n} = 0 \quad \text{em } \partial\Omega \quad (4.3)$$

e φ é um campo escalar. Esta decomposição é conhecida como decomposição de *Helmholtz-Hodge*. Da análise vetorial, tem-se $\nabla \times \nabla\varphi = \mathbf{0}$, e portanto (4.1) é equivalente a separar o vetor $\mathbf{v}$ em componentes de divergência nula e rotacional nulo.

Para entender a teoria do método da projeção é necessário interpretar as equações de *Navier-Stokes* como projeções. Para simplificar este estudo, considere as equações na forma conservativa, com μ e ρ constantes em todo o domínio Ω :

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u} \otimes \mathbf{u}) = -\nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{g} \quad (4.4)$$

$$\nabla \cdot \mathbf{u} = 0 \quad (4.5)$$

onde p e ν são pressão e viscosidade cinemáticas, respectivamente. Assim, pode-se escrever (4.4) como

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla p = S(\mathbf{u}), \quad (4.6)$$

onde

$$S(\mathbf{u}) = \nu \nabla^2 \mathbf{u} + \mathbf{g} - \nabla \cdot (\mathbf{u} \otimes \mathbf{u}). \quad (4.7)$$

Note que $S(\mathbf{u})$ não tem, em geral, divergência e rotacional nulos. Note ainda que

$$\nabla \cdot \left(\frac{\partial \mathbf{u}}{\partial t} \right) - \frac{\partial}{\partial t} (\nabla \cdot \mathbf{u}) = 0, \quad (4.8)$$

$$\nabla \times \nabla p = \mathbf{0}. \quad (4.9)$$

Segundo [Chorin, 1968], a equação (4.6) pode ser interpretada da seguinte forma: dado $\mathbf{u}$, o vetor $S(\mathbf{u})$ é conhecido e pode ser projetado em ambos os subespaços de divergência nula ($\partial \mathbf{u} / \partial t$) e rotacional nulo (∇p). Matematicamente,

$$\frac{\partial \mathbf{u}}{\partial t} = \mathcal{P}[S(\mathbf{u})]; \quad \nabla p = \mathcal{Q}[S(\mathbf{u})]; \quad (4.10)$$

onde $\mathcal{P}$ e $\mathcal{Q}$ são operadores de projeção, que satisfazem as seguintes propriedades:

$$\mathcal{P}^2 = \mathcal{P}; \quad \mathcal{Q}^2 = \mathcal{Q}; \quad \mathcal{P}\mathcal{Q} = \mathcal{Q}\mathcal{P} = 0; \quad (4.11)$$

Note ainda que, dado $\mathbf{v}$ um vetor qualquer, $\mathcal{P}$ projeta este vetor no espaço nulo do operador divergente e $\mathcal{Q}$ o projeta no espaço nulo do operador rotacional, isto é

$$\nabla \cdot \mathcal{P}[\mathbf{v}] = 0, \quad \forall \mathbf{v} \in \Omega, \quad (4.12)$$

$$\nabla \times \mathcal{Q}[\mathbf{v}] = \mathbf{0}, \quad \forall \mathbf{v} \in \Omega. \quad (4.13)$$

Segundo [Gresho, 1990], a seguinte forma para os operadores de projeção pode ser obtida de (4.8) e (4.10):

$$\mathcal{P} = \mathbf{I} - \nabla (\nabla^2)^{-1} (\nabla \cdot) \quad (4.14)$$

$$\mathcal{Q} = \mathbf{I} - \mathcal{P} = \nabla (\nabla^2)^{-1} (\nabla \cdot) \quad (4.15)$$

Através destes operadores, pressão e aceleração local podem ser desacopladas das equações de *Navier-Stokes*. De fato, segundo [Gresho, 1990], enquanto pressão e aceleração podem ser calculados sequencialmente, pressão e velocidade não, pois estão intimamente (ou fortemente) acopladas em escoamentos incompressíveis.

No entanto, na esperança de encontrar um método de aproximação que seja menos custoso computacionalmente, Gresho afirma ainda que a filosofia do método da projeção é “tentar assim mesmo”, sendo a legitimidade deste processo amplamente justificada em seus dois artigos [Gresho, 1990, Gresho and Chan, 1990].

4.3 Método de projeção contínuo

O primeiro passo do método de projeção consiste em resolver a aproximação

$$\frac{\partial \tilde{\mathbf{u}}}{\partial t} = S(\mathbf{u}) - \nabla \tilde{p} \quad (4.16)$$

onde $\tilde{p}$ é uma aproximação para a pressão, que pode vir das condições iniciais, ou do passo anterior no algoritmo. A velocidade intermediária $\tilde{\mathbf{u}}$ resultante de (4.16) não é solenoidal, pois em geral, $\tilde{p} \neq p$. Desta forma, $\tilde{\mathbf{u}}$ pode ser projetada no subespaço de divergência nula, utilizando-se o operador $\mathcal{P}$. Assim, uma solução solenoidal pode ser aproximada pela projeção

$$\mathbf{u}_d = \mathcal{P}[\tilde{\mathbf{u}}], \quad (4.17)$$

que pode ser tomada como aproximação da solução real. No entanto, o operador $\mathcal{P}$ é difícil de ser aplicado diretamente, pois ∇^2 pode ser invertido somente através de uma função de *Green* [Jankowski, 1998], sendo necessária a informação das condições de contorno e da geometria do problema em particular. Para evitar a inversão do operador *Laplaciano*, outra aproximação é feita: levando-se em conta a equação (4.1) o passo de projeção pode ser feito utilizando-se seguinte decomposição

$$\tilde{\mathbf{u}} = \mathbf{u}_d + \nabla \varphi, \quad \text{com} \quad \nabla \cdot \mathbf{u}_d = 0 \quad (4.18)$$

onde φ é o multiplicador de *Lagrange* associado com a projeção da solução intermediária $\tilde{\mathbf{u}}$ no subespaço dos vetores de divergência nula ($\mathbf{u}_d$) e rotacional nulo ($\nabla \varphi$). Como $\nabla \times \nabla \varphi = \mathbf{0}$, então $\nabla \times \tilde{\mathbf{u}} = \nabla \times \mathbf{u}_d$, ou seja, a vorticidade contida pela velocidade intermediária $\tilde{\mathbf{u}}$ não é alterada pela projeção. De (4.18) pode-se resolver $\mathbf{u}_d$ e φ através de um procedimento de dois passos:

1. Aplicando-se o operador divergente em (4.18), pode-se encontrar φ através da equação de *Poisson* resultante

$$\nabla^2 \varphi = \nabla \cdot \tilde{\mathbf{u}} \quad (4.19)$$

2. Sendo φ conhecida, pode-se calcular $\mathbf{u}_d$ diretamente de (4.18) como segue

$$\mathbf{u}_d = \tilde{\mathbf{u}} - \nabla \varphi \quad (4.20)$$

No algoritmo descrito acima, a resolução de (4.16) é chamado de passo da velocidade intermediária e a resolução de (4.18) é chamada de passo de projeção.

4.4 Método de passo fracionário

O método de passo fracionário é um método semi-discreto, que parte das equações de *Navier-Stokes* já discretizadas no tempo. Considere as equações (2.52) e (2.55) discretizadas no tempo por um método semi-implícito, dadas por

$$\frac{\rho^{n+1}\mathbf{u}^{n+1} - \rho^n\mathbf{u}^n}{\Delta t} + ((\mathbf{u} - \hat{\mathbf{u}})^n \cdot \nabla)(\rho\mathbf{u})^n = -\nabla p^{n+1} + \frac{1}{Re} \nabla \cdot \{\mu^n [(\nabla\mathbf{u}^{n+1}) + (\nabla\mathbf{u}^{n+1})^T]\} + \mathbf{b} \quad (4.21)$$

$$\nabla \cdot \mathbf{u}^{n+1} = 0 \quad (4.22)$$

onde

$$\mathbf{b} = \frac{1}{Fr^2} \rho^n \mathbf{g} + \frac{1}{We} \mathbf{f}. \quad (4.23)$$

A idéia do método de passo fracionário é aproximar (4.21) calculando uma velocidade tentativa $\tilde{\mathbf{u}}$, utilizando a equação de conservação de quantidade de movimento sem o termo de pressão, e então usar a pressão para projetar a velocidade tentativa no espaço de funções discretas de divergência nula. Matematicamente, este processo consiste em fazer a seguinte separação

$$\frac{\rho^{n+1}\tilde{\mathbf{u}} - \rho^n\mathbf{u}^n}{\Delta t} + ((\mathbf{u} - \hat{\mathbf{u}})^n \cdot \nabla)(\rho\mathbf{u})^n - \frac{1}{Re} \nabla \cdot \{\mu^n [(\nabla\mathbf{u}^{n+1}) + (\nabla\mathbf{u}^{n+1})^T]\} = \mathbf{b} \quad (4.24)$$

$$\frac{\rho^{n+1}\mathbf{u}^{n+1} - \rho^{n+1}\tilde{\mathbf{u}}}{\Delta t} = -\nabla p^{n+1} \quad (4.25)$$

A pressão em (4.25) é encontrada aplicando-se o operador divergente e utilizando (4.22). Este procedimento resulta na equação de Poisson para a pressão dada por

$$\nabla \cdot \frac{1}{\rho^{n+1}} \nabla p^{n+1} = \frac{1}{\Delta t} \nabla \cdot \tilde{\mathbf{u}} \quad (4.26)$$

Note que para a solução de (4.24) e (4.26), é necessária a imposição de condições de contorno para a velocidade tentativa $\tilde{\mathbf{u}}$ e para a pressão p . O procedimento para a solução das equações é dado na seguinte ordem:

1. Resolve-se $\tilde{\mathbf{u}}$ de (4.24);
2. Resolve-se p^{n+1} de (4.26);
3. Encontra-se a velocidade final $\mathbf{u}^{n+1}$ usando (4.25);

4.5 Método baseado em decomposição LU

Enquanto que no método do passo fracionário, a separação (ou *split*) entre velocidade e pressão é feita ainda sobre a equação na forma contínua, discretizada apenas no tempo, no método baseado em decomposição LU , este *split* é feito somente após a discretização espacial das mesmas. Considere as equações de *Navier-Stokes*, discretizadas espacialmente pelo método de elementos finitos, seguindo o que foi feito no capítulo anterior. Tais equações, discretizadas no tempo por um esquema semi-implícito, são dadas por

$$\mathbf{M}_\rho \left(\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} \right) + \mathbf{A}\mathbf{u}^n + \frac{1}{Re} \mathbf{K}\mathbf{u}^{n+1} - \mathbf{G}\mathbf{p}^{n+1} - \frac{1}{Fr^2} \mathbf{M}_\rho \mathbf{g} - \frac{1}{We} \mathbf{M}\mathbf{f} = \mathbf{0} \quad (4.27)$$

$$\mathbf{D}\mathbf{u}^{n+1} = \mathbf{0} \quad (4.28)$$

A equação (4.27) pode ainda ser reescrita como

$$\left(\mathbf{M}_\rho + \frac{\Delta t}{Re} \mathbf{K} \right) \mathbf{u}^{n+1} = -\Delta t \left(\mathbf{A}\mathbf{u}^n - \mathbf{G}\mathbf{p}^{n+1} - \frac{1}{Fr^2} \mathbf{M}_\rho \mathbf{g} - \frac{1}{We} \mathbf{M}\mathbf{f} \right) + \mathbf{M}_\rho \mathbf{u}^n \quad (4.29)$$

que juntamente com (4.28), formam um sistema de equações que pode ser representado da seguinte forma

$$\begin{bmatrix} \mathbf{B} & -\Delta t \mathbf{G} \\ \mathbf{D} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u}^{n+1} \\ \mathbf{p}^{n+1} \end{bmatrix} = \begin{bmatrix} \mathbf{r}^n \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{bc}_1 \\ \mathbf{bc}_2 \end{bmatrix} \quad (4.30)$$

onde agora o sistema é escrito apenas para as incógnitas do problema, ou seja, $\mathbf{u}^{n+1} = [u_1^{n+1}, \dots, u_{Nu}^{n+1}, v_1^{n+1}, \dots, v_{Nv}^{n+1}]^T$, $\mathbf{p}^{n+1} = [p_1^{n+1}, \dots, p_{Np}^{n+1}]^T$, sendo Nu , Nv e Np o número de incógnitas (nós livres) para velocidade na direção x , velocidade na direção y e pressão respectivamente. A notação para as matrizes e vetores foi mantida a mesma por simplicidade. A matriz $\mathbf{B}$ é dada por

$$\mathbf{B} = \mathbf{M}_\rho + \frac{\Delta t}{Re} \mathbf{K} \quad (4.31)$$

e o lado direito representa as grandezas conhecidas no tempo n ,

$$\mathbf{r}^n = -\Delta t \left(\mathbf{A}\mathbf{u}^n - \frac{1}{Fr^2} \mathbf{M}_\rho \mathbf{g} - \frac{1}{We} \mathbf{M}\mathbf{f} \right) + \mathbf{M}_\rho \mathbf{u}^n, \quad (4.32)$$

mais as condições de contorno que nada mais são do que as contribuições dos valores conhecidos de velocidade e pressão no lado direito do sistema.

O método consiste em decompor a matriz do sistema (4.30) através de uma fatoração

por blocos. Em [Lee et al., 2001] são apresentadas várias formas de se fatorar exatamente esta matriz, cada forma dando origem a uma família de métodos diferentes (nem todos práticos ou mesmo possíveis). Utilizando uma fatoração canônica LU por blocos, tem-se o seguinte sistema

$$\begin{bmatrix} \mathbf{B} & \mathbf{0} \\ \mathbf{D} & \Delta t \mathbf{D} \mathbf{B}^{-1} \mathbf{G} \end{bmatrix} \begin{bmatrix} \mathbf{I} & -\Delta t \mathbf{B}^{-1} \mathbf{G} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{u}^{n+1} \\ \mathbf{p}^{n+1} \end{bmatrix} = \begin{bmatrix} \mathbf{r}^n \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{bc}_1 \\ \mathbf{bc}_2 \end{bmatrix} \quad (4.33)$$

O sistema dado em (4.33), se resolvido exatamente, dá origem a um método conhecido como método de *Uzawa* [Chang et al., 2002]. Contudo, tal solução é muito cara computacionalmente, pois envolve a inversão da matriz $\mathbf{B}$ a cada iteração. Métodos práticos para a solução de (4.33) aproximam a inversa de $\mathbf{B}$ pelas matrizes $\mathbf{B}_1$ e $\mathbf{B}_2$ de modo que o sistema pode ser escrito como

$$\begin{bmatrix} \mathbf{B} & \mathbf{0} \\ \mathbf{D} & \Delta t \mathbf{D} \mathbf{B}_1 \mathbf{G} \end{bmatrix} \begin{bmatrix} \mathbf{I} & -\Delta t \mathbf{B}_2 \mathbf{G} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{u}^{n+1} \\ \mathbf{p}^{n+1} \end{bmatrix} = \begin{bmatrix} \mathbf{r}^n \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \mathbf{bc}_1 \\ \mathbf{bc}_2 \end{bmatrix} \quad (4.34)$$

Segundo [Chang et al., 2002], diferentes aproximações para as matrizes $\mathbf{B}_1$ e $\mathbf{B}_2$ podem ser feitas de modo a controlar o erro feito na separação das equações. Se $\mathbf{B}_1 = \mathbf{B}_2$, então a equação da continuidade (4.28) é satisfeita exatamente, e todo o erro das aproximações aparece na equação de conservação de quantidade de movimento. Caso as aproximações sejam diferentes, o erro cometido na separação pode ser movido inteiramente da equação de conservação de quantidade de movimento para a equação da continuidade. Ainda não é claro qual tipo de erro é melhor, sendo a resposta para esta questão dependente do problema a ser resolvido. Contudo, é desejável, principalmente em problemas envolvendo escoamentos incompressíveis, que a equação da continuidade seja satisfeita exatamente (ainda que a nível discreto), para que haja conservação de massa no problema.

Uma aproximação comum para a inversa $\mathbf{B}^{-1}$ para este problema é considerar $\mathbf{B}_1 = \mathbf{B}_2 = \mathbf{M}_\rho^{-1}$ [Chang et al., 2002], e portanto, o sistema (4.34) fica

$$\mathbf{B} \tilde{\mathbf{u}} = \mathbf{r}^n + \mathbf{bc}_1 \quad (4.35)$$

$$\Delta t \mathbf{D} \mathbf{M}_\rho^{-1} \mathbf{G} \mathbf{p}^{n+1} = -\mathbf{D} \tilde{\mathbf{u}} + \mathbf{bc}_2 \quad (4.36)$$

$$\mathbf{u}^{n+1} = \tilde{\mathbf{u}} + \Delta t \mathbf{M}_\rho^{-1} \mathbf{G} \mathbf{p}^{n+1} \quad (4.37)$$

Note que, apesar da semelhança com o método de passo fracionário para a solução do sistema de equações dado acima, não é necessária a imposição de condições de contorno para a velocidade tentativa e para a pressão, como feito para o método semi-discreto de

passo fracionário. O procedimento para a solução das equações é dado na seguinte ordem:

1. Resolve-se $\tilde{\mathbf{u}}$ de (4.35);
2. Resolve-se $\mathbf{p}^{n+1}$ de (4.36);
3. Encontra-se a velocidade final $\mathbf{u}^{n+1}$ usando (4.37);

Métodos de ordem mais altas podem ser obtidos melhorando a aproximação da matriz inversa $\mathbf{B}^{-1}$. Outros métodos podem ser obtidos mudando-se a forma de decompor a matriz do sistema. Por exemplo, em [Ni et al., 2003] é descrito um esquema de segunda ordem, obtido pela adição do termo $\mathbf{G}\mathbf{p}^n$ no lado direito do sistema $\mathbf{r}^n$.

4.5.1 Matriz de massa “lumped”

A aplicação do método descrito nesta seção envolve a inversão de uma matriz de massa $\mathbf{M}_\rho$. Para se obter a matriz do sistema do passo 2 do algoritmo acima (4.36), deve-se inverter $\mathbf{M}_\rho$ a cada iteração, não sendo uma opção computacionalmente atrativa. Uma alternativa, o que é conhecido na literatura por “*lumping*” consiste em aproximar $\mathbf{M}_\rho$ por uma matriz diagonal $\mathbf{M}_{L\rho}$, definida por

$$m_{L\rho,ii} = \sum_{j=1}^N m_{\rho,ij} \quad (4.38)$$

onde N é a dimensão da matriz $\mathbf{M}_\rho$. Tal procedimento soma as contribuições referente às relações de vizinhança entre os vértices do elemento na diagonal da matriz. Para não afetar a restrição de divergência nula a nível discreto, esta aproximação é utilizada para os passos 2 e 3 do algoritmo. O algoritmo final é dado por

1. Resolve-se $\tilde{\mathbf{u}}$ de

$$\mathbf{B}\tilde{\mathbf{u}} = \mathbf{r}^n + \mathbf{b}\mathbf{c}_1 \quad (4.39)$$

2. Resolve-se $\mathbf{p}^{n+1}$ de

$$\Delta t \mathbf{D} \mathbf{M}_{L\rho}^{-1} \mathbf{G} \mathbf{p}^{n+1} = -\mathbf{D}\tilde{\mathbf{u}} + \mathbf{b}\mathbf{c}_2 \quad (4.40)$$

3. Encontra-se a velocidade final $\mathbf{u}^{n+1}$ usando

$$\mathbf{u}^{n+1} = \tilde{\mathbf{u}} + \Delta t \mathbf{M}_{L\rho}^{-1} \mathbf{G} \mathbf{p}^{n+1} \quad (4.41)$$

Capítulo 5

Representação da interface

Este capítulo descreve a discretização da interface entre fluidos no escoamento, no contexto do método *ALE*. A seguir é descrito o método utilizado para calcular a tensão interfacial e sua inclusão nas equações de conservação discretizadas.

5.1 Métodos de representação da interface

Existem várias formas de se representar a interface, que podem ser classificadas em métodos do tipo *front-tracking*, métodos do tipo *front-capturing* e métodos híbridos.

Em métodos do tipo *front-tracking*, a interface é representada por elementos computacionais que se movem com a velocidade do fluido, calculada em uma malha computacional geralmente fixa no espaço. A transferência de informações entre a malha e a interface é feita através de interpolações, que nem sempre conservam fluxos na interface. Uma boa característica deste tipo de método é sua conservação de massa, superior a dos métodos do tipo *front-capturing*. No entanto, mudanças topológicas da interface (como coalescência ou ruptura) são difíceis de se representar nestes métodos. Um método *front-tracking* é o proposto por Tryggvason [Unverdi and Tryggvason, 1992, Tryggvason et al., 2001], o qual utiliza discretizações por diferenças finitas sobre uma malha cartesiana. Neste método, a distribuição de massa específica e viscosidade no domínio é feita resolvendo-se uma equação de *Poisson*. Para a massa específica, por exemplo, considera-se uma função de Heaviside $H : \mathbb{R}^n \rightarrow \{0, 1\}$ definida como 1 onde há um certo fluido e 0 caso contrário, e pode-se escrever

$$\rho = \rho_1 H + \rho_0(1 - H) \quad (5.1)$$

cujo gradiente é dado por

$$\nabla \rho = \rho_1 \nabla H - \rho_0 \nabla H = \Delta \rho \nabla H - \Delta \rho \int_I \delta(\mathbf{x} - \mathbf{x}_I) \mathbf{n} dS \quad (5.2)$$

onde $\Delta \rho = \rho_1 - \rho_0$, e a última igualdade é demonstrada em [Tryggvason et al., 2001]. A distribuição de massa específica é obtida aplicando-se o operador divergente a esta última equação e resolvendo-se a equação de *Poisson* resultante. Um procedimento análogo é feito para a distribuição de viscosidades na malha. Tal procedimento espalha estas propriedades na interface, e pode causar erros na aproximação longe dela.

Ao contrário de métodos *front-tracking*, métodos do tipo *front-capturing* buscam reconstruir a interface a toda iteração, não havendo informações explícitas sobre a posição da mesma. Exemplos são os métodos VOF (*Volume of Fluid*) [Hirt and Nichols, 1981, Ginzburg and Wittum, 2001], onde funções de volume são utilizadas para a reconstrução da interface, e o método *level-set* [Sussman et al., 1994, Sussman and Smereka, 1997], onde a interface é representada pelo conjunto de nível zero de uma função suave ϕ . Neste método, o avanço da interface é feito através da solução de uma equação de transporte

$$\frac{\partial \phi}{\partial t} + \mathbf{u} \cdot \nabla \phi = 0, \quad (5.3)$$

e a distribuição de propriedades é facilmente obtida através de funções *Heaviside* suavizadas a partir da função *level-set*, ou seja,

$$H_\varepsilon(\phi) = \begin{cases} 0 & \text{se } \phi < -\varepsilon \\ \nu(\phi/\varepsilon) & \text{se } -\varepsilon \leq \phi \leq \varepsilon \\ 1 & \text{se } \phi > \varepsilon \end{cases} \quad (5.4)$$

sendo $\nu(\xi)$ uma transição suave entre os valores 0 e 1 assumidos por $H(\phi)$. Por exemplo, em [Sussman and Smereka, 1997] é utilizado

$$\nu(\xi) = \frac{1}{2} \left\{ 1 + \xi + \frac{1}{\pi} \sin(\pi \xi) \right\}, \quad (5.5)$$

enquanto que em [Tornberg and Engquist, 2000] é utilizada a seguinte função

$$\nu(\xi) = \frac{1}{2} + \frac{1}{32} (45\xi - 50\xi^3 + 21\xi^5). \quad (5.6)$$

Deste modo, as propriedades dos fluidos podem ser aproximadas como

$$\rho(\phi) = \rho_1 H_\varepsilon(\phi) + \rho_0(1 - H_\varepsilon(\phi)) \quad (5.7)$$

$$\mu(\phi) = \mu_1 H_\varepsilon(\phi) + \mu_0(1 - H_\varepsilon(\phi)) \quad (5.8)$$

Note que, devido à suavização da função de *Heaviside*, há também um espalhamento de ρ e μ na interface. Uma vantagem desse tipo de representação é que a curvatura pode ser calculada a partir da função de *level-set*, ou seja,

$$\kappa = \nabla \cdot \mathbf{n} \quad \text{onde} \quad \mathbf{n} = \frac{\nabla \phi}{|\nabla \phi|} \quad (5.9)$$

Porém, para que este cálculo se mantenha preciso, é importante que ϕ seja sempre uma função distância, ou seja, $|\nabla \phi| = 1$. No entanto, o transporte de ϕ dado por (5.3) pode não preservar esta propriedade, sendo necessário algum processo de reinicialização de ϕ .

Apesar da facilidade na representação da interface, principalmente no tratamento de mudanças topológicas, o método *level-set* tem problemas de conservação de massa, originados em grande parte da discretização de (5.3). Há vários trabalhos na literatura que tentam minimizar estes erros. Por exemplo, em [van der Pijl et al., 2003], é proposto um método que tem boa conservação de massa, combinando os métodos VOF e *level-set*, e em [Sousa and Mangiavacchi, 2004] é implementado um método *level-set* sobre uma malha *Lagrangeana*, eliminando-se a necessidade de resolver (5.3).

Há ainda uma terceira classe de métodos para representação da interface, denominada de métodos híbridos. Tais métodos combinam características de *front-tracking* e *front-capturing*, como o método de projeção de segmentos [Törnberg, 2000], que representa a interface através de pontos e segmentos nas direções x e y . Este método tem uma conservação de massa melhor que o método *level-set*, e ainda preserva a facilidade de representação de trocas topológicas na interface.

Um outro método, que também pode ser classificado como um híbrido entre *front-tracking* e *front-capturing*, foi desenvolvido em [Sousa et al., 2004], onde a interface é representada por uma malha *Lagrangeana* 2D movendo-se sobre uma malha *Euleriana* 3D. A malha 2D marca a posição da interface e é transportada por velocidades interpoladas da malha 3D. Nesta malha, velocidade e pressão são calculados utilizando-se esquemas de diferenças finitas. Para a distribuição de forças na malha *Euleriana*, a interface é representada pelo gradiente de uma função *Heaviside* discreta, obtida a partir da classificação das células nesta malha.

Figura 5.1: Exemplo de representação da interface.

5.1.1 Representação *Lagrangeana* da interface no método *ALE*

Na metodologia *Lagrangeana-Euleriana* utilizada neste trabalho, os pontos da malha movem-se com velocidade $\hat{\mathbf{u}}$ arbitrária. Assim, a maneira mais imediata de se representar a interface entre fluidos é utilizar um conjunto de pontos e arestas pertencentes à malha triangulada, porém, algumas restrições devem ser impostas sobre estes pontos e arestas:

1. A movimentação dos pontos de interface deve ser feita de acordo com o campo de velocidade do escoamento, de modo a permanecer sempre sobre a mesma;
2. As arestas que dividem elementos de fluidos diferentes devem permanecer na triangulação, ou serem substituídas por arestas que ainda preservem as informações contidas nos elementos.

A figura (5.1) mostra como é feita a representação da interface separando dois fluidos. Adicionalmente, cada célula da malha computacional carrega informação da fase a qual este elemento pertence, o que facilita a identificação das arestas de interface e a distribuição das propriedades dos fluidos.

A movimentação *Lagrangeana* dos pontos caracteriza um método do tipo *front-tracking*. No entanto, a força de tensão interfacial é calculada e distribuída segundo o gradiente de uma função *Heaviside*, tal como em [Sousa et al., 2004], construída a partir de informações contidas na classificação de células na malha. Esta distribuição dá ao método uma característica de métodos do tipo *front-capturing*, onde a interface computacional (aquela onde são aplicadas as forças de tensão interfacial) é obtida através do gradiente de uma função marcadora.

Figura 5.2: Determinação direta da força concentrada, aplicada pontualmente nos vértices de interface da malha.

5.2 Cálculo da tensão interfacial

Como visto no capítulo 2, há vários modelos para o termo de tensão interfacial, adicionado como fonte na equação de conservação de quantidade de movimento.

Da mesma forma, há várias maneiras de se discretizar este termo na representação interfacial dada acima. A forma mais imediata é avaliar o efeito da tensão interfacial como sendo uma força concentrada agindo nos nós da interface. A força concentrada resultante f_C em cada vértice é obtida diretamente da definição de tensão interfacial como sendo

$$f_C = f_1 + f_2 = \sigma(t_2 - t_1) \quad (5.10)$$

onde f_1 e f_2 são as tensões tangenciais à interface que agem sobre um vértice, como na figura (5.2).

No entanto, a aplicação direta da força interfacial como uma força concentrada exige uma discretização que permita representar a descontinuidade de pressão presente na interface. Se aproximações contínuas são utilizadas, pode haver oscilações ou efeitos de “*over/undershooting*” para a pressão. Como o elemento escolhido neste trabalho para a discretização das equações utiliza uma aproximação contínua linear por partes para a pressão (mini-elemento), a aplicação direta da força não é adequada.

Uma forma de se eliminar os efeitos de “*over/undershooting*” para os elementos contínuos é utilizar uma distribuição da força interfacial que seja compatível com uma distribuição de pressão linear por partes. Assim, considere o termo de tensão interfacial dado por

$$f_D = -\sigma\kappa\nabla H. \quad (5.11)$$

Neste esquema, f_D representa a força distribuída de tensão interfacial por unidade de volume. A intensidade da força ($\sigma\kappa$) é calculada e projetada na direção do gradiente de uma função *Heaviside* contínua e linear por partes, que distribui a força de tensão interfacial nos nós livres vizinhos aos vértices. A seguir são apresentadas as aproximações

Figura 5.3: Curvatura interpretada como variação dos vetores normal e tangente no comprimento de arco infinitesimal ds , e sua discretização.

para estes dois cálculos.

5.2.1 Cálculo da curvatura e distribuição da força

Utilizando as fórmulas de *Frenet* em 2D [Kreyszig, 1991], a curvatura κ pode ser interpretada como a variação do vetor normal ou vetor tangente ao longo da curva que define a interface, ou seja,

$$\kappa \mathbf{n} = \frac{\partial \mathbf{t}}{\partial s}, \quad \kappa \mathbf{t} = -\frac{\partial \mathbf{n}}{\partial s}. \quad (5.12)$$

Discretizando $\kappa \mathbf{n}$ na interface computacional, obtém-se

$$\kappa \mathbf{n} = \frac{\mathbf{t}_2 - \mathbf{t}_1}{\Delta s}, \quad (5.13)$$

onde $\mathbf{t}_1$ e $\mathbf{t}_2$ são as aproximações dos vetores tangentes à interface calculados nos centros das arestas da triangulação, e Δs é o comprimento de arco aproximado entre estes vetores, conforme ilustrado na figura 5.3. Pode-se observar também a semelhança entre as equações (5.10) e (5.13), que demonstra que

$$\mathbf{f}_C = \sigma \kappa \mathbf{n} \Delta s, \quad (5.14)$$

ou ainda

$$\sigma \kappa = \frac{1}{\Delta s} \mathbf{f}_C \cdot \mathbf{n}. \quad (5.15)$$

Das equações (5.11) e (5.13), tem-se

$$\mathbf{f}_D = -\frac{1}{\Delta s} \mathbf{f}_C \cdot \mathbf{n} \nabla H. \quad (5.16)$$

Note que $\mathbf{f}_C$ e $\mathbf{n}$ têm a mesma direção, sendo que o sinal de $\mathbf{f}_C \cdot \mathbf{n}$ determina o sentido correto de aplicação da intensidade da força $|\sigma \kappa| = |\mathbf{f}_C \cdot \mathbf{n}|$ na direção de ∇H . Desta

forma, pode-se determinar $\mathbf{n}$ como sendo

$$\mathbf{n} = -\frac{\nabla H}{|\nabla H|}, \quad (5.17)$$

e portanto

$$\mathbf{f}_D = \frac{1}{\Delta s} \mathbf{f}_C \cdot \frac{\nabla H}{|\nabla H|} \nabla H, \quad (5.18)$$

que representa um esquema de distribuição da força $\mathbf{f}_C$ que é compatível com uma aproximação contínua e linear por partes do campo de pressão.

5.2.2 Discretização do termo de tensão interfacial

Como visto acima, a distribuição da força de tensão interfacial é feita considerando-se o gradiente de uma função *Heaviside*. Para a definição desta função, deve-se levar em consideração que ∇H e ∇p devem estar no mesmo espaço. Como, para o elemento utilizado, p é definido no espaço de funções contínuas lineares por parte, toma-se H neste mesmo espaço, bastando para isso, definir o valor de H nos vértices da malha. Finalmente, toma-se

$$H_{\lambda,j} = \begin{cases} 1, & \text{se o vértice } j \text{ pertence à fase } \lambda \\ \frac{1}{2}, & \text{se o vértice } j \text{ pertence à interface da fase } \lambda \\ 0, & \text{se o vértice } j \text{ não pertence à fase } \lambda \end{cases} \quad (5.19)$$

Na prática, não é necessário construir H e calcular ∇H em todo o domínio, bastando calcular ∇H diretamente nos nós dos elementos vizinhos da interface.

A discretização do termo de tensão interfacial (5.11) pode ser feita utilizando uma formulação variacional, como descrito no capítulo 3. Considerando a distribuição da força vista aqui, obtém-se a aproximação

$$\mathbf{M}\mathbf{f}_D = \mathbf{\Sigma}\mathbf{G}\mathbf{h}_\lambda, \quad (5.20)$$

onde $\mathbf{\Sigma}$ é uma matriz diagonal que tem como elementos $\sigma\kappa_1, \dots, \sigma\kappa_{NV}$, que são aproximações para a intensidade da pressão capilar nos nós livres de velocidade, e $\mathbf{h}_\lambda = [H_{\lambda,1}, \dots, H_{\lambda,NP}]^T$ é a função de *Heaviside* discreta nos nós livres de pressão. Note que a curvatura é definida somente nos vértices de interface, mas para caso da força distribuída, é também definida nos nós vizinhos (de velocidade) como sendo uma média ponderada dos valores definidos na interface. Assim, o termo $\mathbf{\Sigma}\mathbf{G}\mathbf{h}_\lambda$ resulta em um vetor de dimensão NV . Na equação acima, as matrizes de massa $\mathbf{M}$ e de gradiente $\mathbf{G}$ são as mesmas definidas na seção 3.2.5. Finalmente, a discretização do vetor de força distribuída pode ser escrita

Figura 5.4: Mudança topológica na interface: os elementos inválidos são identificados (esquerda) e suas fases trocadas (direita).

como

$$\mathbf{f}_D = \mathbf{M}^{-1}(\mathbf{\Sigma G h}_\lambda) \quad (5.21)$$

e pode ser substituída na equação de conservação de quantidade de movimento linear discreta (3.125).

5.3 Captura de mudanças topológicas na interface

Aposar da interface ser identificada explicitamente por pontos e arestas da malha computacional, é possível capturar mudanças topológicas na interface utilizando as informações sobre as fases contidas nos elementos.

O método consiste em identificar na malha de elementos finitos aqueles que são inválidos, ou seja, possuem os três vértices sobre duas interfaces diferentes. Neste caso, basta alterar a fase dos elementos inválidos e atualizar as arestas, obtendo assim a coalescência das duas interfaces ou colapso da película. A figura 5.4 mostra como este procedimento é feito.

Vale a pena enfatizar que uma fiel representação das mudanças topológicas em um escoamento é difícil de se obter, sendo necessária uma resolução de malha ao nível das perturbações que acontecem nas interfaces quando estas se aproximam. Como o método proposto aqui considera a distância equivalente a um elemento como sendo a mínima possível entre as interfaces, resoluções mais grosseiras podem antecipar a coalescência ou ruptura, mudando completamente a física do escoamento.

Por outro lado, devido às suas características de adaptatividade, este método admite reduzir o tamanho dos elementos na proximidade das interfaces, permitindo a simulação do processo de mudança topológica com a precisão que for desejada.

Capítulo 6

Controle de malha

Este capítulo descreve o método de controle de malha, desde o cálculo da velocidade de movimentação dos pontos até o método de inserção e remoção baseado em triangulações de *Delaunay*. Também descreve estratégias de refinamento adaptativo em malhas dinâmicas.

6.1 Velocidade da malha dinâmica

Na formulação *Lagrangeana-Euleriana* arbitrária, a velocidade $\hat{\mathbf{u}}$, conhecida como velocidade do referencial ou ainda velocidade da malha, determina a movimentação dos pontos pertencentes a malha de elementos finitos que discretiza o problema. Como visto no capítulo 2, em uma estratégia puramente *Euleriana* esta velocidade é nula, enquanto que em uma estratégia puramente *Lagrangeana*, ela é a própria velocidade do escoamento. No entanto, em um caso intermediário, onde $\hat{\mathbf{u}}$ é diferente de zero e da velocidade do escoamento, tem-se um movimento arbitrário da malha. É desejável que este movimento reposicione os pontos de forma a melhorar a razão de aspecto dos elementos, segundo alguma métrica pré-estabelecida, para melhorar a aproximação do método. De maneira geral, pode-se definir

$$\hat{\mathbf{u}} = \beta_1 \mathbf{u} + \beta_2 \mathbf{u}_e \quad (6.1)$$

onde $\mathbf{u}$ é a velocidade do escoamento e $\mathbf{u}_e$ é uma velocidade dita *elástica*, normalmente definida segundo alguma estratégia de suavização da malha. As constantes $0 \leq \beta_1, \beta_2 \leq 1$ controlam as contribuições de cada velocidade na velocidade final da malha. Assim, $\beta_1 = 1$ e $\beta_2 = 0$ resultam em uma movimentação puramente *Lagrangeana*, enquanto que $\beta_1 = 0$

e $\beta_2 = 1$ resultam em uma movimentação puramente elástica dos pontos da malha. Note ainda que as constantes não são necessariamente excludentes, ou seja, pode-se combinar as duas estratégias utilizando-se por exemplo $\beta_1 = 1$ e $\beta_2 = 1$, permitindo que os pontos tendam a se mover no sentido do escoamento, mas sejam ainda controlados pela suavização da malha.

6.1.1 Suavização da malha

Como visto acima, a velocidade elástica $\mathbf{u}_e$ é determinada segundo um método de suavização, buscando o reposicionamento dos pontos de modo que os elementos satisfaçam algum critério geométrico, geralmente resultando em uma malha isotrópica, ou físico, podendo resultar em uma malha anisotrópica.

A maioria destes métodos envolvem processos iterativos que deslocam cada vértice da malha individualmente para melhorar a qualidade local dos elementos. Os métodos de suavização podem ser classificados em

- **Métodos baseados em médias:** Métodos que reposicionam os vértices da malha considerando-se alguma média das coordenadas dos vértices vizinhos. O método mais conhecido é a suavização *Laplaciana*, que calcula as novas coordenadas dos vértices como sendo a média das coordenadas dos vértices adjacentes, ou seja,

$$\hat{\mathbf{x}}_i = \frac{1}{n} \sum_{j \in S(\mathbf{x}_i)} \mathbf{x}_j \quad (6.2)$$

onde $S(\mathbf{x}_i)$ é o conjunto dos pontos adjacentes a $\mathbf{x}_i$ e n é o número de vértices em $S(\mathbf{x}_i)$. Esta aproximação não é suficiente para redistribuir os pontos de maneira ótima, mas quando aplicada sistematicamente, num processo iterativo, converge para uma distribuição suave dos pontos, melhorando a qualidade dos elementos. A média dada em (6.2) funciona bem quando o polígono formado pelos pontos em $S(\mathbf{x}_i)$ é convexo, e pode ainda criar elementos inválidos se este polígono não é convexo, conforme exemplo na figura 6.1. Há várias versões da suavização *Laplaciana*, inclusive métodos que restringem o movimento inválido dos vértices, chamados de suavização *Laplaciana* inteligente.

- **Métodos baseados em otimização:** São métodos que buscam o reposicionamento dos vértices através da solução de problemas de otimização. Os mais comuns buscam a minimização de erros baseando-se em critérios geométricos, como por exemplo a minimização da razão de aspecto dos elementos. Malhas anisotrópicas podem

Figura 6.1: Exemplo do reposicionamento de um vértice pela suavização *Laplaciana*: caso convexo e não convexo.

ser obtidas utilizando-se métricas que tenham informações sobre o processo físico simulado na minimização da razão de aspecto. Várias medidas de qualidade e métricas isotrópicas e anisotrópicas podem ser vistas em [Shewchuk, 1997];

- **Métodos baseados em simulações físicas:** Reposicionam os pontos a partir da simulação de um sistema físico com forças de atração e repulsão, buscando um regime de equilíbrio. Há basicamente duas abordagens para este método: considerar um sistema onde as arestas que ligam os vértices são vistas como molas, ou considerar que os vértices são partículas com forças de atração e repulsão, dependendo da distância entre elas, resultando em um problema de minimização de energia potencial. Normalmente o problema resulta na solução de equações diferenciais ordinárias, tornando-se um método caro se utilizado com frequência.

Os métodos citados acima são processos iterativos, onde a cada iteração obtém-se uma aproximação para uma malha ótima. Como neste trabalho considera-se uma malha dinâmica que se move a toda iteração do método de projeção utilizado para a solução das equações de *Navier-Stokes*, a convergência para uma malha ótima é computacionalmente custosa e desnecessária. Assim, para o cálculo da velocidade elástica em (6.1), considera-se apenas uma iteração do método de suavização *Laplaciana*, ou seja, para cada iteração do método de projeção, conhecido o tamanho do passo no tempo Δt , a velocidade elástica da malha é calculada como

$$\mathbf{u}_e(\mathbf{x}_i, t + \Delta t) = \frac{\hat{\mathbf{x}}_i - \mathbf{x}_i}{\Delta t}, \quad (6.3)$$

onde $\mathbf{x}_i$ é a posição do vértice i e a sua nova posição $\hat{\mathbf{x}}_i$ é calculada por (6.2). Note que, como este cálculo está inserido em um processo iterativo, a tendência é a suavização da malha ao longo do tempo, dependendo do parâmetro β_2 utilizado, mesmo que a malha

Figura 6.2: Exemplo da movimentação de pontos na superfície de uma bolha ascendendo em um fluido, degradando a sua representação discreta.

esteja se movendo com a velocidade do fluido, controlada pelo parâmetro β_1 .

6.1.2 Movimentação da interface

No método *Lagrangeano-Euleriano* arbitrário utilizado neste trabalho, a velocidade $\hat{u}$ dada em (6.1) é utilizada para a movimentação dos pontos da malha de elementos finitos que discretiza o domínio. Contudo, como visto no capítulo 5, a interface entre os fluidos é representada por vértices e arestas pertencentes à malha, e esta deve ser movida com a velocidade do fluido, portanto, a velocidade que movimenta os pontos de interface não pode ser calculada diretamente de (6.1).

No entanto, o movimento da interface utilizando-se a velocidade do fluido leva a uma distribuição não uniforme, aumentando a necessidade de se fazer um controle de densidade dos pontos que discretizam a interface, inserindo-os ou removendo-os sempre que as arestas fiquem muito grandes ou pequenas, respectivamente. Especialmente no caso em que os pontos de interface fazem parte da triangulação, este controle de densidade é necessário para garantir que os elementos adjacentes não tenham sua qualidade prejudicada.

Um exemplo que demonstra a necessidade desse controle é a simulação de ascensão de bolhas em um fluido. Neste tipo de escoamento, o campo de velocidade obtido tende a arrastar os vértices da parte de cima da bolha para a parte de baixo, tornando necessária a inserção de pontos na parte de cima e a remoção de pontos na parte de baixo da bolha (veja exemplo na figura 6.2). Como a inserção e remoção de pontos resultam na necessidade de se fazer interpolações para os valores não conhecidos e a perda de valores conhecidos, é desejável minimizar tais operações para minimizar a necessidade de interpolações e perda de informações.

Aproveitando-se da flexibilidade do método *ALE*, dois esquemas de movimentação dos pontos na interface foram considerados neste trabalho. O primeiro consiste em uti-

Figura 6.3: Movimentação de um ponto na interface em um incremento de tempo diferencial $dt \rightarrow 0$, do tempo t para o tempo $t + dt$, utilizando a velocidade do escoamento $\mathbf{u}$ e a velocidade $\mathbf{u}_I$ calculada por (6.5).

lizar somente a componente normal da velocidade do escoamento para a movimentação da interface, ou seja,

$$\mathbf{u}_I = (\mathbf{u} \cdot \mathbf{n})\mathbf{n}, \quad (6.4)$$

onde $\mathbf{u}$ é o campo de velocidade do escoamento e $\mathbf{n}$ é o vetor normal à interface. Note ainda que alternativamente ao cálculo da projeção da velocidade na direção normal, pode-se eliminar a componente tangencial subtraindo-se de $\mathbf{u}$ a sua projeção na direção tangencial (veja figura 6.3), ou seja,

$$\mathbf{u}_I = \mathbf{u} - \beta_3(\mathbf{u} \cdot \mathbf{t})\mathbf{t}, \quad (6.5)$$

onde $0 \leq \beta_3 \leq 1$ introduzido aqui é um fator de controle da aplicação deste esquema. Esta estratégia tende a preservar a distribuição dos pontos, minimizando a necessidade do controle de densidade.

O segundo esquema consiste em considerar somente a componente tangencial da velocidade elástica na interface, ou seja,

$$\mathbf{u}_I = \mathbf{u} + \beta_2(\mathbf{u}_e \cdot \mathbf{t})\mathbf{t}, \quad (6.6)$$

onde $\mathbf{t}$ é o vetor tangente à interface (ver figura 6.4). Assim, é permitido o movimento dos vértices na direção tangencial à interface, e como a velocidade elástica é computada de modo a suavizar a malha, este esquema faz com que o movimento original dado pelo campo de velocidade $\mathbf{u}$ seja suavizado somente na direção tangencial, não prejudicando o movimento da interface e garantindo uma distribuição mais uniforme dos pontos.

Finalmente, combinando-se os dois esquemas de movimentação da interface, a veloci-

Figura 6.4: Movimentação de um ponto na interface em um incremento de tempo diferencial $dt \rightarrow 0$, do tempo t para o tempo $t + dt$, utilizando a velocidade do escoamento $\mathbf{u}$ e a velocidade $\mathbf{u}_f$ calculada por (6.6).

dade da malha $\hat{\mathbf{u}}$ calculada neste trabalho pode ser generalizada como

$$\hat{\mathbf{u}}(\mathbf{x}) = \begin{cases} \beta_1 \mathbf{u} + \beta_2 \mathbf{u}_e, & \text{se } \mathbf{x} \text{ não pertence à interface} \\ \mathbf{u} + \beta_2 (\mathbf{u}_e \cdot \mathbf{t}) \mathbf{t} - \beta_3 (\mathbf{u} \cdot \mathbf{t}) \mathbf{t}, & \text{se } \mathbf{x} \text{ pertence à interface} \end{cases} \quad (6.7)$$

Note que os esquemas mostrados acima são exatos num incremento de tempo infinitesimal, contudo, os erros cometidos pela aproximação destes esquemas são da mesma ordem do erro cometido na integração da velocidade para obter a nova posição dos pontos.

6.2 Controle da malha dinâmica

A malha inicial deve ser uma triangulação, como visto anteriormente. Para garantir boas propriedades de qualidade nesta malha, esta é obtida através de uma triangulação de *Delaunay*. A seguir são discutidas as propriedades da triangulação de *Delaunay*, e um algoritmo para sua geração.

6.2.1 Triangulação de *Delaunay*

Seja $\Omega \subset \mathbb{R}^m$ um domínio limitado e fechado, $A = \{\mathbf{x}_0, \dots, \mathbf{x}_m\} \subset \Omega$ um conjunto de pontos em posição geral, T uma triangulação de Ω de forma que os vértices de todos os elementos de T estão em A , e ainda

$$\Omega = \bigcup_{t_i \in T} t_i, \quad (6.8)$$

Figura 6.5: (a) Exemplo de uma triangulação que falha no critério de *Delaunay*; (b) Exemplo de uma triangulação de *Delaunay*.

com $t_i \cap t_j = \emptyset$ ou $t_i \cap t_j$ é uma aresta ou vértice comum a t_i e t_j . Diz-se que T é uma triangulação de *Delaunay* de A se a circunferência de todo elemento de T não contém nenhum ponto de A em seu interior. Exemplos de triangulações que respeitam ou não este critério podem ser vistos na figura 6.5

Considere ainda uma decomposição de $\mathbb{R}^m$ em células c_i de modo que

$$y \in c_i \Leftrightarrow d(y, x_i) < d(y, x_j), \quad \forall x_j \in A, x_j \neq x_i, \quad (6.9)$$

onde $d(\cdot, \cdot)$ é uma medida de distância. Note que c_i é o conjunto de todos os pontos de $\mathbb{R}^m$ que estão mais próximos de x_i do que qualquer outro ponto de A . A decomposição de $\mathbb{R}^m$ feita desta forma é chamada de *Diagrama de Voronoi*, e pode conter células finitas e infinitas.

Existe uma correspondência biunívoca entre a triangulação de *Delaunay* e o diagrama de *Voronoi*, que associa cada vértice da triangulação de *Delaunay* a uma célula de *Voronoi*, cada triângulo da triangulação de *Delaunay* a um vértice do diagrama de *Voronoi* e cada aresta da triangulação de *Delaunay* a sua aresta correspondente de *Voronoi*. Tal correspondência é chamada de dualidade, ou seja, a triangulação de *Delaunay* é dual do diagrama de *Voronoi*. Um exemplo de triangulação de *Delaunay* e seu dual diagrama de *Voronoi* pode ser visto na figura 6.6. Graças a essa dualidade, é possível provar que a triangulação de *Delaunay* existe e é única, através da existência e unicidade do diagrama de *Voronoi*.

Uma triangulação de *Delaunay* possui as seguintes propriedades:

Figura 6.6: Exemplo de triangulação de *Delaunay* e diagrama de *Voronoi*.

Figura 6.7: Exemplos de arestas que possui (esquerda) e que não possui (direita) *flipping*.

- A triangulação de *Delaunay* é única;
- Todo elemento da triangulação possui circunsfera vazia, ou seja, sem pontos do conjunto A em seu interior;
- Maximiza o ângulo mínimo de todos os elementos (somente em $\mathbb{R}^2$);
- Minimiza o maior circuncírculo da triangulação (somente em $\mathbb{R}^2$);
- Minimiza o maior círculo mínimo da triangulação (o menor círculo que contém o elemento; somente em $\mathbb{R}^2$);

Tais propriedades garantem que a triangulação de *Delaunay* de um conjunto de pontos A é a melhor possível, no sentido de tentar minimizar o aparecimento de elementos muito finos, o que pode degradar a solução numérica por elementos finitos.

6.2.2 *Flipping* 2D

Considere um conjunto de quatro pontos e o quadrilátero formado por eles, como na figura (6.7). Este quadrilátero possui duas diagonais chamadas de *flipping* uma da outra. Se

Figura 6.8: Definição de aresta localmente *Delaunay*.

uma das diagonais é externa ao quadrilátero, diz-se que a diagonal interna não possui *flipping*.

Definição 1 (Aresta localmente Delaunay) *Seja e uma aresta compartilhada por dois triângulos t_1 e t_2 , e x_1 , x_2 os vértices opostos a e em t_1 e t_2 , como na figura 6.8. A aresta e é chamada de localmente Delaunay se o circuncírculo de t_1 não inclui x_2 e o circuncírculo de t_2 não inclui x_1 .*

O seguinte teorema faz a ligação entre a definição de triangulação de *Delaunay* e as arestas localmente *Delaunay*, e sua demonstração pode ser encontrada em [Shewchuk, 1997].

Teorema 6.1 *Seja T uma triangulação onde todas as arestas são localmente Delaunay. Então T é a triangulação de Delaunay.*

Assim, finalmente é possível chegar ao resultado que dá origem aos algoritmos baseados em *flipping* para construção de triangulações de *Delaunay*, que é colocado aqui na forma de um teorema:

Teorema 6.2 (Flipping) *Seja e uma aresta de uma triangulação T qualquer. Então, ou e é localmente Delaunay, ou possui *flipping* e seu *flipping* é localmente Delaunay.*

Este teorema diz que qualquer aresta de uma triangulação que não seja localmente *Delaunay*, pode ser “flipada” para obter uma aresta localmente *Delaunay*. Tal resultado é utilizado para as estratégias de inserção e remoção de pontos descritas a seguir.

6.2.3 Inserção e remoção de pontos

A movimentação dos vértices da malha segundo a velocidade $\hat{u}$ dada em (6.7) pode gerar elementos distorcidos, que podem prejudicar a precisão da aproximação por elementos

finitos. Apesar dos esquemas de suavização de malha apresentados anteriormente, a interface se move livremente, fazendo com que a distribuição de pontos mude drasticamente ao longo da simulação. Um controle de malha deve então ser aplicado, inserindo novos vértices onde a densidade de pontos diminui (seguindo algum critério de medição), e removendo vértices onde a densidade de pontos é muito alta. Todo este processo deve ser feito de forma a manter uma triangulação do domínio com certas garantias de qualidade, para não prejudicar o método de aproximação.

A malha inicial é gerada através de uma triangulação de *Delaunay*, como visto acima, porém, com o movimento dos pontos, esta deixa de ter as propriedades de uma triangulação de *Delaunay*. Manter a malha sempre *Delaunay* a todo passo é computacionalmente muito custoso, e até desnecessário, bastando que a qualidade dos elementos seja garantida.

A seguir são descritas estratégias de inserção e remoção de pontos na triangulação que preservam, ao menos localmente, as propriedades de uma triangulação de *Delaunay*.

Estratégia de inserção de pontos

Dada uma malha inicial que seja uma triangulação de *Delaunay* de um domínio Ω , a estratégia de refinamento desenvolvida aqui busca subdividir os elementos onde uma precisão maior é necessária, de maneira que, localmente, a malha resultante seja *Delaunay*. Este procedimento é chamado de enriquecimento de malha. Mantendo a malha localmente *Delaunay*, está se garantindo um certo critério de qualidade para os novos elementos inseridos, o que é bastante desejável para não degradar a solução numérica.

Seja e uma aresta de tamanho ℓ , esta é subdividida pela inserção de um novo vértice em seu centro sempre que $\ell > \ell_{max}$, onde ℓ_{max} é determinado segundo algum critério de qualidade. A figura 6.9 ilustra a inserção de um ponto no centro de e .

A inserção de um novo vértice v pode infringir a propriedade de *Delaunay* de algum dos seus triângulos vizinhos. Seja $S(v)$ o conjunto de todos os triângulos que contém v , denominado *estrela* de v , e seja $L(v)$ o conjunto de todas as arestas dos triângulos de $S(v)$ que não têm v como extremidade, denominado de *link* de v . Para deixar a malha novamente *Delaunay*, testa-se somente as arestas que estão em $L(v)$.

Caso a aresta testada não seja localmente *Delaunay*, basta trocá-la por seu *flipping* que, de acordo com o teorema 6.2, é localmente *Delaunay* (veja figura 6.9 (c)). A partir daí, testa-se as novas arestas de $L(v)$, e assim por diante. Este procedimento pára quando a vizinhança do vértice v é *Delaunay*. Note que o número máximo de arestas testadas é $\mathcal{O}(n)$, onde n é o número total de pontos na triangulação.

Figura 6.9: Etapas do processo de refinamento: (a) Aresta a ser subdividida; (b) Inserção de vértice infringiu a propriedade de *Delaunay* de um triângulo; (c) A aresta que não é localmente *Delaunay* é trocada por seu *flipping*; (d) Exemplo de malha resultante, se todas as arestas forem subdivididas.

Se o procedimento acima for aplicado a todas as arestas da triangulação, obtém-se uma malha com espaçamento menor que a malha original, mas que mantém a mesma estrutura quanto a distribuição dos pontos. Um exemplo do resultado final pode ser visto na figura 6.9 (d).

Estratégia de remoção de pontos

Dada uma malha inicial que seja localmente uma triangulação de *Delaunay* de um domínio Ω , a estratégia de remoção de pontos busca remover da triangulação arestas de tamanho ℓ tal que $\ell < \ell_{min}$. No entanto, há várias maneiras de se remover uma aresta de uma triangulação como, por exemplo, o colapso de seus vértices em um ponto médio ou ainda remoção da aresta e todos os seus vizinhos. Para manter localmente a propriedade de *Delaunay* da malha, neste trabalho optou-se por remover na aresta a ser deletada, o vértice que tenha a menor somatória dos comprimentos das arestas adjacentes. Assim, além de resumir o processo de remoção de arestas a um problema de remoção de um vértice da triangulação de *Delaunay*, a escolha do vértice que tenha menor somatória dos comprimentos das arestas adjacentes procura aumentar o tamanho médio das arestas naquela região.

No entanto, garantir que a malha resultante seja *Delaunay* não é trivial. Para tanto, serão utilizadas idéias estabelecidas em dois trabalhos principais, a saber [Xu et al., 1998] e [Devillers, 2002].

Seja T_k uma triangulação que é um subconjunto da triangulação de *Delaunay* T_n , ou seja, $T_k \subset T_n$, associado ao conjunto de pontos $A_k \subset A_n$. Segundo [Xu et al., 1998], uma propriedade que deriva diretamente da definição de triangulação de *Delaunay* é que T_k é uma triangulação de *Delaunay*.

Desta forma, para uma triangulação de *Delaunay* T_n associada com um conjunto de pontos A_n , suponha que um vértice, digamos $x_d \in A_n$, deve ser removido. Seja $T_{old} = S(x_d)$ o conjunto de todos os triângulos que contém o vértice x_d . Sejam A_{old} o conjunto de vértices associados a T_{old} e $A_{new} = A_{old} - x_d$. Note que x_d é ligado a todos os vértices de A_{new} via uma aresta dos triângulos em T_{old} , e que não há outro vértice em $A_n - A_{old}$ que seja ligado a x_d desta maneira. Considere ainda que T_{new} é uma triangulação de *Delaunay* gerada a partir dos vértices de A_{new} , como na figura 6.11.

O processo de remoção de um vértice consiste em primeiro remover T_{old} de T_n (veja figura 6.10) e depois adicionar T_{new} a T_n (no “buraco” deixado por T_{old}) formando uma nova triangulação T (figura 6.12). A prova de que T é uma triangulação de *Delaunay* vem da demonstração do próximo teorema, que pode ser encontrada em [Xu et al., 1998]:

Figura 6.10: Triangulação inicial T_n e triangulação resultante da remoção de T_{old} de T_n .

Figura 6.11: Triangulações T_{old} a partir de A_{old} e T_{new} a partir de A_{new} .

Figura 6.12: Nova triangulação obtida da inserção de T_{new} no lugar de T_{old} .

Teorema 6.3 *A triangulação $T = T_m - T_{old} + T_{new}$ é uma triangulação de Delaunay para o conjunto de vértices $A = A_m - x_d$.*

O teorema acima é válido, desde que a triangulação T_{new} seja uma triangulação de Delaunay, o que consiste no segundo problema a ser resolvido para se implementar a remoção de pontos em uma malha Delaunay. Segundo [Devillers, 2002], existem diversos métodos para se garantir tal propriedade, no entanto, o algoritmo desenvolvido por este mesmo autor se mostrou o mais eficiente. Antes de apresentar tal algoritmo, considere as seguintes definições:

Definição 2 (Orelha) *Considere um polígono $P = \{x_0, x_1, \dots, x_k = x_0\}$. Uma orelha do polígono P é um triângulo formado pelos vértices (x_i, x_{i+1}, x_{i+2}) desde que o segmento $x_i x_{i+2}$ esteja no interior de P e que não intercepte sua fronteira.*

Definição 3 (Power) *Seja C um círculo de centro em um ponto p e raio r . O “power” de um ponto x , denotado por $\text{power}(x, C)$ é dado por*

$$\text{power}(x, C) = \|x - p\|^2 - r^2 \quad (6.10)$$

Com estas definições em mãos, é possível escrever o teorema que é a base do algoritmo proposto em [Devillers, 2002] para a triangulação de um buraco numa malha Delaunay:

Teorema 6.4 *Considere um polígono $P = \{x_0, x_1, \dots, x_{k-1}, x_k = x_0\}$ e um ponto x_d tal que as arestas $x_i x_{i+1}$ pertençam à triangulação de Delaunay de $\{x_0, x_1, \dots, x_{k-1}, x_d\}$. Considere ainda C o circuncírculo de uma orelha (x_i, x_{i+1}, x_{i+2}) . Se $|\text{power}(x_d, C)|$ é mínimo dentre todas as orelhas de P , então x_i, x_{i+2} é uma aresta da triangulação de Delaunay de $\{x_0, x_1, \dots, x_{k-1}\}$.*

Assim, a triangulação de um buraco em uma malha Delaunay pode ser implementada de maneira fácil, mantendo uma lista de prioridade das orelhas do polígono que define o buraco, ordenadas pelo seu power. O algoritmo pode ser resumido como segue:

1. Dados um polígono que define um buraco na triangulação de Delaunay e o vértice x_d a ser removido, estabeleça uma lista de prioridades das orelhas deste polígono, ordenadas pelos valores de $|\text{power}(x_d, C)|$;
2. Insira na triangulação a orelha de maior prioridade, ou seja, a que possui o power mínimo;

3. Atualize as fronteiras do buraco. Se ainda existe buraco na triangulação de *Delaunay*, retorne ao passo 1;

Note porém que, como restrição inicial, os algoritmos descritos acima necessitam que a malha inicial seja uma triangulação de *Delaunay*. No entanto, no caso de malhas dinâmicas, isto não é garantido. Uma forma de se aproveitar este conceito é garantir que, antes da remoção, a estrela do vértice a ser removido seja localmente *Delaunay*, através da aplicação do teorema 6.2. Como a remoção afeta apenas a estrela do vértice removido, este procedimento garante a validade dos algoritmos propostos acima.

6.3 Refinamento adaptativo

O refinamento adaptativo é uma técnica que permite melhorar a precisão das aproximações, através da subdivisão ou redistribuição dos elementos que discretizam o domínio, ou do aumento do grau dos polinômios interpoladores utilizados nas aproximações. Isto de fato aumenta a precisão do método de elementos finitos, uma vez que sua ordem de convergência é $\mathcal{O}(h^{p+1})$, onde h representa o tamanho característico dos elementos e p o grau dos polinômios de interpolação. Existem várias técnicas de refinamento, normalmente aplicadas a malhas estáticas:

- **h -refinamento:** A mesma classe de elementos é utilizada em todo o domínio, mas há mudança no tamanho. Em algumas localizações são feitos menores, em outras maiores, para obter precisão com o máximo de economia. Entre os tipos de h -refinamento, pode-se destacar:
 - Enriquecimento de malha: os elementos são alterados localmente, acrescentando ou removendo vértices e arestas onde necessário;
 - Remalhamento: a malha é totalmente refeita, considerando-se os novos tamanhos dos elementos;
 - r -refinamento: o número de vértices continua constante, mas a posição é alterada de modo a obter uma distribuição que minimize o erro;
- **p -refinamento:** O tamanho dos elementos é mantido, mas altera-se o grau dos polinômios de interpolação utilizados na aproximação. Isto pode ser feito em toda a malha, ou localmente de forma hierárquica;
- **hp -refinamento:** Trata-se de um híbrido entre os dois métodos, alterando tanto o tamanho h dos elementos, quanto o grau p dos polinômios de interpolação;

Tais técnicas podem ainda serem aplicadas para malhas dinâmicas com suas devidas adaptações. Neste trabalho, aproveitando-se as estratégias de controle de malha discutidas acima, calcula-se uma distribuição de tamanhos $\ell(\mathbf{x})$ tal que pontos são inseridos, removidos e/ou posicionados de forma a respeitar esta distribuição. A seguir é descrita a determinação desta distribuição.

6.3.1 Refinamento para malhas dinâmicas

Dado um espaçamento de malha ℓ_D , é desejável que a distribuição de pontos na malha seja feita respeitando-se este espaçamento, e adicionalmente resulte em uma transição suave entre os espaçamentos máximos e mínimos, garantido uma melhor qualidade na aproximação numérica. Contudo, obter um espaçamento pré-estabelecido suave nem sempre é prático. Neste trabalho, considera-se a solução de uma equação de *Helmholtz* para obter uma distribuição de pontos suave, a partir da distribuição desejada ℓ_D . Matematicamente, resolve-se ℓ da equação

$$(1 - k\nabla^2)\ell = \ell_D, \quad (6.11)$$

onde k é uma constante que determina a suavização da distribuição ℓ_D . Note que, da equação acima,

$$\nabla^2\ell = \frac{1}{k}(\ell - \ell_D), \quad (6.12)$$

e portanto, se k é muito grande, o termo fonte é aproximadamente zero, resultando em uma equação de *Laplace* ($\nabla^2\ell = 0$), cuja solução é muito suave, amortecendo qualquer mudança brusca de ℓ_D . Em contrapartida, se k é pequeno, o termo *Laplaciano* é praticamente nulo, obtendo-se uma distribuição ℓ muito próxima da distribuição original ℓ_D . Um exemplo do efeito da variação desta constante k pode ser visto na figura 6.13.

A equação (6.11) pode ser discretizada através do método de elementos finitos, seguindo os procedimentos do capítulo 3, para a distribuição dos tamanhos nas arestas, e a solução de (6.11) pode ser utilizada para determinar os limitantes máximo e mínimo (como um fator de tolerância para ℓ) para a subdivisão ou remoção das arestas da triangulação.

6.3.2 Estimativa do espaçamento da malha

A determinação da distribuição desejada dos pontos (ℓ_D) pode ser feita através do conhecimento *a priori* das regiões do escoamento que necessitam ser refinadas, ou utilizando algum critério mais dinâmico, como uma estimativa de erro, de forma ℓ_D seja proporcionalmente menor nas regiões onde o erro estimado seja grande.

Figura 6.13: Exemplo das soluções obtidas para a equação de *Helmholtz* unidimensional, para diferentes valores de k . A linha contínua representa o termo fonte ℓ_D , a linha tracejada foi obtida com $k = 0.0028$ e a linha pontilhada com $k = 0.0443$.

Existem várias maneiras de se estimar este erro, e algumas delas podem ser encontradas em [Löhner, 1995]. Considere h um tamanho característico de aresta, a aproximação via série de *Taylor* de um campo escalar $f(x + h_x, y + h_y)$ em torno de (x, y) , é dada por

$$f(x + h_x, y + h_y) \approx f(x, y) + h_x \frac{\partial f}{\partial x} + h_y \frac{\partial f}{\partial y} + \frac{h_x^2}{2} \frac{\partial^2 f}{\partial x^2} + h_x h_y \frac{\partial^2 f}{\partial x \partial y} + \frac{h_y^2}{2} \frac{\partial^2 f}{\partial y^2} + O(h^3) \quad (6.13)$$

Como a aproximação dos operadores diferenciais por elementos finitos utilizada neste trabalho não tem termos de segunda ordem (observe que as funções de interpolação para o mini-elemento dadas no capítulo 3 possuem apenas termos lineares e cúbicos), os erros mais significativos estão nos termos de derivadas de segunda ordem. Assim, utilizando a definição das matrizes *Hessiana* dos campos $u(\mathbf{x}, t)$ e $v(\mathbf{x}, t)$, dadas por

$$\mathbf{H}(u)|_{(\mathbf{x}, t)} = \begin{bmatrix} \frac{\partial^2 u}{\partial x^2} & \frac{\partial^2 u}{\partial x \partial y} \\ \frac{\partial^2 u}{\partial x \partial y} & \frac{\partial^2 u}{\partial y^2} \end{bmatrix}, \quad \mathbf{H}(v)|_{(\mathbf{x}, t)} = \begin{bmatrix} \frac{\partial^2 v}{\partial x^2} & \frac{\partial^2 v}{\partial x \partial y} \\ \frac{\partial^2 v}{\partial x \partial y} & \frac{\partial^2 v}{\partial y^2} \end{bmatrix} \quad (6.14)$$

pode-se estimar o erro para as aproximações dos campos escalares $u(\mathbf{x}, t)$ e $v(\mathbf{x}, t)$ como sendo

$$E(u) = \frac{1}{2} \mathbf{h}^T \mathbf{H}(u) \mathbf{h}, \quad E(v) = \frac{1}{2} \mathbf{h}^T \mathbf{H}(v) \mathbf{h} \quad (6.15)$$

onde $\mathbf{h} = [h_x, h_y]^T$ são os tamanhos característicos das arestas. Note que, considerando um espaçamento isotrópico, $h = h_x = h_y$, tem-se

$$|E(u)| = \left| \frac{1}{2} \mathbf{h}^T \mathbf{H}(u) \mathbf{h} \right| \leq \frac{h^2}{2} \left(\left| \frac{\partial^2 u}{\partial x^2} \right| + 2 \left| \frac{\partial^2 u}{\partial x \partial y} \right| + \left| \frac{\partial^2 u}{\partial y^2} \right| \right), \quad (6.16)$$

$$|E(v)| = \left| \frac{1}{2} \mathbf{h}^T \mathbf{H}(v) \mathbf{h} \right| \leq \frac{h^2}{2} \left(\left| \frac{\partial^2 v}{\partial x^2} \right| + 2 \left| \frac{\partial^2 v}{\partial x \partial y} \right| + \left| \frac{\partial^2 v}{\partial y^2} \right| \right). \quad (6.17)$$

Considerando-se uma tolerância ε , é desejável que o erro associado às aproximações de u e v seja menor que ε , ou seja,

$$\|\mathbf{E}(\mathbf{u})\| \leq |E(u)| + |E(v)| \leq \varepsilon \quad (6.18)$$

onde $\mathbf{E}(\mathbf{u}) = [E(u), E(v)]^T$. Note que, tomando-se

$$h \leq \sqrt{\frac{2\varepsilon}{H(u) + H(v)}} \quad (6.19)$$

onde $H(u) = \left| \frac{\partial^2 u}{\partial x^2} \right| + 2 \left| \frac{\partial^2 u}{\partial x \partial y} \right| + \left| \frac{\partial^2 u}{\partial y^2} \right|$ e $H(v) = \left| \frac{\partial^2 v}{\partial x^2} \right| + 2 \left| \frac{\partial^2 v}{\partial x \partial y} \right| + \left| \frac{\partial^2 v}{\partial y^2} \right|$, a inequação (6.18) é satisfeita. Assim, determinando-se a distribuição ℓ_D de modo que o tamanho das arestas respeitem a condição acima, é possível gerar uma solução cujo erro estimado pela *Hessiana* seja menor que um valor de tolerância ε .

Capítulo 7

Validação e resultados numéricos

Neste capítulo são apresentados os resultados numéricos obtidos a partir do programa `UMeshFlow`, que é a implementação em linguagem orientada a objeto `C++`, desenvolvida neste trabalho para o caso bidimensional, de todos os métodos vistos até aqui. Primeiramente serão apresentadas validações obtidas de escoamentos monofásicos confinados com e sem refinamento adaptativo, e em seguida algumas validações para escoamentos multifásicos. Finalmente são apresentados resultados mostrando a captura da inversão da força de sustentação (*lift*) de bolhas ascendendo em um escoamento cisalhante, e a comparação com resultados obtidos com outra técnica em 3D, mostrando o potencial do método desenvolvido.

7.1 Validações para escoamentos monofásicos confinados

São apresentados nesta seção exemplos de simulações de escoamentos monofásicos confinados sem a influência da gravidade. Primeiramente é feita uma comparação entre os resultados obtidos para uma cavidade com e sem refinamento adaptativo, seguida da validação do escoamento em um canal. Por fim, são apresentadas validações para escoamentos em um degrau e em torno de um cilindro circular.

7.1.1 Escoamento em uma cavidade

Para simular o escoamento circulatório em uma cavidade, considere uma caixa quadrada cheia de fluido, onde são impostas condições de contorno de velocidades nulas, exceto

Figura 7.1: Domínio para o escoamento em uma cavidade, com valores adimensionais.

Figura 7.2: Malhas estáticas utilizadas na validação, com número de elementos igual a (a) 236, (b) 932 e (c) 3690.

na parte superior, onde é imposta uma velocidade tangencial adimensional igual a 1. O domínio pode ser visto na figura 7.1.

Este escoamento foi simulado para $Re = 10$ em três malhas estáticas, com espaçamento $h = 0.1$, $h = 0.05$ e $h = 0.025$, resultando em malhas com respectivamente 236, 932 e 3690 elementos, ilustradas na figura 7.2. Adicionalmente, foram simulados casos em malhas com refinamento adaptativo baseado na *Hessiana* do campo de velocidades, e a velocidade da malha $\hat{u}$ calculada com $\beta_1 = 0$ e $\beta_2 = 1$ obtendo-se uma suavização puramente elástica. As simulações foram iniciadas com uma malha relativamente grossa ($h = 0.2$) e refinadas utilizando quatro tolerâncias ε (definida na equação (6.18)) diferentes. Para os valores de $\varepsilon = 0.2$, $\varepsilon = 0.1$, $\varepsilon = 0.05$ e $\varepsilon = 0.025$, obtém-se malhas finais com 463, 893, 1702 e 3374 elementos respectivamente, ilustradas na figura 7.3

A figura 7.4 ilustra a malha adaptativa obtida com $\varepsilon = 0.05$, e a escala de cores dos campos de velocidade u e v . Note que o refinamento se concentra nas áreas de maior gradiente, sendo a malha especialmente refinada próximo às singularidades devidas à condição de contorno.

Figura 7.3: Malhas adaptativas finais obtidas com refinamento segundo a *Hessiana* das velocidades, com número de elementos igual a (a) 463, (b) 893, (c) 1702 e (d) 3374.

Figura 7.4: Escala de cores dos campos de velocidade nas direções x e y . O refinamento foi realizado nas áreas onde há maior gradiente.

Malhas	u_{min}	v_{max}	v_{min}	Ne	CPU
$h = 0.1$	-0.1934	0.3669	-0.3855	236	0.8 s
$h = 0.05$	-0.1964	0.3705	-0.3824	932	10.0 s
$h = 0.025$	-0.2033	0.3674	-0.3731	3690	164.8 s
$\varepsilon = 0.2$	-0.2072	0.3669	-0.3765	463	12.8 s
$\varepsilon = 0.1$	-0.2112	0.3648	-0.3828	893	54.4 s
$\varepsilon = 0.05$	-0.2104	0.3676	-0.3861	1702	196.3 s
$\varepsilon = 0.025$	-0.2061	0.3740	-0.3888	3374	2032.7 s
<i>Meshless</i>	-0.2041	0.3482	-0.3563	≈ 1000	N/D
<i>Referência</i>	-0.2081	0.3635	-0.3959	10000	N/D

Tabela 7.1: Comparação entre os valores máximo e mínimo das velocidades u e v em diversas malhas.

Uma comparação mais quantitativa, utilizando os valores máximo e mínimo das velocidades u e v , pode ser vista na tabela 7.1. Estes são comparados aos esquemas *meshless* [Mangiavacchi et al., 2002], e com um resultado de referência produzido por um método de diferenças finitas em uma malha cartesiana com espaçamento uniforme $h = 0.01$, reportado em [Mangiavacchi et al., 2002].

Pode-se notar nos dados da tabela 7.1, que em geral os resultados obtidos pela discretização por elementos finitos estão próximos do valor de referência. Note ainda que, utilizando a técnica de refinamento adaptativo com $\varepsilon = 0.1$, por exemplo, obteve-se uma malha de 893 elementos, com resultados bem mais próximos aos valores de referência do que o obtido com malha uniforme $h = 0.05$, com praticamente o mesmo número de elementos. O tempo de CPU reportado na tabela 7.1 refere-se ao tempo em segundos que cada simulação levou para atingir o tempo adimensional $t = 1$. Nota-se nesta tabela que a técnica de refinamento aumenta consideravelmente o tempo de computação, o que se explica pelo tamanho dos elementos nas malhas adaptativas ser muito menor que o das malhas estáticas (por exemplo, obteve-se $h = 0.0003$ para a malha adaptativa com $\varepsilon = 0.025$), e devido à restrição da condição CFL, o passo no tempo é proporcionalmente muito menor.

7.1.2 Escoamento de Hagen-Poiseuille

Este exemplo consiste em simular o escoamento de fluido através de um canal bidimensional. As condições de entrada e saída de fluido são aplicadas nas fronteiras laterais, enquanto que nas outras fronteiras são impostas velocidades nulas. Considerou-se para este exemplo um canal com dimensões $3L \times L$, sendo L o tamanho da fronteira de entrada de fluido.

Figura 7.5: Malha adaptativa resultante e campo de velocidade na direção x .

Este tipo de escoamento é um dos poucos em que as equações de *Navier-Stokes* admitem solução analítica [Fortuna, 2000], neste caso, dada por:

$$u(y) = -\frac{1}{2\nu} \frac{\partial p}{\partial x} (yL - y^2) \quad (7.1)$$

onde $\nu = \mu/\rho$ é a viscosidade cinemática, a derivada da pressão é dada por

$$\frac{\partial p}{\partial x} = -12 \frac{\nu Q}{L^3} \quad (7.2)$$

e Q é a vazão, definida por

$$Q = \int_0^L u(y) dy. \quad (7.3)$$

A pressão é uniforme ao longo de qualquer secção reta do canal, pois a velocidade normal à direção x é constante igual a zero. Considerando uma vazão constante $Q = L$, a derivada da pressão fica

$$\frac{\partial p}{\partial x} = -12 \frac{\nu}{L^2} \quad (7.4)$$

e finalmente a solução é dada por

$$u(y) = \frac{6}{L^2} (yL - y^2) \quad (7.5)$$

Este caso foi simulado para a malha mostrada na figura 7.5, com $h_{min} = 0.05$, obtida através de refinamento adaptativo utilizando a *Hessiana* das velocidades a partir de uma malha uniforme com espaçamento $h = 0.1$. A velocidade de entrada foi imposta como sendo um perfil parabólico gerado pela equação (7.5). A comparação entre as soluções aproximada e analítica pode ser vista na figura 7.6. Para este caso, o erro relativo obtido é $\|u - u(y)\|/\|u(y)\| = 0.0025$.

Figura 7.6: Comparação entre os perfis das velocidades aproximada e exata.

7.1.3 Escoamento em um degrau

Este teste consiste em uma expansão de fluido sobre um degrau, gerando uma zona de recirculação abaixo da camada limite do escoamento. A geometria, dimensões e condições de entrada e saída podem ser vistos na figura 7.7.

A simulação foi feita utilizando-se uma malha estática com 2407 elementos e $h_{min} = 0.05$, mostrada na figura 7.8. Considerando-se $Re = 50$, o tamanho da zona de recirculação, onde a camada limite se religa ao contorno do escoamento, é conhecido. Neste caso, foram utilizados $L = 1$ e $U = 1$ como valores de referência para comprimentos e para velocidade, sendo o tamanho do degrau igual a 1. Para esta configuração, segundo [Armaly et al., 1983], o tamanho da zona de recirculação é dado por $x_1/L = 3.0$, enquanto para esta simulação obteve-se $x_1/L = 2.95$, um erro de 1.17%. As linhas de corrente obtidas para esta simulação podem ser vistas na figura 7.9.

Figura 7.7: Domínio para o escoamento em um degrau, com valores adimensionais.

Figura 7.8: Malha utilizada na simulação do degrau.

Figura 7.9: Linhas de corrente após a simulação atingir o estado estacionário, mostrando a circulação abaixo da camada limite.

7.1.4 Desprendimento de vórtices

O último caso de validação para escoamentos monofásicos consiste na simulação do escoamento bidimensional em torno de um cilindro circular. Para números de *Reynolds* na faixa $40 < Re < 10000$, o escoamento em torno do cilindro causa o desprendimento de vórtices, sendo este laminar até $Re = 300$. Um esquema mostrando as condições de contorno para este escoamento pode ser visto na figura 7.10. Nas paredes superior e inferior foram impostas condições de escorregamento (*free-slip*).

A simulação foi feita sobre uma malha fixa com 4140 elementos e $h_{min} = 0.1$, mostrada

Figura 7.10: Esquema do escoamento em torno de um cilindro, com valores dimensionais.

Figura 7.11: Malha utilizada na simulação do escoamento em torno de um cilindro.

na figura 7.11. O número de *Reynolds* utilizado foi $Re = 100$, suficiente para o desprendimento dos vórtices. A figura 7.12 mostra a vorticidade do escoamento em vários tempos diferentes.

A validação para este caso pode ser feita comparando-se o número de *Strouhal* com valores observados experimentalmente, sendo este definido por

$$St = \frac{\omega D}{U}, \quad (7.6)$$

onde U é a velocidade característica e ω é a frequência de desprendimento dos vórtices, que para este valor de *Reynolds* pode ser escrito somente em função da velocidade e do diâmetro característico D . O número de *Strouhal* obtido na simulação foi $St = 0.195$, enquanto que os observados experimentalmente estão em uma faixa $0.162 < St < 0.174$. A discrepância entre os valores obtidos numericamente e experimentalmente pode ser explicada pela proximidade das paredes inferior e superior, aumentando a frequência de oscilação. Um gráfico da oscilação da velocidade v pode ser visto na figura 7.13.

7.2 Validações para escoamentos multifásicos

Nesta seção serão apresentadas algumas validações para simulações de escoamentos multifásicos, com e sem a influência da gravidade. Primeiramente é feita uma validação para o cálculo da tensão superficial através da simulação de uma bolha estática e de uma bolha oscilante. Um teste de convergência é feito para o caso da ascensão de uma bolha, e comparado com resultados obtidos de outros trabalhos. Finalmente, algumas comparações são feitas para ascensão de bolhas utilizando-se estratégias *Lagrangeana* e *ALE* com suavização de malha.

Figura 7.12: Evolução temporal da vorticidade no escoamento.

Figura 7.13: Oscilação da componente v da velocidade, em $\mathbf{x} = (3.1, 5)$.

Figura 7.14: Domínio para a simulação de uma bolha estática.

7.2.1 Bolha estática

Para validar o cálculo da tensão superficial e da distribuição da força agindo sobre a interface, é feita a simulação de uma bolha estática imersa em um fluido, sem a influência da gravidade. Na ausência de viscosidade e de forças externas, os efeitos de tensão superficial fazem com que uma bolha permaneça esférica. A expressão analítica para o salto de pressão na interface é dada pela fórmula de *Laplace* [Brackbill et al., 1992], que para o caso de um cilindro infinito imerso em um fluido, é dada por

$$\Delta p = \sigma \kappa = \frac{\sigma}{R} \quad (7.7)$$

onde R é o raio de curvatura da bolha. A simulação foi feita para o domínio dado na figura 7.14, tomando-se $R = 0.5$, $\rho_b = 1$ e $\rho_f = 0.5$ como sendo as massas específicas

Malhas	Ne	Δp	E	$\max\{ u , v \}$
$h = 0.2$	248	1.97847	1.088%	1.638×10^{-4}
$h = 0.1$	946	1.99485	0.258%	2.015×10^{-5}
$h = 0.05$	3782	1.99872	0.064%	4.535×10^{-6}
$h = 0.025$	14972	1.99969	0.015%	4.252×10^{-6}

Tabela 7.2: Comparação entre os saltos de pressão e dos erros cometidos nas diversas malhas. Aqui, Ne é o número de elementos em cada malha, $\Delta p = p_b - p_f$ onde p_b é a pressão interna e p_f a pressão externa à bolha, e E a porcentagem do erro cometido na aproximação

da bolha e do fluido respectivamente, e coeficiente de tensão superficial $\sigma = 1$. Assim, a expressão analítica para o salto de pressão na interface é $\Delta p = 2$.

Esta simulação foi feita em diversas malhas uniformes, com espaçamentos $h = 0.2$, $h = 0.1$, $h = 0.05$ e $h = 0.025$. As figuras 7.15 e 7.16 mostra o salto de pressão obtido com o esquema de aplicação de força distribuída, vista no capítulo 5, para diversos espaçamentos de malha. Pode-se notar que a pressão é constante em cada fluido, salvo na interface, onde a pressão assume um valor intermediário. Isso mostra a precisão e consistência do esquema de distribuição da força desenvolvido neste trabalho.

Para avaliar a precisão do cálculo da curvatura, foi feito um estudo de convergência com os valores do salto de pressão para as malhas utilizadas, sendo estes mostrados na tabela 7.2. A ordem de convergência entre as malhas $h = 0.2$ e $h = 0.1$ foi $n = 2.0637$, entre as malhas $h = 0.1$ e $h = 0.05$ foi $n = 2.0084$, e entre as malhas $h = 0.05$ e $h = 0.025$ foi $n = 2.0454$, e portanto pode-se constatar ordem de convergência quadrática para este exemplo. Adicionalmente, os valores máximos das velocidades das correntes parasitas no escoamento podem ser vistos na tabela 7.2.

7.2.2 Gota oscilante

Esta validação consiste em avaliar a frequência de oscilação de uma gota de fluido imersa em outro fluido. Como visto acima, bolhas ou gotas tendem a permanecer esféricas (ou circulares no caso bidimensional), ponto de equilíbrio entre as forças de tensão interfacial. Assim, se o raio de uma bolha ou gota é inicialmente perturbado, esta oscila até atingir o equilíbrio, amortecida pela viscosidade.

Foi utilizado um domínio unitário discretizado por uma malha de 2426 elementos e espaçamento não uniforme, com $h_{min} = 0.02$. Os parâmetros da gota são diâmetro $D = 0.4$, massa específica $\rho_g = 1$ e viscosidade $\mu_g = 0.01$, sendo o diâmetro inicial da gota perturbado horizontalmente em 5%. O fluido externo possui massa específica

Figura 7.15: Gráfico da pressão na bolha estática, para $h = 0.05$.

Figura 7.16: Perfis do salto de pressão na bolha estática, para várias malhas diferentes.

Figura 7.17: Inversão no sentido da velocidade u no escoamento, caracterizando oscilação.

$\rho_f = 0.05$ e viscosidade $\mu_f = 0.001$. A malha e a interface foram transportadas utilizando-se um esquema *Lagrangeano*, ou seja, todos os pontos movendo-se com a velocidade do escoamento.

A componente u do campo de velocidade do escoamento pode ser vista na figura 7.17, onde pode-se notar a inversão no sentido da velocidade, caracterizando a oscilação. Um gráfico da evolução do diâmetro da gota no tempo pode ser visto na figura 7.18, ilustrando o amortecimento causado pela viscosidade.

A frequência da oscilação analítica é dada por [Tryggvason et al., 1998]

$$\omega_A^2 = \frac{(n^2 - n)\sigma}{(\rho_g + \rho_f)R^3} \quad (7.8)$$

onde $R = D/2$ é o raio da bolha (não perturbado), e n caracteriza o modo de oscilação, que neste caso é 2. A frequência analítica utilizando os dados deste escoamento resulta em $\omega_A = 27.38$, enquanto que a frequência observada na solução numérica foi $\omega_N = 24.32$, resultando em um erro de 9.01%. Este resultado é compatível com os obtidos em [Esmaeeli and Tryggvason, 1998], que reporta um erro de 9.3% em uma malha com 32^2 células quadradas, o que equivaleria a uma malha com 2048 triângulos.

Figura 7.18: Evolução do diâmetro da gota no tempo, ilustrando o amortecimento devido a viscosidade.

7.2.3 Ascensão de bolhas

Análise de convergência

Nesta seção é feita uma análise de convergência do método proposto neste trabalho, através da simulação da ascensão de uma bolha em um fluido. O domínio para esta simulação pode ser visto na figura 7.19. Os dados adimensionais, tomados de [Tornberg, 2000], são $M = 0.1$, $Eo = 10$, $\rho_f/\rho_b = 100$, $\mu_f/\mu_b = 2$ e $R = 0.5$. Este problema foi simulado para malhas com espaçamento uniforme $h_1 = 0.2$, $h_2 = 0.1$ e $h_3 = 0.05$, para os esquemas *Lagrangeano* e *ALE*, com parâmetros para o cálculo da velocidade da malha $\beta_1 = 1$ e $\beta_2 = 0$ no *Lagrangeano*, e $\beta_1 = 1$ e $\beta_2 = 1$ para o *ALE*. As malhas estão ilustradas na figura 7.20.

Os resultados da análise de convergência estão reportados na tabela 7.3. A norma $L1$ que aparece nesta tabela é dada por

$$\|u\|_1 = \frac{1}{N} \sum_{i=1}^N (u_i^2 + v_i^2)^{\frac{1}{2}}, \quad (7.9)$$

onde N é o número de pontos que discretizam u . Os valores obtidos com os esquemas *Lagrangeano* e *ALE* estão próximos dos valores reportados em [Tornberg, 2000]. Esta análise foi feita para os tempos $t = 0.15$ e $t = 0.3$, mas somente os dados para o tempo $t = 0.15$ são encontrados em [Tornberg, 2000]. Os resultados são compatíveis com a precisão formal dos métodos, que é de segunda ordem.

Figura 7.19: Domínio para a análise de convergência para o problema de ascensão de uma bolha.

(a) $h = 0.2$

(b) $h = 0.1$

(c) $h = 0.05$

Figura 7.20: Malhas utilizadas para o teste de convergência.

Método	$t = 0.15$		$t = 0.3$	
	$\frac{\ u_{h_1} - u_{h_2}\ _1}{\ u_{h_2} - u_{h_3}\ _1}$	n	$\frac{\ u_{h_1} - u_{h_2}\ _1}{\ u_{h_2} - u_{h_3}\ _1}$	n
<i>Lagrangeano</i>	3.83	1.74	3.98	1.88
<i>ALE</i>	3.34	1.60	3.77	1.79
<i>Segment projection</i>	3.78	1.92	N/D	N/D
<i>Level-set</i>	3.20	1.68	N/D	N/D
<i>Front-tracking</i>	3.48	1.80	N/D	N/D

Tabela 7.3: Análise da ordem de convergência n para $t = 0.15$ e $t = 0.3$, e comparação com os métodos *Segment projection*, *Level-set* e *Front-tracking* com malhas estáticas. Os dados para estes métodos foram extraídos de [Törnberg, 2000].

Comparação entre as técnicas *Lagrangeana* e *ALE*

Nesta seção é feita uma comparação entre os esquemas *Lagrangeano* e *ALE* para o caso de ascensão de bolhas, variando-se o parâmetro β_2 definido em (6.1). Considere os mesmos parâmetros adimensionais do caso descrito acima, porém com domínio $4R \times 12R$, onde $R = 0.5$ é o raio da bolha. A malha inicial é discretizada por 1306 elementos, com $h_{min} = 0.08$. Os esquemas adotados para esta comparação são o *Lagrangeano* e *ALE*, para os coeficientes $\beta_1 = 1$ fixo, e tomando-se $\beta_2 = 1$, $\beta_2 = 0.5$ e $\beta_2 = 0.1$ na determinação da velocidade final da malha (6.1).

Para todas as simulações, a distribuição do tamanho das arestas foi feita tomando $k \gg 1$ na equação (6.11), utilizando como condições de contorno para esta equação $h = 0.3$ nas fronteiras rígidas e $h = 0.08$ na interface, produzindo como resultado uma distribuição suave entre estes valores.

Uma comparação entre as malhas obtidas variando-se o parâmetro β_2 pode ser vista na figura 7.21. Pode-se notar o efeito elástico agindo na malha conforme β_2 aumenta. Este efeito tem consequência direta no comportamento da malha: no esquema *Lagrangeano*, como não há suavização de malha, é necessário um maior número de inserções e remoções de vértices para manter a qualidade da mesma. Por outro lado, se a velocidade da malha é predominantemente elástica, o movimento da interface tende a forçar inserções e remoções mais bruscas, de vários vértices ao mesmo tempo.

A figura 7.22 mostra o número de *Reynolds* da bolha e sua área em função do tempo adimensional. Note que, para o caso onde $\beta_2 = 1$, aparecem oscilações na velocidade de ascensão, causadas pela inserção e remoção de vários vértices ao mesmo tempo devido à

Figura 7.21: Malha inicial (a) e comparação entre as malhas obtidas no tempo $t = 12.5$ para os diversos parâmetros: (b) $\beta_2 = 0$, (c) $\beta_2 = 0.1$, (d) $\beta_2 = 0.5$, (e) $\beta_2 = 1.0$.

elasticidade da malha. É possível notar a diminuição deste efeito com a diminuição no parâmetro β_2 , sendo o resultado obtido com $\beta_2 = 0$ bem mais suave, devido às inserções e remoções de vértices serem mais distribuídas ao longo da simulação e do domínio. Contudo, nesta mesma figura, é possível observar a diminuição da área da bolha, sendo esta de 4.8% para $\beta_2 = 0$ contra 0.9% para $\beta_2 = 1$. Este fato se explica pela necessidade de se inserir e remover vértices da interface (que são operações que não conservam massa) mais frequentemente, quando $\beta_2 = 0$, uma vez que os pontos se movimentam de cima para baixo ao longo da bolha, como descrito no capítulo 6.

O efeito do aumento de operações na interface pode também ser visto na evolução dos diâmetros horizontal e vertical da bolha, ilustrada na figura 7.23. É possível observar uma oscilação no diâmetro horizontal da bolha para $\beta_2 = 0$ e $\beta_2 = 0.1$, causado pela frequente inserção e remoção de pontos na interface, o que não acontece para valores maiores de β_2 . Uma comparação entre os perfis da interface no tempo adimensional $t = 12.5$ pode ser vista na figura 7.24.

Pode-se concluir portanto que a melhor escolha para β_1 e β_2 depende do problema em questão. Enquanto o esquema *Lagrangeano* ficou prejudicado pela perda de massa da bolha, causada pelo grande número de operações de controle da interface, este pode se beneficiar de simulações que não envolvam bolhas. Já o método *ALE*, prejudicado neste

Figura 7.22: Número de *Reynolds* da bolha e conservação de massa para os casos simulados.

Figura 7.23: Comparação entre a evolução no tempo dos diâmetros horizontal (D_x) e vertical (D_y).

Figura 7.24: Comparação entre os perfis das bolhas obtidas com $\beta_2 = 0$ (preta); $\beta_2 = 0.1$ (verde); $\beta_2 = 0.5$ (azul); $\beta_2 = 1.0$ (vermelha).

exemplo pela oscilação na velocidade de ascensão, é muito bem sucedido em situações com referencial se movendo com a bolha, como é o caso dos escoamentos mostrados no final deste capítulo, onde o número de operações de inserção e remoção é reduzido.

7.3 Escoamento com mudança topológica na interface

Para ilustrar o método de captura de mudanças topológicas na interface descrito no capítulo 5, foi simulada a coalescência de duas bolhas do mesmo fluido ascendendo em outro fluido. O domínio e os parâmetros adimensionais utilizados foram os mesmos do exemplo da seção anterior. As bolhas foram posicionadas uma embaixo da outra, a uma distância de $0.1D$, onde D é o diâmetro de ambas.

A evolução no tempo da coalescência das bolhas é ilustrada na figura 7.25. Pode-se observar o efeito de tensão interfacial agindo imediatamente após a coalescência, mostrando um comportamento compatível com outras simulações deste mesmo problema (ver por exemplo em [Sousa et al., 2004] e [Sousa and Mangiavacchi, 2004]).

7.4 Inversão da força de sustentação em escoamentos de bolhas

Esta seção apresenta a aplicação do método apresentado neste trabalho para a investigação de um fenômeno presente em processos industriais, e de grande interesse da comunidade científica atualmente. Trata-se da inversão da força de sustentação agindo em bolhas ascendendo verticalmente em escoamentos cisalhantes. Este fenômeno é causa de problemas de bombeamento de óleos pesados de águas profundas onde é empregada uma técnica de injeção de gás para diminuição do peso da coluna de petróleo. Nestes escoamentos, sob certas circunstâncias, bolhas tendem a se concentrar no centro do escoamento vertical, formando longas golfadas (*slugs*) de ar, prejudicando o bombeamento.

Apesar da importância de tal fenômeno, e do grande número de trabalhos experimentais publicados sobre este tema [Takemura and Magnaudet, 2003, Magnaudet et al., 2003, Sakata et al., 2004, Tomiyama, 2002], há poucos trabalhos que apresentam simulações numéricas para o problema [Ervin and Tryggvason, 1997, Deen et al., 2004]. Nesta seção, são apresentados vários resultados numéricos sobre ascensão de bolhas em escoamentos verticais cisalhantes, obtidos utilizando-se um método de *front-capturing/front-tracking* implementado em uma malha cartesiana 3D utilizando-se técnicas de diferenças finitas [Sousa et al., 2004]. Estes resultados são comparados com dados experimentais extraídos de [Tomiyama, 2002].

Figura 7.25: Coalescência de duas bolhas ascendendo em um fluido.

Figura 7.26: Esquema de representação do domínio referencial: W_b é a componente vertical da velocidade da bolha, W_0 é a velocidade de referência no contorno, L é o tamanho do domínio, D_b é o diâmetro da bolha, e $\omega = \partial v / \partial x = W_0 / L$ é a intensidade do escoamento cisalhante.

Adicionalmente, para ilustrar o poder do método desenvolvido neste trabalho, os mesmos problemas foram investigados e comparados com os resultados numéricos bidimensionais reportados em [Ervin and Tryggvason, 1997].

7.4.1 Soluções obtidas com método *front-tracking/front-capturing* 3D

Para que a solução numérica deste problema seja comparável com os dados experimentais, é necessário que a bolha suba a uma distância equivalente a muitos diâmetros da mesma. Simulações com referenciais parados são proibitivos, principalmente em métodos que utilizem malhas uniformes para discretizar as equações. Neste sentido, foi considerado para esta simulação um domínio que se move na mesma velocidade da bolha, de forma que, verticalmente, a bolha permaneça parada em seu referencial. A figura 7.26 representa o domínio utilizado.

O método *front-tracking/front-capturing* aplicado aqui foi implementado utilizando-se discretizações via diferenças finitas, e a interface é representada por uma malha *Lagrangiana* 2D movendo-se sobre uma malha *Euleriana* 3D estruturada uniforme. As forças na interface são aplicadas através do cálculo de uma função de *Heaviside* discreta, baseada na classificação das células na malha 3D. Mais detalhes sobre este método podem ser encontrados em [Sousa et al., 2002a, Sousa et al., 2002b, Sousa and Mangiavacchi, 2002, Sousa et al., 2004].

As simulações foram realizadas com os parâmetros encontrados em [Tomiya, 2002]: $\rho_f = 1166 \text{ kg/m}^3$, $\mu_f = 0.022 \text{ kg/m} \cdot \text{s}$ e $\sigma = 0.061 \text{ N/m}$ são respectivamente a massa específica, viscosidade e coeficiente de tensão superficial para a fase líquida. Foram con-

Figura 7.27: Número de *Reynolds* relativo das bolhas pequena e grande, respectivamente.

sideradas bolhas de diâmetros, $D = 3.52 \text{ mm}$ e $D = 5.53 \text{ mm}$, daqui para frente referenciadas como bolha pequena e bolha grande. A intensidade do escoamento cisalhante utilizada foi $\omega = 6.2 \text{ s}^{-1}$.

Os parâmetros adimensionais que representam este problema são o número de *Morton* definido em (2.50), e o número de *Eötvös*, dado por (2.48) utilizando-se $\rho_0 = \rho_f - \rho_b$, onde ρ_b é a massa específica da fase gasosa. Para as simulações consideradas aqui, tem-se $M = 10^{-5}$ para as duas bolhas, sendo o número de *Eötvös* da bolha menor calculado como $Eo = 2.3$ e da bolha maior como $Eo = 5.7$.

A figura 7.27 mostra o número de *Reynolds* relativo para ambas as bolhas, dado por

$$Re = \frac{\rho_f |\mathbf{u}_b| D}{\mu_f} \quad (7.10)$$

onde $\mathbf{u}_b$ é a velocidade relativa do centro de massa da bolha, obtida subtraindo-se a velocidade do campo de escoamento cisalhante relativa à posição horizontal da bolha, da velocidade total da mesma.

As linhas de corrente do campo de velocidade, no referencial da bolha, podem ser vistas em 7.28, onde é detectada uma assimetria na esteira da bolha pequena, e recirculações na esteira da bolha grande. É importante frisar que este tipo de comportamento do campo de velocidades é muito difícil de se detectar utilizando-se apenas dados experimentais, o que demonstra a importância da simulação numérica para o entendimento dos fenômenos físicos relevantes.

Uma comparação qualitativa entre os formatos da bolha, obtidos experimentalmente [Tomiya, 2002] e numericamente, pode ser vista na figura 7.29. Nota-se nesta figura que a simulação numérica conseguiu prever a deformação das duas bolhas.

Figura 7.28: Corte bidimensional do domínio, mostrando as linhas de corrente para as bolhas pequena e grande.

Figura 7.29: Comparação entre os formatos das bolhas: (a) pequena, experimental; (b) pequena, numérica; (c) grande, experimental; (d) grande, numérica.

Figura 7.30: Trajetórias para as bolhas pequena e grande, com e sem escoamento cisalhante. As linhas representam as soluções numéricas.

A comparação entre as trajetórias das bolhas pode ser vista em 7.30. Nesta figura, as trajetórias das bolhas ascendendo sem um escoamento cisalhante são também mostrados e comparados com os dados experimentais de [Tomiya, 2002]. Enquanto que para as bolhas grandes obteve-se uma boa concordância, os resultados para bolhas pequenas não coincidem com os dados experimentais.

Uma comparação quantitativa pode ser vista na figura 7.31, onde a evolução no tempo do coeficiente de força de sustentação transversal é mostrada. A força de sustentação transversal neste caso é dada por

$$\mathbf{F}_T = -C_T V \rho_f (\mathbf{u}_b - \mathbf{u}_f) \times (\nabla \times \mathbf{u}_f) \quad (7.11)$$

onde C_T é o coeficiente da força de sustentação transversal, V é o volume da bolha, ρ_f é a massa específica do líquido e $\mathbf{u}_b$ e $\mathbf{u}_f$ são respectivamente as velocidades das fases gasosa e líquida.

Como mostrado na figura 7.31, o coeficiente de força de sustentação transversal calculado se aproxima bem dos dados experimentais para a bolha grande. Considerando que a bolha está totalmente desenvolvida no escoamento cisalhante, o coeficiente calculado é $C_T = -0.292$ enquanto que o experimental é dado por $C_T = -0.28$. Similarmente, o coeficiente para a bolha pequena foi calculado como sendo $C_T = 0.176$, enquanto que o coeficiente baseado nos resultados experimentais é $C_T = 0.29$ [Tomiya, 2002].

Um estudo de resolução de malha pode ser visto na figura (7.32). Enquanto os resultados para a bolha grande mostram que a resolução utilizada é adequada, os resultados para a bolha pequena mostram a necessidade da utilização de malhas mais finas. No gráfico das trajetórias da bolha pequena é possível notar que para as duas malhas mais finas, os resul-

Figura 7.31: Evolução temporal do coeficiente de força de sustentação transversal para as bolhas pequena e grande.

Figura 7.32: Estudo de resolução para as bolhas pequena e grande, onde D é o diâmetro da bolha e h é o espaçamento da malha. A razão D/h representa a quantidade de células presente no diâmetro equivalente da bolha na direção x .

tados obtidos foram quase idênticos. Porém, em se tratando de um método de diferenças finitas utilizando-se malhas uniformes no espaço, os níveis de refinamento mostrados na figura (7.32) são proibitivos, em termos de tempo computacional necessário para se resolver o problema. Para tornar a simulação viável, além do referencial se movendo com velocidade da bolha, foi necessária uma drástica redução no domínio computacional nas três direções, acarretando na interferência das condições de contorno nos resultados obtidos, degradando a solução numérica.

Tal fato justifica o desenvolvimento de métodos que implementem técnicas de refinamento adaptativo e malhas dinâmicas, tornando mais flexível a definição do grau de refinamento desejado nas áreas mais importantes do escoamento, e aumentando o tamanho dos elementos nas áreas menos importantes, proporcionando uma grande economia em tempo de simulação e um ganho significativo na precisão obtida.

Na próxima seção são apresentados vários resultados como os obtidos aqui com o código de *front-tracking/front-capturing*. Nestes resultados pode-se ver claramente as vantagens da utilização da metodologia de malhas dinâmicas com técnicas de refinamento adaptativo.

7.4.2 Soluções obtidas com método ALE 2D

Nesta seção são apresentados resultados semelhantes aos dos gerados pelo método *front-tracking/front-capturing* visto acima, como aplicação do método desenvolvido neste trabalho a um problema relevante. Para o primeiro caso simulado, são considerados os mesmos dados da seção anterior, extraídos de [Tomiya, 2002], utilizando um domínio bidimensional. O segundo caso consiste na captura da inversão da força de sustentação para as bolhas simuladas em [Ervin and Tryggvason, 1997] por um método de *front-tracking*, também para o caso bidimensional.

Dados de [Tomiya, 2002]

Para este caso, foram utilizados os mesmos parâmetros físicos da seção anterior, ou seja, $\rho_f = 1166 \text{ kg/m}^3$, $\mu_f = 0.022 \text{ kg/m} \cdot \text{s}$ e $\sigma = 0.061 \text{ N/m}$. Foram consideradas bolhas de diâmetros, $D = 2.91 \text{ mm}$, $D = 3.52 \text{ mm}$ e $D = 5.53 \text{ mm}$ imersas em um escoamento cisalhante cuja intensidade é $\omega = 6.2 \text{ s}^{-1}$.

Para evitar problemas com a interação entre a esteira das bolhas e as fronteiras do escoamento, foi utilizado um domínio longo $L \times 3L$, onde $L = 30 \text{ mm}$ é a largura do canal. Este move-se com a mesma velocidade da bolha, seguindo a idéia do referencial móvel da seção anterior. Assim, a bolha permanecerá parada em seu referencial, movendo-se

Figura 7.33: Evolução temporal da velocidade vertical, vista no referencial da bolha, para $D = 2.91 \text{ mm}$. Os tempos são adimensionalizados por ω .

livremente na direção horizontal. Portanto, como a bolha se moverá pouco no domínio, foi adotado um movimento de malha puramente elástico, considerando $\beta_1 = 0$ e $\beta_2 = 1$ (Note que para a interface, $\beta_1 = 1$ sempre).

A malha foi refinada na região da interface e da esteira da bolha, e feita mais grossa nas regiões mais afastadas da bolha. Os valores dos tamanhos mínimos nas malhas são $h_{min} = 0.1 \text{ mm}$ para a bolha com $D = 2.91 \text{ mm}$, $h_{min} = 0.15 \text{ mm}$ para a bolha com $D = 3.52 \text{ mm}$ e $h_{min} = 0.2 \text{ mm}$ para a bolha com $D = 5.53 \text{ mm}$. O refinamento na esteira da bolha foi garantido pela adição de um termo fonte adequado ℓ_D na equação (6.11).

As figuras (7.33)-(7.35) mostram o campo de velocidade na direção vertical para as três bolhas consideradas neste exemplo. Por estas figuras, é possível notar que as esteiras das bolhas com $D = 3.52 \text{ mm}$ e $D = 5.53 \text{ mm}$ são instáveis, devido a liberação de vórtices. Este fato faz com que a trajetória seja oscilatória, como pode ser visto na figura 7.36. Verifica-se nesta figura que as bolhas se dirigiram à parede onde há maior velocidade no escoamento cisalhante, diferentemente do que foi observado na seção anterior. Contudo, não há como comparar estes resultados com os dados experimentais de [Tomiya, 2002], uma vez que o domínio bidimensional representa bolhas cilíndricas e não elipsoidais, sendo a física do escoamento diferente do experimento.

Observou-se nos resultados obtidos, que o tamanho da esteira da bolha é maior que a largura L do canal, reforçando a hipótese de que o domínio para simulação deste problema

Figura 7.34: Evolução temporal da velocidade vertical, vista no referencial da bolha, para $D = 3.52 \text{ mm}$. Os tempos são adimensionalizados por ω .

Figura 7.35: Evolução temporal da velocidade vertical, vista no referencial da bolha, para $D = 5.53 \text{ mm}$. Os tempos são adimensionalizados por ω .

Figura 7.36: Trajetórias e número de *Reynolds* das bolhas.

Figura 7.37: Domínio utilizado para o estudo da inversão da força de sustentação.

deve ser longo o suficiente para que as condições de contorno não interfiram no escoamento da bolha. O próximo exemplo simula bolhas bidimensionais, com dados extraídos do trabalho de Ervin & Tryggvason [Ervin and Tryggvason, 1997], que também fez simulações bidimensionais para capturar a inversão da força de sustentação de bolhas diferentes.

Dados de [Ervin and Tryggvason, 1997]

Segundo o que foi concluído por [Tomiya, 2002], o sentido da migração horizontal da bolha em um escoamento cisalhante depende da deformação da mesma. Bolhas maiores se deformam mais, e portanto sofrem mais a ação do escoamento, forçando a migração para zonas de maior velocidade. Em contrapartida, bolhas menores tendem a se manter esféricas, e portanto se comportam praticamente como partículas sólidas, migrando para zonas de menor velocidade no escoamento.

Figura 7.38: Trajetórias e número de *Reynolds* para as bolhas.

Como visto acima, a comparação entre simulações bidimensionais e dados experimentais é difícil de ser feita, uma vez que descrevem problemas físicos diferentes. Para corroborar o método desenvolvido aqui, é feita uma comparação com os resultados numéricos obtidos em [Ervin and Tryggvason, 1997] também para um domínio bidimensional. Considere um domínio $L \times 3L$ (veja figura 7.37), se movendo com a velocidade da bolha, como no exemplo anterior, e os números de Reynolds baseado no escoamento cisalhante Re_ω e constante de capilaridade, dados por

$$Re_\omega = \frac{\rho_f \omega D^2}{\mu_f}, \quad Ca = \frac{\omega D \mu_f}{2\sigma}, \quad (7.12)$$

onde ω é a intensidade do escoamento cisalhante, ou seja, $\omega = U/L$, e U é a velocidade que gera tal escoamento. Considera-se para este caso $Re_\omega = 16$ e $Fr^2 = 0.2$.

Para o estudo da inversão da força de sustentação para este exemplo, considera-se duas bolhas de mesmo tamanho $D/L = 0.2$, mas com constantes de capilaridade diferentes, $Ca = 0.01$ e $Ca = 0.08$. Para $Ca = 0.01$, a tensão interfacial mantém a bolha aproximadamente circular portanto induzindo um movimento lateral para a direita (onde se encontra a parede com menor velocidade), e para $Ca = 0.08$, a bolha se deforma, e sua interação o escoamento cisalhante induz um movimento para a esquerda (onde se encontra a parede com maior velocidade).

A figura 7.38 mostra as trajetórias e o número de *Reynolds* em função do tempo para as bolhas. A evolução da velocidade na direção vertical e os perfis das bolhas obtidos com método *ALE* podem ser vistos nas figuras (7.39) e (7.40).

Uma comparação entre os perfis obtidos e da posição horizontal das bolhas pode ser visto na figura 7.41. Para o tempo adimensional $t^* = 3.0$, os deslocamentos ho-

Figura 7.39: Evolução temporal da interface e velocidade vertical vista no referencial da bolha, para $Ca = 0.01$. Os tempos mostrados são adimensionais ($t^* = \omega t$).

Figura 7.40: Evolução temporal da interface e velocidade vertical vista no referencial da bolha, para $Ca = 0.08$. Os tempos mostrados são adimensionais ($t^* = \omega t$).

rizontais obtidos por [Ervin and Tryggvason, 1997] foram de 0.105 para $Ca = 0.01$ e -0.125 para $Ca = 0.08$, e os obtidos pelo método *ALE* foram respectivamente 0.0991 e -0.0887 . A discrepância pode ser atribuída ao fato de que as simulações feitas em [Ervin and Tryggvason, 1997] utilizam um domínio periódico vertical, ao invés de um referencial se movendo com a bolha, como nas simulações feitas aqui. A malha utilizada para os resultados de [Ervin and Tryggvason, 1997] foi de 128^2 células quadradas para discretizar um domínio $L \times L$, o que equivaleria a uma malha com 32768 elementos triangulares, enquanto as malhas utilizadas aqui são da ordem de 2500 elementos triangulares para discretizar um domínio $L \times 3L$. Exemplos da malha computacional em tempos diferentes podem ser vistos na figura 7.42.

Esta comparação revela que o método foi extremamente bem sucedido na captura deste fenômeno, com comportamento semelhante a outra técnica computacional. Estes resultados revelam o grande potencial do método proposto aqui para este tipo de aplicação, sendo a sua extensão para 3D altamente justificável.

Figura 7.41: Comparação entre os perfis e posição das bolhas inicial (linha serrilhada), $Ca = 0.01$ (direita) e $Ca = 0.08$ (esquerda), no tempo $t^* = 3.0$.

Figura 7.42: Configuração das malhas em tempos diferentes, para $Ca = 0.01$ (a)-(b) e para $Ca = 0.08$ (c)-(d).

Capítulo 8

Considerações finais

Esta dissertação de doutorado teve como finalidade descrever um método numérico desenvolvido para a solução das equações de *Navier-Stokes* para escoamentos multifásicos em malhas dinâmicas. Para tanto, as equações de conservação de quantidade de movimento linear e conservação de massa foram desenvolvidas em um referencial se movendo com velocidade arbitrária, e discretizadas utilizando-se o método de elementos finitos. O tipo de elemento escolhido para a discretização foi o mini-elemento, conhecido da literatura de elementos finitos. Apesar de ser de ordem linear, sua escolha foi justificada devido a natureza do método proposto, que implementa técnicas de malhas dinâmicas e refinamento adaptativo para melhorar a precisão.

Para a solução das equações de conservação, foi utilizado um método de projeção baseado em decomposição LU por blocos, onde uma velocidade intermediária e pressão são calculados sequencialmente, sendo a velocidade final obtida a partir da correção da velocidade intermediária, utilizando-se o gradiente da pressão encontrada. Tal método possui boas propriedades de conservação de massa, encontrando soluções para a velocidade que são discretamente solenoidais. Este método foi adaptado para resolver escoamentos multifásicos, com massa específica e viscosidade variáveis em todo o domínio, mas constantes no interior de cada fase.

O método implementado foi amplamente validado para escoamentos monofásicos, utilizando-se testes bem conhecidos na literatura, comparando-se os resultados com resultados de outros métodos e também com expressões analíticas. De maneira geral, o método de projeção implementado sobre uma discretização por elementos finitos se mostrou equivalente aos outros métodos disponíveis, com resultados muito próximos aos reportados na

literatura.

A representação da interface foi feita utilizando-se a característica dinâmica da malha computacional. Esta é representada por um conjunto de vértices e arestas que fazem parte da triangulação do domínio, e os elementos são marcados como sendo de um determinado fluido. Desta maneira, foi possível implementar uma técnica para separação e junção de interfaces no código, baseando-se no fato de que elementos que compartilhem interfaces diferentes identificam as regiões onde devem ser feitas as mudanças topológicas. A aplicação desta técnica à simulação do colapso de duas bolhas ascendendo em um fluido foi bastante satisfatória, com resultados comparáveis a outros reportados na literatura.

O cálculo da tensão interfacial foi feito através da distribuição da força, inicialmente concentrada nos vértices, aos nós livres vizinhos à interface, utilizando-se o gradiente de uma função *Heaviside* discretizada nos pontos da malha. A distribuição é necessária devido ao tipo de elemento escolhido, o qual representa a pressão por funções de interpolação contínuas. A utilização da força concentrada nos vértices pode causar oscilações no campo de pressão, levando a efeitos de *undershooting* e *overshooting*. Este esquema foi validado através da simulação de uma bolha estática e de uma bolha oscilante, com bons resultados em ambas. Adicionalmente, verificou-se que o cálculo da tensão interfacial e distribuição da força tem ordem de convergência quadrática.

O movimento da malha no método em questão foi feito através da combinação entre a velocidade calculada do escoamento e uma velocidade dita elástica, que tende a posicionar os vértices de modo a suavizar a distribuição de espaçamentos na malha. A combinação entre estas duas abordagens é realizada através de parâmetros que indicam a porcentagem da utilização de cada velocidade. A interface é movimentada segundo um esquema *Lagrangeano*, ou seja, utilizando-se o campo de velocidade do fluido, mais um esquema de movimentação de pontos sobre a interface, na direção tangencial. As técnicas de movimentação dos pontos na interface minimizaram o número de operações de inserção e remoção na malha, como pôde ser observado nos resultados, quando comparado com uma estratégia puramente *Lagrangeana*.

Conforme a malha se movimenta no domínio, foi necessário implementar técnicas de controle, para manter uma boa triangulação e consequentemente não degradar a solução numérica. Neste trabalho, técnicas de inserção e remoção de pontos utilizando-se triangulações de *Delaunay* foi implementada. Com o movimento, a malha deixa de ser *Delaunay*, porém as técnicas garantem a qualidade dos novos elementos, satisfazendo as propriedades de *Delaunay* localmente.

Adicionalmente, uma técnica para refinamento adaptativo foi desenvolvida. Esta

baseia-se na solução de uma equação de *Helmholtz* para a distribuição dos tamanhos na malha, suavizando uma distribuição de tamanhos desejada pré-estabelecida. Esta distribuição, adicionada como termo fonte na equação de *Helmholtz*, pode ser obtida a partir de uma estimativa de erro do escoamento, neste caso, utilizando a *Hessiana* das velocidades. O esquema de refinamento adaptativo se mostrou bastante vantajoso para os resultados mostrados aqui.

A aplicação das técnicas desenvolvidas aqui foi feita para um problema atual e relevante para a comunidade científica. Trata-se da captura da inversão da força de sustentação de bolhas ascendendo em escoamentos cisalhantes. Dados experimentais mostram que bolhas pequenas tentam a se aproximar da parede estacionária, enquanto bolhas grandes tendem a se aglomerar no centro do escoamento vertical. Este fenômeno, ainda não inteiramente compreendido, é de grande interesse para a indústria de extração de petróleo.

A análise deste fenômeno foi feita na literatura utilizando-se experimentos, porém, técnicas numéricas são mais vantajosas no sentido de propiciarem um maior entendimento do que está acontecendo no escoamento. Apesar disto, há poucos trabalhos disponíveis que apresentem simulações numéricas deste problema. Neste sentido, o código desenvolvido aqui foi utilizado para a simulação deste fenômeno, e comparado com outro código tridimensional, com discretizações por diferenças finitas e implementando uma técnica de *front-tracking/front-capturing*. As soluções para o código tridimensional foram bastante satisfatórias, capturando qualitativamente a deformação e sentido da migração horizontal das bolhas. Contudo, uma comparação quantitativa revela a necessidade do aumento do domínio computacional, tornando proibitiva a utilização de métodos de solução baseados em malhas estruturadas uniformes.

Assim, para ilustrar o potencial do método desenvolvido, o mesmo problema foi simulado, com resultados bastante satisfatórios. Simulações foram feitas utilizando-se dados experimentais, que revelaram complexas circulações na esteira da bolha, que influenciam altamente a sua migração horizontal. Contudo, uma comparação quantitativa com os mesmos dados experimentais não é possível devido à diferenças entre as representações bidimensional e tridimensional.

Para ilustrar a captura da inversão da força de sustentação no código bidimensional, foram simulados exemplos extraídos de [Ervin and Tryggvason, 1997], que obteve resultados numéricos para este problema utilizando um código *front-tracking* bidimensional. Uma comparação qualitativa mostra que houve a inversão na força de sustentação, assim como reportado em [Ervin and Tryggvason, 1997], o que corrobora os resultados obtidos com o método aqui desenvolvido.

Observa-se que o método *ALE* com suavização de malha empregado nestas simulações obteve bons resultados, sendo o ideal para este tipo de problema, onde a interface se deforma, mas se movimenta pouco em relação a seu domínio referencial. Este fato ilustra o grande potencial do método para estes tipos de aplicação, onde técnicas de refinamento e malhas dinâmicas são imprescindíveis para obter soluções com um nível de precisão aceitável.

Simulações tridimensionais garantem uma maior representatividade do problema em questão, portanto, a extensão deste método para 3D é fortemente recomendada, e perfeitamente factível. As maiores dificuldades na extensão para 3D são a representação da estrutura de dados e o método de controle de malha, que envolve operações de *flipping* afim de manter a malha localmente *Delaunay*, elevando consideravelmente a complexidade de implementação em um espaço tridimensional. Uma alternativa, por exemplo, poderia ser a utilização de técnicas de controle de malha não-*Delaunay* com controle de qualidade.

De forma geral, o método proposto neste trabalho constitui uma significativa contribuição para a área de simulação numérica de escoamentos multifásicos, e para o desenvolvimento de técnicas mais robustas para a simulação numérica em malhas dinâmicas, com um grande número de aplicações a problemas relevantes para a comunidade científica e indústria.

Apêndice A

Teoremas importantes

A seguir, são apresentados alguns teoremas utilizados no desenvolvimento das equações de conservação para escoamentos multifásicos, e na formulação variacional de tais equações. As demonstrações destes teoremas são bem conhecidas e podem ser facilmente encontradas na literatura disponível sobre cálculo diferencial.

Teorema A.1 (Aproximação de Weierstrass) *Seja $\phi : U \subset \mathbb{R}^m \rightarrow \mathbb{R}$ uma função contínua definida no compacto U . Então, para qualquer $\varepsilon > 0$, existe um inteiro $n = n(\varepsilon)$ e uma forma polinomial $P_n(\mathbf{x})$ de grau no máximo n tal que*

$$|\phi(\mathbf{x}) - P_n(\mathbf{x})| < \varepsilon ,$$

para todo $\mathbf{x} \in U$.

Teorema A.2 (Integração por partes) *Seja $\Omega \subset \mathbb{R}^m$, $\Gamma = \partial\Omega$ o bordo de Ω e $\phi, \psi : U \subset \mathbb{R}^m \rightarrow \mathbb{R}$ campos escalares. Então*

$$\int_{\Omega} \phi \nabla \psi \cdot \mathbf{n} \, d\Omega - \int_{\Gamma} \phi \psi \, d\Gamma = \int_{\Omega} \psi \nabla \phi \cdot \mathbf{n} \, d\Omega$$

Teorema A.3 (Primeira forma de Green) *Seja $\Omega \subset \mathbb{R}^m$, $\Gamma = \partial\Omega$ o bordo de Ω e $\phi, \psi : U \subset \mathbb{R}^m \rightarrow \mathbb{R}$ campos escalares. Então*

$$\int_{\Omega} (\phi \nabla^2 \psi + \nabla \phi \cdot \nabla \psi) \, d\Omega = \int_{\Gamma} \phi \nabla \psi \cdot \mathbf{n} \, d\Gamma .$$

Teorema A.4 (Segunda forma de Green) *Seja $\Omega \subset \mathbb{R}^m$, $\Gamma = \partial\Omega$ o bordo de Ω e $\phi, \psi : U \subset \mathbb{R}^m \rightarrow \mathbb{R}$ campos escalares. Então*

$$\int_{\Omega} (\phi \nabla^2 \psi + \psi \nabla^2 \phi) \, d\Omega = \int_{\Gamma} \left(\phi \frac{\partial \psi}{\partial n} + \psi \frac{\partial \phi}{\partial n} \right) \, d\Gamma .$$

Teorema A.5 (Teorema de Green para campos vetoriais) *Seja $\Omega \subset \mathbb{R}^m$, $\Gamma = \partial\Omega$ o bordo de Ω e $\mathbf{u}, \mathbf{w} : U \subset \mathbb{R}^m \rightarrow \mathbb{R}^m$ campos vetoriais. Então*

$$\int_{\Omega} \{ (\nabla^2 \mathbf{u}) \cdot \mathbf{w} + (\nabla \mathbf{u} : \nabla \mathbf{w}^T) \} \, d\Omega = \int_{\Gamma} \mathbf{n} \cdot (\nabla \mathbf{u} \cdot \mathbf{w}) \, d\Gamma .$$

Teorema A.6 (Divergência de Gauss) *Seja $\Omega \subset \mathbb{R}^m$, $\Gamma = \partial\Omega$ o bordo de Ω e $\mathbf{u} : U \subset \mathbb{R}^m \rightarrow \mathbb{R}^m$ um campo vetorial. Então*

$$\int_{\Omega} \nabla \cdot \mathbf{u} \, d\Omega = \int_{\Gamma} \mathbf{u} \cdot \mathbf{n} \, d\Gamma .$$

onde $\mathbf{n}$ é o vetor normal à Γ .

Teorema A.7 (Teorema de Leibnitz) *Seja $R(t) \subset \mathbb{R}^m$ uma região movendo-se no espaço com velocidade $\mathbf{w}$, $S(t) = \partial R(t)$ o bordo de $R(t)$, e $\mathbf{T}$ uma grandeza de dimensão $k < \infty$. Então*

$$\frac{d}{dt} \int_{R(t)} \mathbf{T}(\mathbf{x}, t) \, dV = \int_R \frac{\partial}{\partial t} \mathbf{T}(\mathbf{x}, t) \Big|_{\mathbf{x}} \, dV + \int_S (\mathbf{n} \cdot \mathbf{w}) \mathbf{T} \, dS .$$

Apêndice B

Produtos de vetores e tensores

B.1 Produto diádico ($\otimes$)

O produto diádico é definido por

$$\begin{aligned}\otimes : \mathbb{R}^m \times \mathbb{R}^m &\rightarrow \mathbb{R}^{m^2} \\ (\mathbf{u}, \mathbf{v}) &\mapsto \mathbf{u} \otimes \mathbf{v} = \mathbf{A} = (u_i v_j)\end{aligned}$$

B.2 Produto escalar entre tensores ($:$)

O produto escalar entre tensores é definido por

$$\begin{aligned}: : \mathbb{R}^{m^2} \times \mathbb{R}^{m^2} &\rightarrow \mathbb{R} \\ (\mathbf{A}, \mathbf{B}) &\mapsto \mathbf{A} : \mathbf{B} = \sum_i \sum_j A_{ij} B_{ji}\end{aligned}$$

Referências Bibliográficas

- [Amsden and Harlow, 1970] Amsden, A. and Harlow, F. H. (1970). The SMAC Method: A Numerical Technique for Calculating Incompressible Fluid Flows. *Los Alamos Scientific Lab. Report LA-4370*.
- [Aris, 1962] Aris, R. (1962). *Vectors, tensors, and the basic equations of fluid mechanics*. Dover, New York.
- [Armaly et al., 1983] Armaly, B. F., Durst, F., Pereira, J. C. F., and Sconung, B. (1983). Experimental and theoretical investigation of backward facing step flow. *Journal of Fluid Mechanics*, 127:473–496.
- [Batchelor, 1994] Batchelor, G. K. (1994). *An introduction to fluid dynamics*. Cambridge University Press.
- [Brackbill et al., 1992] Brackbill, J. U., Kothe, D. B., and Zemach, C. (1992). A Continuum Method for Modeling Surface Tension. *Journal of Computational Physics*, 100:335–354.
- [Chang et al., 2002] Chang, W., Giraldo, F., and Perot, B. (2002). Analysis of an Exact Fractional Step Method. *Journal of Computational Physics*, 180:183–199.
- [Chorin, 1968] Chorin, A. J. (1968). Numerical solution of the Navier-Stokes equations. *Mathematics of Computation*, 22:745–762.
- [Cuvelier et al., 1986] Cuvelier, C., Segal, A., and van Steenhoven, A. A. (1986). *Finite Element Methods and Navier-Stokes Equations*. Reidel Publishing Company, Dordrecht, Holland.
- [Davies, 1980] Davies, A. J. (1980). *The finite element method: a first approach*. Clarendon Press - Oxford.

- [Deen et al., 2004] Deen, N. G., van SintAnnaland, M., and Kuipers, J. A. M. (2004). Multi-scale modeling of dispersed gas liquid two-phase. *Chemical Engineering Science*, 59:1853–1861.
- [Devillers, 2002] Devillers, O. (2002). On Deletion in Delaunay Triangulations. *International Journal of Computational Geometry & Applications*, 12(3):193–205.
- [Dufour and Pelletier, 1998] Dufour, S. and Pelletier, D. (1998). An adaptive finite element method for multiphase flows with surface tension. *Computational Mechanics: New Trends and Applications*, pages 1–18.
- [Edelsbrunner et al., 2000] Edelsbrunner, H., Li, X.-Y., Miller, G. L., Stathopoulos, A., Talmor, D., Teng, S.-H., Ungör, A., and Walkington, N. (2000). Smoothing cleans up sliver. In *Proceedings of 32nd Annu. ACM Symposium in Theory of Computation*, pages 273–277.
- [Ervin and Tryggvason, 1997] Ervin, E. A. and Tryggvason, G. (1997). The rise of bubbles in a vertical shear flow. *Journal of Fluids Engineering*, 119:443–449.
- [Esmaeeli and Tryggvason, 1998] Esmaeeli, A. and Tryggvason, G. (1998). Direct Numerical Simulations of Bubbly Flows. Part 1. Low Reynolds Number Arrays. *Journal of Fluid Mechanics*, 337:313–345.
- [Esmaeeli and Tryggvason, 1999] Esmaeeli, A. and Tryggvason, G. (1999). Direct Numerical Simulations of Bubbly Flows. Part 2. Moderate Reynolds Number Arrays. *Journal of Fluid Mechanics*, 385:325–358.
- [Fortin, 1981] Fortin, M. (1981). Old and new finite elements for incompressible flows. *International Journal for Numerical Methods in Fluids*, 1:347–364.
- [Fortuna, 2000] Fortuna, A. O. (2000). *Técnicas Computacionais para Dinâmica dos Fluidos*. Edusp.
- [Ginzburg and Wittum, 2001] Ginzburg, I. and Wittum, G. (2001). Two-phase flows on interface refined grids modeled with VOF, staggered finite volumes, and splines interpolants. *Journal of Computational Physics*, 166:302–335.
- [Glimm et al., 1988] Glimm, J., Grove, J., Lindquist, W. B., McBryan, O., and Tryggvason, G. (1988). The bifurcation of tracked scalar waves. *SIAM Journal of Computation*, pages 61–79.

- [Gresho, 1990] Gresho, P. (1990). On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via the finite element method that also introduces a nearly consistent mass matrix. Part 1: Theory. *International Journal for Numerical Methods in Fluids*, 11:587–620.
- [Gresho and Chan, 1990] Gresho, P. and Chan, S. T. (1990). On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via the finite element method that also introduces a nearly consistent mass matrix. Part 2: Implementation. *International Journal for Numerical Methods in Fluids*, 11:621–659.
- [Grossetete, 1995] Grossetete, C. (1995). Experimental investigation of void profile development in a vertical cylindrical pipe. *Advances in Multiphase Flow*, pages 333–346.
- [Harlow and Welch, 1965] Harlow, F. H. and Welch, J. E. (1965). Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface. *The Physics of Fluids*, 8(1):2182–2189.
- [Hirt et al., 1997] Hirt, C. W., Amsden, A. A., and Cook, J. L. (1997). An Arbitrary Lagrangian-Eulerian Computing Method for All Flow Speeds. *Journal of Computational Physics*, 135:203–216.
- [Hirt et al., 1970] Hirt, C. W., Cook, J. L., and Butler, T. D. (1970). A Lagrangean Method for Calculating the Dynamics of an Incompressible Fluid with Free Surface. *Journal of Computational Physics*, 5:103–124.
- [Hirt and Nichols, 1981] Hirt, C. W. and Nichols, B. D. (1981). Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39:201–225.
- [Hughes, 1981] Hughes, T. J. R. (1981). Lagrangian-eulerian finite element formulation for incompressible viscous flows. *Computational Methods Applied to Mechanical Engineering*, 29:329–349.
- [Jaeger and Carin, 2002] Jaeger, M. and Carin, M. (2002). The Front-Tracking ALE Method: Application to a Model of the Freezing of Cell Suspensions. *Journal of Computational Physics*, 179:704–735.
- [Jankowski, 1998] Jankowski, J. A. (1998). *A non-hydrostatic model for free surface flows*. PhD thesis, Universität Hannover.

- [Kreyszig, 1991] Kreyszig, E. (1991). *Differential Geometry*. Dover, New York.
- [Lee et al., 2001] Lee, M. J., Oh, B. D., and Kim, Y. B. (2001). Canonical fractional-step methods and consistent boundary conditions for the incompressible navier-stokes equations. *Journal of Computational Physics*, 168:73–100.
- [Liu, 1993] Liu, T. J. (1993). Bubble size and entrance length effects on void development in a vertical channel. *International Journal of Multiphase Flow*, 19:99–113.
- [Luo et al., 2004] Luo, H., Baum, J. D., and Löhner, R. (2004). On the computation of multi-material flows using ALE formulation. *Journal of Computational Physics*, 194:304–328.
- [Löhner, 1995] Löhner, R. (1995). Mesh Adaptation in Fluid Mechanics. *Engineering Fracture Mechanics*, 50(5/6):819–847.
- [Magnaudet et al., 2003] Magnaudet, J., Takagi, S., and Legendre, D. (2003). Drag, deformation and lateral migration of a buoyant drop moving near a wall. *Journal of Fluid Mechanics*, 476:115–157.
- [Mangiavacchi et al., 2002] Mangiavacchi, N., Nonato, L. G., Izquierdo, D. R., and Castelo, A. (2002). A least-square meshless finite-difference method for the Navier-Stokes equations. In *Proceedings of ACOMEN'2002*, volume 1, Liege.
- [McKee et al., 2004] McKee, S., Tomé, M. F., Cuminato, J. A., Castelo, A., and Ferreira, V. G. (2004). Recent advances in the Marker and Cell Method. *Archives in Computational Methods for Engineering*, 11(2):107–142.
- [Nachbin, 2001] Nachbin, A. (2001). Aspectos de modelagem matemática em dinâmica dos fluidos. 23º Colóquio Brasileiro de Matemática, IMPA.
- [Ni et al., 2003] Ni, M.-J., Komori, S., and Morley, N. (2003). Projection methods for the calculation of incompressible unsteady flows. *Numerical Heat Transfer*, 44:553–551.
- [Patankar, 1980] Patankar, S. (1980). *Numerical heat transfer and fluid flow*. Hemisphere, New York.
- [Patankar and Spalding, 1972] Patankar, S. and Spalding, D. (1972). A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *International Journal of Heat and Mass Transfer*, 15:1787–1806.

- [Perot, 1993] Perot, J. B. (1993). An analysis of the fractional step method. *Journal of Computational Physics*, 108:51–58.
- [Sakata et al., 2004] Sakata, Y., Fujiwara, A., and Hishida, K. (2004). Three-dimensional bubble deformation and surrounding flow structure measured by time-resolved PIV/LIF/SIT. In *Proceedings 5th International Conference on Multiphase Flow*, Yokohama, Japan. CD-ROM.
- [Santos et al., 2001] Santos, F. L. P., Mangiavacchi, N., Castelo, A. C., Tomé, M. F., Ferreira, V. G., and Cuminato, J. A. (2001). Numerical simulation of multi-phase flows using the FreeFlow 2D system. In *Anais do XVI Congresso Brasileiro de Engenharia Mecânica (COBEM)*, Uberlândia, MG. CD-ROM.
- [Shewchuk, 1997] Shewchuk, J. R. (1997). *Delaunay Refinement Mesh Generation*. PhD thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, Pennsylvania.
- [Shridar and Katz, 1995] Shridar, G. and Katz, J. (1995). Drag and forces on microscopic bubbles entrained by a vortex. *Physics of Fluids*, 7:389–399.
- [Sousa and Mangiavacchi, 2002] Sousa, F. S. and Mangiavacchi, N. (2002). Simulação Numérica de Escoamentos Multifásicos Usando o Sistema FreeFlow-3D. Dissertação de mestrado, ICMC-USP.
- [Sousa and Mangiavacchi, 2004] Sousa, F. S. and Mangiavacchi, N. (2004). A lagrangian level-set approach for the simulation of incompressible two-fluid flows. *International Journal for Numerical Methods in Fluids*, 47:1393–1401.
- [Sousa et al., 2002a] Sousa, F. S., Mangiavacchi, N., Castelo, A., Nonato, L. G., Tomé, M. F., and Cuminato, J. A. (2002a). Numerical simulations of 3D free-surface flows with surface tension. In *Proceedings 5th International Meeting on High Performance Computing for Computational Science (VECPAR)*, volume 1, pages 225–238, Porto, Portugal.
- [Sousa et al., 2002b] Sousa, F. S., Mangiavacchi, N., Castelo, A. C., Nonato, L. G., Tomé, M. F., and Cuminato, J. A. (2002b). Numerical simulation of 3D multi-fluid flows. In *TEMA: Tendências em Matemática Aplicada e Computacional*, volume 3, pages 203–212. SBMAC.

- [Sousa et al., 2004] Sousa, F. S., Mangiavacchi, N., Nonato, L. G., Castelo, A. C., Tomé, M. F., Ferreira, V. G., Cuminato, J. A., and McKee, S. (2004). A front-tracking/front-capturing method for the simulation of 3d multi-fluid flows with free surfaces. *Journal of Computational Physics*, 198:469–499.
- [Sussman and Smereka, 1997] Sussman, M. and Smereka, P. (1997). Axisymmetric free boundary problems. *Journal of Fluid Mechanics*, pages 269–294.
- [Sussman et al., 1994] Sussman, M., Smereka, P., and Osher, S. (1994). A Level Set Approach for Computing Solutions to Incompressible Two-Phase Flow. *Journal of Computational Physics*, 114:146–159.
- [Takemura and Magnaudet, 2003] Takemura, F. and Magnaudet, J. (2003). The transverse force on clean and contaminated bubbles rising near a vertical wall at moderate reynolds number. *Journal of Fluid Mechanics*, 495:235–253.
- [Tomé and McKee, 1994] Tomé, M. F. and McKee, S. (1994). GENSMAC: A Computational Marker-and-Cell Method for Free Surface Flows in General Domains. *Journal of Computational Physics*, 110(1):171–189.
- [Tomiya, 2002] Tomiya, A. (2002). Transverse migration of single bubbles in simple shear flows. *Chemical Engineering Science*, 57:1849–1858.
- [Törnberg, 2000] Törnberg, A.-K. (2000). *Interface tracking methods with applications to multiphase flows*. PhD thesis, Royal Institute of Technology (KTH), Stockholm, Sweden.
- [Törnberg and Engquist, 2000] Törnberg, A.-K. and Engquist, B. (2000). A finite element based level set method for multiphase flow applications. *Computing and Visualization in Science*, 3:93–101.
- [Tryggvason et al., 1998] Tryggvason, G., Bunner, B., Ebrat, O., and Tauber, W. (1998). Computations of Multiphase Flows by a Finite Difference/Front Tracking Method. I Multi-Fluid Flows. Lecture Series 1998-03, Von Karman Institute for Fluid Dynamics.
- [Tryggvason et al., 2001] Tryggvason, G., Bunner, B., Esmaeili, A., Juric, D., Al-Rawahi, N., Tauber, W., Han, J., Nas, S., and Janz, Y.-J. (2001). A Front-Tracking Method for the Computations of Multiphase Flow. *Journal of Computational Physics*, 169:708–759.

- [Uchiyama, 2001] Uchiyama, T. (2001). ALE finite element method for gas-liquid two-phase flow including moving boundary based on an incompressible two-fluid model. *Nuclear Engineering and Design*, 205:69–82.
- [Unverdi and Tryggvason, 1992] Unverdi, S. O. and Tryggvason, G. (1992). A Front-Tracking Method for Viscous, Incompressible, Multi-Fluid Flows. *Journal of Computational Physics*, 100:25–37.
- [van der Pijl et al., 2003] van der Pijl, S. P., Segal, A., and Vuik, C. (2003). A Mass-Conserving Level-Set (MCLS) Method for Modelling of Multi-phase Flows. Reports of the Department of Applied Mathematical Analysis Report 03-03, Delft University of Technology, Delft, Holland.
- [Xu et al., 1998] Xu, X., Pain, C. C., Goddard, A. J. H., and Oliveira, C. R. E. (1998). An Automatic Adaptive Meshing Technique for Delaunay Triangulations. *Computer Methods in Applied Mechanics and Engineering*, 161:297–303.
- [Zienkiewicz and Taylor, 2000] Zienkiewicz, O. C. and Taylor, R. L. (2000). *The Finite Element Method Volume 1: The Basics*. Butterworth-Heinemann, 5th edition.