



“Modelagem Automática de Escoamentos em Meios
Porosos via Método dos Elementos Finitos”

Por

Bruno Gustavo Borges Luna

Dissertação de Mestrado

Recife, Agosto/2011



UNIVERSIDADE FEDERAL DE PERNAMBUCO

Bruno Gustavo Borges Luna

**“Modelagem Automática de Escoamentos em Meios
Porosos via Método dos Elementos Finitos”**

Trabalho apresentado ao Programa de Pós-graduação em Engenharia Mecânica do Centro de Tecnologia e Geociências da Universidade Federal de Pernambuco como requisito parcial para obtenção do grau de Mestre em Engenharia Mecânica.

Orientador: *Paulo Roberto Maciel Lyra*
Co-Orientador: *Ramiro Brito Willmersdorf*

Recife, Agosto/2011

Dedico à ...

Agradecimentos

Gostaria de agradecer ...

Resumo

A simulação de escoamentos multifásicos em meios porosos impõe vários desafios de ordem numérica devido a uma série de fatores como os meios altamente anisotrópicos e heterogêneos tratados neste tipo de análise, as equações de natureza acoplada elíptica-hiperbólica associadas às Equações Diferenciais Parciais (EDP) que descrevem o fenômeno, entre outros. Uma das maiores dificuldades encontradas na simulação de numérica de EDPs em geral é a solução de grandes sistemas de equações lineares. Os métodos Multigrid permitem obter uma aceleração de convergência de resolvedores de tais sistemas de equações. Entretanto, tais métodos geralmente exigem uma sequência de malhas progressivamente menos refinadas, cuja geração pode se tornar um problema caso não seja adequadamente tratada. Originalmente este sequência era construída definindo diferentes níveis de refinamento, manual ou automaticamente, começando pela malha mais grosseira. Contudo, esta abordagem frequentemente leva a dificuldades na representação de detalhes geométricos na malha mais fina. Esta estratégia também pode demandar buscas sucessivas de modo a realizar a interpolação através dos diversos níveis, introduzindo erros adicionais. Uma alternativa para superar estes problemas é a geração de toda a hierarquia de malhas de um modo completamente algébrico utilizando o chamado Método Multigrid Algébrico (AMG). Este trabalho apresenta a implementação de um software criado utilizando a ferramenta FEniCS para a geração automática de código de baixo-nível e PyAMG para a criação completamente automática de uma estrutura multiníveis de matrizes que são aplicadas na solução numérica de escoamentos em meios porosos usando o Método dos Elementos Finitos (MEF). Os métodos descritos aqui são gerais o suficiente para lidar com problemas tridimensionais, heterogêneos e anisotrópicos. Exemplos são apresentados e resultados discutidos para problemas bidimensionais com domínios homogêneo e heterogêneos com tensores de permeabilidade iso- e anisotrópicos.

Palavras-chave: Multigrid, Reservatório de Petróleo, Escoamento em Meios Porosos

Abstract

The simulation of multiphase flows in porous media impose many numerical challenges due to a series of factors as the high anisotropic and heterogeneous media handled in this type of analysis, the coupled elliptic-hyperbolic mathematical nature of the associated Partial Differential Equations (PDEs), among others. One of the main difficulties faced by numerical simulations of PDEs is the solution of large systems of linear equations. Multigrid methods allow a faster convergence of linear and non-linear equations systems solvers. However, they normally require a sequence of progressive coarser meshes whose generation can become a issue if not adequately addressed. Originally this sequence has been made setting different levels of refinement, manually or automatically, starting with the coarsest mesh. Nevertheless this approach often leads to difficulties in representing the geometrical details in the first mesh. It can also demand successive searches in order to perform interpolations between levels, introducing extra errors. An alternative to overcome these problems is to generate all the mesh hierarchy in a full algebraic manner using the so-called Algebraic Multigrid(AMG) approach. This work presents the implementation of a software created using the FEniCs tool for automated low-level code generation and PyAMG for the fully automatic creation of consistent multilevel matrix structures to be applied to the numerical solution of porous media flows using the Finite Element Method (FEM). The methods described here are general enough to handle with three-dimensional, heterogeneous and anisotropic problems. Examples are shown and results discussed for two-dimensional problems in homogenous and heterogenous domains with iso- and anisotropic permeability tensors.

Keywords: Multigrid, Oil Reservoir, Porous Media Flow

Sumário

Lista de Figuras	ix
Lista de Tabelas	xii
Lista de Acrônimos	xii
1 Introdução	1
1.1 Motivação	1
1.2 Histórico da Simulação de Reservatórios de Petróleo	3
1.3 Histórico do Multigrid	3
1.4 Histórico da Modelagem Automática de Equações Diferenciais	3
1.5 Objetivos	3
1.6 Organização do Trabalho	4
2 Formulação Matemática	7
2.1 Propriedades	8
2.1.1 Porosidade	9
2.1.2 Permeabilidade Absoluta	10
2.1.3 Permeabilidade Relativa	11
2.2 Equação da Conservação de Massa	13
2.3 Equação de Pressão	14
2.4 Equação de Saturação	16
2.5 Sistema de Equações para Escoamento Bifásico	16
3 Formulação Numérica	17
3.1 Método dos Elementos Finitos de Galerkin	17
3.2 Multigrid	20

4	Implementação Computacional	24
4.1	Estrutura Geral do Programa	25
4.2	Pré-Processamento	25
4.3	Processamento	28
4.3.1	FEniCS	31
4.4	Pós-Processamento	33
5	Resultados	36
5.1	Problemas Elípticos	36
5.1.1	Meio Homogêneo e Isotrópico	37
5.1.2	Meio Homogêneo e Anisotrópico	41
5.1.3	Meio Heterogêneo e Anisotrópico	44
5.2	Problemas de Escoamentos Bifásicos em Meios Porosos	48
5.2.1	1/4 de Cinco Poços Homogêneo	50
5.2.2	1/4 de Cinco Poços Heterogêneo	50
6	Conclusões e Trabalhos Futuros	51
6.1	Conclusões	51
6.2	Trabalhos Futuros	51
Referências Bibliográficas		52
A	Delfine - Manual do usuário	56
A.1	Instalação do Programa	56
A.2	Dados de Entrada	56
A.3	Exemplo Detalhado	56
B	Delfine - Trechos de Código Selecionados	57
B.1	Montagem da Parte Elíptica	57
B.2	Cálculo das Velocidades	57
B.3	Montagem da Parte Hiperbólica	57
B.4	Montagem da Sistema de Equações Lineares	57

Listas de Figuras

2.1	Exemplo de meio poroso com destaque para os volumes de matriz sólida (rocha) e de poros (vazio).	10
2.2	Exemplo de meio poroso saturado com escoamento bifásico.	11
2.3	Exemplo de correlação entre porosidade e permeabilidade absoluta (retirado de Schneider (2003)).	12
2.4	Exemplo de modelo de permeabilidade relativa para escoamentos bifásicos em meios porosos.	13
3.1	Comparação dos métodos Multigrid Geométrico e Algébrico (retirado de Trottenberg <i>et al.</i> (2001)).	21
3.2	Ciclos V, W e F. S representa suavização e E é a solução no nível mais grosso.	23
4.1	Fluxograma da etapa de pré-processamento.	29
4.2	Fluxograma da resolução da parte elíptica.	30
4.3	Interação entre os diversos componentes do projeto <i>FEniCS</i> para definição do problema, seguidos pela resolução no <i>Delfine</i> (adaptado de Rathgeber (2010)).	32
4.4	Estrutura modular do <i>DOLFIN</i> (retirado de Logg and Wells (2010)). . .	32
4.5	Fluxograma da resolução da parte hiperbólica.	34
4.6	Fluxograma da etapa de pós-processamento.	35
5.1	Caso homogêneo e isotrópico. Campo escalar para malha 64x64	38
5.2	Caso homogêneo e isotrópico. Comparação da evolução dos resíduos para malha 32x32	41
5.3	Caso homogêneo e anisotrópico. Campo escalar para malha 64x64 . . .	42
5.4	Caso homogêneo e anisotrópico. Comparação da evolução dos resíduos para malha 64x64	44

Lista de Tabelas

5.1	Erro e taxa de convergência para problema homogêneo e isotrópico . . .	39
5.2	Erro e taxa de convergência para problema homogêneo e isotrópico - literatura	39
5.3	Erro e taxa de convergência para problema homogêneo e anisotrópico .	42
5.4	Erro e taxa de convergência obtidos na literatura para problema homogêneo e anisotrópico	43
5.5	Erro e taxa de convergência para problema heterogêneo e anisotrópico .	47

Listas de Acrônimos

AMG Multigrid Algébrico

EDP Equações Diferenciais Parciais

FFC *FEniCS Form Compiler*

GMG Multigrid Geométrico

MDF Método das Diferenças Finitas

MEF Método dos Elementos Finitos

MEFM Método dos Elementos Finitos Mistos

MVF Método dos Volumes Finitos

RNC *Relax NG Compact Syntax*

RNV *Relax NG Validator*

SUPG *Streamline Upwind Petrov-Galerkin*

UFC *Unified Form-assembly Code*

UFL *Unified Form Language*

XML *Extensible Markup Language*

Capítulo 1

Introdução

1.1 Motivação

Atualmente, a simulação numérica do escoamento de fluidos em meios porosos é fundamental em várias áreas da engenharia, como na análise de transporte de contaminantes em aquíferos (Bear *et al.*, 1992), gerenciamento de água e calor em membranas poliméricas para células a combustível (Matamoros and Brueggemann, 2006) ou para a simulação de escoamentos multifásicos em reservatórios de petróleo (Peaceman, 1977; Aziz and Settari, 1979; Ewing, 1983; Carvalho, 2005; Silva, 2008).

A modelagem científica do escoamento de fluidos em meios porosos para a indústria do petróleo vem sendo feita desde os anos 30 do século passado, inicialmente utilizado modelos de areia compactada para compreender a produção de água em reservatórios de petróleo e o porquê da proporção água-óleo aumentar ao longo do tempo. No final dos anos 30 e começo dos 40 alguns experimentos como modelos eletrolíticos foram usados para representar o escoamento em meios porosos, tendo sido nesta época já reconhecida a analogia entre as leis que regem o fluxo de corrente elétrica e as leis de Darcy (Peaceman and Nash, 1990). Todos esses modelos físicos análogos permitiram a obtenção de um conhecimento mais profundo sobre os fenômenos que governam este tipo de problema, porém apenas com o advento dos computadores eletrônicos é que foi possível para os engenheiros simular problemas do "mundo real", como o de reservatórios de óleo em 2 ou 3 dimensões.

Paralelamente ao desenvolvimento de novos hardwares, houve também um trabalho extensivo em métodos computacionais aplicados ao escoamento multifásico em meios porosos. O primeiro método utilizado em larga escala para resolver Equações Diferenciais Parciais (EDP) numericamente e ainda padrão na indústria do petróleo foi o Método das

1.1. MOTIVAÇÃO

Diferenças Finitas (MDF) (Peaceman and Nash, 1990). Entretanto, este tipo de método apresenta certas desvantagens, como a dificuldade para lidar com domínios complexos, condições de contorno gerais ou propriedades de materiais variáveis ao longo do domínio (Chen *et al.*, 2006), especialmente quando comparado com métodos mais adequados para lidar com malhas não-estruturadas que vêm sendo aplicados mais recentemente nesta área como é o caso do Método dos Volumes Finitos (MVF) (Carvalho, 2005; Cordazzo, 2006) ou o Método dos Elementos Finitos (MEF) (Chen *et al.*, 2006). Por estes motivos, um método da última classe citada foi usado ao longo deste trabalho.

Um aspecto comum a todos estes métodos computacionais é o tempo considerável necessário para o desenvolvimento de programas de computador que implementem formulações para casos gerais e/ou complexos. Muito desse tempo é consumido codificando tarefas que são comuns a praticamente qualquer software de simulação numérica, como por exemplo a montagem das matrizes, manipulação de dados de entrada e saída (I/O) ou a solução iterativa de sistemas de equações lineares. A abordagem adotada neste trabalho para superar esta dificuldade foi utilizar a ferramenta de código aberto *FEniCS/DOLFIN* (Logg and Wells, 2010), a qual permite efetuar a geração automática de código de baixo nível (em C++) baseada em informações fornecidas através de uma interface com um programa escrito em código de alto nível (*Python* ou C++), permitindo ao programador se concentrar no desenvolvimento e teste de diferentes formulações matemáticas e numéricas para o problema de interesse usando uma sintaxe muito similar àquela encontrada na descrição matemática do problema, ao invés de despender tempo escrevendo códigos auxiliares para tarefas administrativas.

Além disso, existe uma necessidade de reduzir o tempo de CPU de um programa de modo a viabilizar o uso de simulações dentro do prazo de um projeto. Normalmente, a parte que mais consome tempo de processamento em uma simulação é a solução do sistema de equações lineares resultante da discretização das EDPs através de algum método numérico (Saad, 2003). Várias técnicas estão disponíveis para resolver de modo iterativo este tipo de sistema, cada uma delas com suas próprias vantagens e desvantagens em termos de velocidade e generalidade, sendo que estas características frequentemente apontam para direções distintas, isto é, um método extremamente rápido para uma classe de problemas muitas vezes não é geral e vice-versa. Neste trabalho foi utilizado um método conhecido por acoplar de modo quase ideal estas duas características: o método Multigrid (Trottenberg *et al.*, 2001; Saad, 2003; Briggs *et al.*, 2000). Todavia, esta particularidade não é obtida sem custo, sendo neste caso as principais dificuldades adicionais associadas aos métodos Multigrid a necessidade de uma sequência de malhas

1.2. HISTÓRICO DA SIMULAÇÃO DE RESERVATÓRIOS DE PETRÓLEO

sucessivamente mais grosseiras cuja geração pode se tornar um problema caso não seja abordada adequadamente e o uso de operadores de transferência de informações entre níveis consecutivos. Seguindo o princípio de economizar tempo de codificação com tarefas gerais e manter o foco no problema de interesse, neste trabalho foi utilizada a biblioteca de código aberto *PyAMG* (Bell *et al.*, 2008), a qual não apenas efetua de modo completamente automático a geração de uma sequência de matrizes dada uma malha inicial, como também permite o uso de diferentes variantes do Método do Multigrid Algébrico (AMG), incluindo os algoritmos clássicos de Ruge-Stüben e o de Agregação Suavizada (*Smooth Aggregation*) (Trottenberg *et al.*, 2001).

1.2 Histórico da Simulação de Reservatórios de Petróleo

1.3 Histórico do Multigrid

1.4 Histórico da Modelagem Automática de Equações Diferenciais

1.5 Objetivos

Dada a relevância do problema proposto, como mencionado na seção 1.1, surgiu o objetivo principal deste trabalho, o qual é o desenvolvimento de um sistema computacional para a simulação de escoamentos bifásicos óleo-água em meios porosos, considerando a flexibilidade de lidar com geometrias bi- e tridimensionais, domínios homo- e heterogêneos e tensores de permeabilidades iso- e anisotrópicos. Além disso, como objetivos secundários temos o uso de técnicas de alto desempenho para diminuição do tempo necessário de processamento para as análises e a utilização de pacotes computacionais que permitem a automação de parte do processo de desenvolvimento do software.

Existem na literatura diversas opções disponíveis de métodos matemáticos, numéricos e computacionais para resolução dos problemas inerentes às metas pretendidas neste trabalho. Em relação à descrição matemática do problema, optou-se pela utilização da clássica formulação de Darcy para modelo o fenômeno de interesse. Esta formulação possui limitações bastante discutidas na literatura, já que a mesma foi originalmente desenvolvida para um contexto diferente (escoamento monofásico laminar em um tubo preenchido com areia). No capítulo 2 serão discutidas as hipóteses que devem ser

consideradas para manter a validade da formulação utilizada. Em todo caso, para o objetivo ao qual este trabalho se destina, o modelo de Darcy é utilizado largamente na indústria e academia (Peaceman and Nash, 1990; Aziz and Settari, 1979; Marle, 1981; Ewing, 1983), tendo se mostrado adequado desde que respeitadas as limitações impostas pelo mesmo.

Analisando as alternativas disponíveis em termos de métodos numéricos, optou-se pelo uso do MEF pela sua combinação de sólida base teórica disponível na literatura (Chavent and Jaffre, 1986; Chen *et al.*, 2006; Hughes, 2000) e grande flexibilidade para lidar com casos que apresentem geometrias e/ou condições de contorno complexas. Como objetivos específicos dentro deste quesito, temos a discretização das EDP usando tanto a formulação de Galerkin quanto do Método dos Elementos Finitos Mistos (MEFM) para a equação de pressão e o método *Streamline Upwind Petrov-Galerkin* (SUPG) para a equação de saturação, estando as duas variáveis acopladas de modo sequencial implícito via velocidade total. Em relação à resolução do sistema de equações provenientes de tal discretização, foi definido como meta específica o uso de alguma técnica de aceleração de convergência que permitisse uma grande escalabilidade de modo a ser possível a simulação de problemas de grande porte. Para este fim, foi escolhida dentre as opções disponíveis o uso do método AMG, devido a sua robustez e eficiência.

Por fim, foi estabelecida a meta de elaborar um sistema computacional de modo prático e utilizando técnicas de geração de código automática, tendo assim um programa mais fácil de ser gerenciado e expandido conforme surjam novas necessidades. No capítulo 4 serão discutidas as opções disponíveis na literatura e o porquê da escolha entre elas da ferramenta *FEniCS/DOLFIN*.

1.6 Organização do Trabalho

Esta dissertação está dividida em seis capítulos, incluindo introdução e conclusão, organizados de modo a apresentar sequencialmente os passos necessários para se alcançar os objetivos definidos na seção 1.5. Além disso, foram anexados ao final do trabalho dois apêndices que lidam com aspectos práticos da implementação e uso do código desenvolvido para obtenção dos resultados apresentados.

- *Capítulo 1: Introdução.* Este presente capítulo contém a motivação para a realização deste trabalho, um curto histórico das principais áreas relacionadas com uma revisão bibliográfica de alguns dos principais trabalhos de outros autores e suas contribuições, e por fim uma apresentação clara dos objetivos desta dissertação.

1.6. ORGANIZAÇÃO DO TRABALHO

- *Capítulo 2: Formulação Matemática.* No segundo capítulo são apresentadas as equações governantes do problema que se deseja modelar. As propriedades de rocha, fluido e de interação rocha-fluido são expostas e a notação utilizada ao longo do trabalho é definida. Além disso, são discutidas as principais hipóteses simplificadoras utilizadas.
- *Capítulo 3: Formulação Numérica.* Lida com a transformação de um problema contínuo definido pelas equações apresentadas no capítulo 2 em um problema discreto que pode ser resolvido de modo aproximado utilizando métodos numéricos, os quais também são discutidos em detalhes.
- *Capítulo 4: Implementação Computacional.* Neste capítulo são abordados os aspectos práticos relativos à implementação de todo o arcabouço teórico desenvolvido. Os principais programas e bibliotecas utilizados são apresentados e comparados com outros similares existentes e as razões para a escolha de cada um é discutida. Por fim, um algoritmo em diagramas de blocos da estrutura geral do programa é exposto, de onde se pode observar as interligações entre os diferentes pacotes adotados e os pontos de interface com o código escrito especificamente para este trabalho.
- *Capítulo 5: Resultados.* A flexibilidade e acurácia do programa desenvolvido é comprovada através do uso de diversos exemplos que variam desde casos simples homogêneos e isotrópicos com solução analítica fechada, até casos mais complexos considerando heterogeneidades no domínio e o uso de tensores de permeabilidade anisotrópicos. Sempre que possível, as soluções obtidas foram comparadas com *benchmarks* da literatura ou com soluções analíticas.
- *Capítulo 6: Conclusões e Trabalho Futuros.* Neste último capítulo são feitas as observações principais a respeito dos objetivos atingidos neste trabalho quando comparado com as metas estabelecidas. Finalmente, são apresentadas algumas possibilidades de trabalhos futuros baseados no que foi discutido ao longo desta dissertação.
- *Anexo 1: Delfine - Manual do Usuário.* Este anexo tem um enfoque bastante prático, cujo objetivo é servir como um tutorial para o leitor interessado em utilizar as ferramentas desenvolvidas neste trabalho. Todos os passos, desde a obtenção do código através da internet até o uso do mesmo, passando pela instalação e

1.6. ORGANIZAÇÃO DO TRABALHO

configuração de pacotes adicionais são discutidos com base em um exemplo simples usado como orientação.

- *Anexo 2: Delfine - Trechos de Código Selecionados.* Compilação de trechos de códigos escritos para este trabalho e discutidos especialmente no capítulo 4. São apresentados ao final da dissertação por um questão de clareza e organização.

Capítulo 2

Formulação Matemática

Neste trabalho, foi adotada a formulação matemática clássica proposta em Peaceman (1977) para o escoamento simultâneo de duas fases imiscíveis em um meio poroso saturado. Esta abordagem vem sendo utilizada por muitos pesquisadores (Ewing, 1983; Chavent and Jaffre, 1986; Carvalho, 2005; Silva, 2008) e tem com uma de suas principais características a manipulação da equação de conservação da massa usando a lei de Darcy de modo a formar um sistema com uma equação de pressão parabólica-elíptica e uma equação de saturação parabólica-hiperbólica (Carvalho, 2005), em oposição a outras formulações nas quais os campos de pressão e saturação são resolvidos simultaneamente em um sistema de EDPs parabólicas (Aziz and Settari, 1979).

A formulação segregada de Peaceman permite o uso de métodos especializados capazes de explorar as particularidades matemáticas de cada equação do sistema resultante. O acoplamento entre estes dois campos é obtido através do uso de um termo de velocidade total.

De modo a utilizar as equações apresentadas ao longo deste capítulo, faz-se necessário a adoção de algumas hipóteses simplificadoras (Carvalho, 2005; Peaceman, 1977):

- Meio poroso totalmente saturado;
- Rochas e fluidos incompressíveis;
- Fluidos imiscíveis;
- Escoamento isotérmico;
- Lei de Darcy válida para as velocidades consideradas.

Para o correto entendimento da formulação utilizada, faz-se necessário introduzir alguns conceitos, os quais serão discutidos na seção seguinte, usados para a descrição macroscópica das propriedades de rochas, fluidos e da interação entre os mesmos.

2.1 Propriedades

Antes de iniciar a descrição das diversas propriedades necessárias para a elaboração do modelo matemático utilizado, é importante fazer um comentário a respeito do modo como essas propriedades se apresentam na realidade e como são de fato representadas na simulação do fenômeno no domínio de interesse.

Para uma perfeita descrição do escoamento de fluidos em meios porosos, seria necessário adotar uma abordagem microscópica, a qual necessitaria do conhecimento das características geométricas de cada um dos poros para a definição das fronteiras e das condições de contorno. De posse dessas informações, seria possível utilizar uma formulação matemática, como as equações de Navier-Stokes (Fortuna, 2000), para simular o escoamento de cada um dos fluidos existentes no meio e a interação entre eles. Desnecessário comentar que tal alternativa é impraticável tanto do ponto de vista computacional, já que seria possível analisar domínios apenas em escalas cuja ordem de grandeza esteja na escala dos poros, quanto do ponto de vista da aquisição de dados de entrada, pois é impossível atualmente se obter toda a descrição morfológica de um meio como um reservatório de petróleo, do qual se obtém normalmente apenas algumas amostras, chamadas de testemunhas, as quais são retiradas de pontos que estão várias centenas de metros ou até quilômetros de distância entre si (Thomas, 2001).

Sendo assim, para todos os efeitos práticos se utiliza na análise numérica de reservatórios de petróleo uma abordagem macroscópica, onde o valor das propriedades são consideradas variáveis contínuas definidas em todo o domínio ocupado pelo meio poroso. Tais valores das grandezas físicas representam para cada ponto específico uma média volumétrica dos valores da região circunvizinha a este ponto. Ou seja, não é mais necessário uma descrição da morfologia exata do reservatório, já que ao invés disso serão utilizadas propriedades não mensuráveis no nível microscópico, mas que representam no nível macroscópico o efeito equivalente ao da estrutura do meio poroso (Hurtado, 2005).

Entretanto, é válido comentar que esta descrição macroscópica do meio poroso apresenta também dificuldades. As formações rochosas que formam os reservatórios de petróleo são frequentemente descritas através de modelos geológicos na escala de metros (Aarnes *et al.*, 2007). Tal escala tornaria em muitos casos inviável a simulação de

escoamentos de fluidos em tais reservatórios, já que os mesmos têm ordem de grandeza de quilômetros, o que implicaria para casos tridimensionais em malhas com vários milhões de células, passíveis de serem analisadas apenas em clusters de grande porte.

Apesar do grande aumento percebido na capacidade computacional do hardware utilizado e também do uso de técnicas avançadas para extrair o máximo de desempenho de tais computadores, os modelos de reservatório a serem analisados na prática frequentemente tem que contar com o uso de técnicas de *Upscaling* (de Souza Menezes, 2009), ou seja, uma restrição dos parâmetros geofísicos obtidos em modelos geológicos para malhas tipicamente dezenas ou centenas de vezes mais grosseiras que serão então analisadas via algum método numérico (Aarnes *et al.*, 2007).

2.1.1 Porosidade

Uma rocha, para ter a capacidade de armazenar petróleo, deve possuir vazios no seu interior, os quais são chamados de poros. Além disso, para que haja fluxo dentro desta rocha-reservatório é necessário que tal poros estejam interconectados para que existam um caminho que os fluidos possam percorrer. Para efeitos de cálculo de porosidade, existe uma diferença entre a porosidade *absoluta*, a qual abrange todos os poros, estejam eles interconectados ou não, e a porosidade *efetiva*, onde apenas os poros ligados entre si são considerados. Apenas esse segundo tipo de porosidade será usado neste trabalho, já que é o de maior interesse para a análise de escoamentos. Portanto, sempre que houver referência à porosidade, pode-se entender como a variante efetiva da mesma.

A porosidade de uma rocha-reservatório é definida matematicamente como a razão entre o volume dos espaços vazios existentes entre os poros da rocha e o volume total da amostra (Thomas, 2001). Logo, podemos descrever a porosidade ϕ como:

$$\phi = \frac{V_p}{V_t} \quad (2.1)$$

onde o volume total é $V_t = V_p + V_s$, V_p é o volume poroso e V_s o volume da matriz sólida. Segundo a abordagem macroscópica adotada, a porosidade será uma função contínua no espaço. Logo, se está assumindo que o valor calculado pela Eq. (2.1) representará a média de um volume representativo com dimensão pequena quando comparada ao meio poroso como um todo, porém grande em comparação com as dimensões características dos poros.

As Fig. (2.1) apresenta esquematicamente uma amostra de meio poroso, com uma aproximação em uma parte da mesma onde se pode observar claramente os conceitos

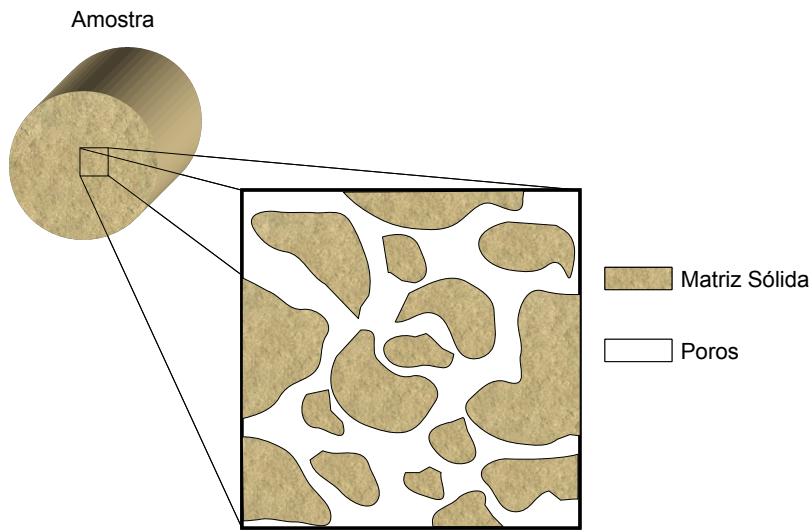


Figura 2.1 Exemplo de meio poroso com destaque para os volumes de matriz sólida (rocha) e de poros (vazio).

de volume poroso e de matriz sólida representados. A caracterização das fases em um escoamento bifásico é vista na Fig. (

2.1.2 Permeabilidade Absoluta

A permeabilidade absoluta é definida como uma medida da capacidade de um meio poroso de escoar uma única fase sob certas condições. Esta é uma propriedade mais difícil de ser medida do que a porosidade, porém mais importante no que tange ao escoamento de fluidos. Não necessariamente irá existir uma proporcionalidade direta entre estes dois parâmetros, já que para a permeabilidade importa não apenas a quantidade de poros interconectados, mas também o modo como eles estão conectados e a tortuosidade do caminho formado por eles. Existe também a possibilidade de se ter rochas com baixíssimas porosidades, como muitas vezes é o caso de algumas rochas carbonáticas, que porém apresentam uma alta permeabilidade devido à presença de fraturas ou falhas. Todavia, para vários casos específicos podem ser geradas correlações entre as duas propriedades, como pode ser visto na Fig. (2.3) (Schneider, 2003).

Normalmente a permeabilidade é representada matematicamente através de um tensor, denominado de K ao longo deste trabalho. Isto significa que não apenas os valores da

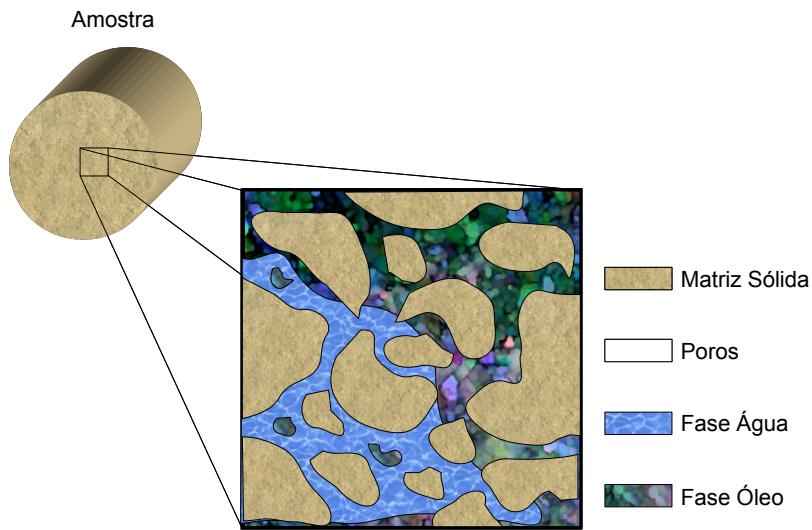


Figura 2.2 Exemplo de meio poroso saturado com escoamento bifásico.

permeabilidade absoluta variam de acordo com a orientação da rocha, como também que, devido aos termos cruzados de um tensor cheio, os valores em uma direção influenciam a permeabilidade nas outras direções. Neste caso o tensor é chamado de anisotrópico. Se K puder ser representado por uma função escalar, ou seja, a permeabilidade em todas as direções for a mesma, este tensor degenerado em um escalar é chamado de isotrópico. Além disso, devido a grande heterogeneidade entre as diferentes formações rochosas, a permeabilidade pode variar espacialmente em ordens de grandeza, sendo que variações locais no intervalo entre 1mD e 10D não são incomuns (Aarnes *et al.*, 2007).

O sistema elaborado neste trabalho pode lidar de modo efetivo tanto com tensores cheios (anisotropia), quanto com variações do mesmo ao longo do domínio (heterogeneidade).

2.1.3 Permeabilidade Relativa

Apesar de termos assumido total imiscibilidade entre as fases é considerado para efeitos de simulação, segundo a abordagem macroscópica adotada neste trabalho, a presença de todas as fases simultaneamente no mesmo ponto, já que como mencionado anteriormente na seção 2.1 o comportamento do escoamento na escala dos poros não é de interesse prático para a análise de reservatórios. Isto significa que todas as fases terão influência

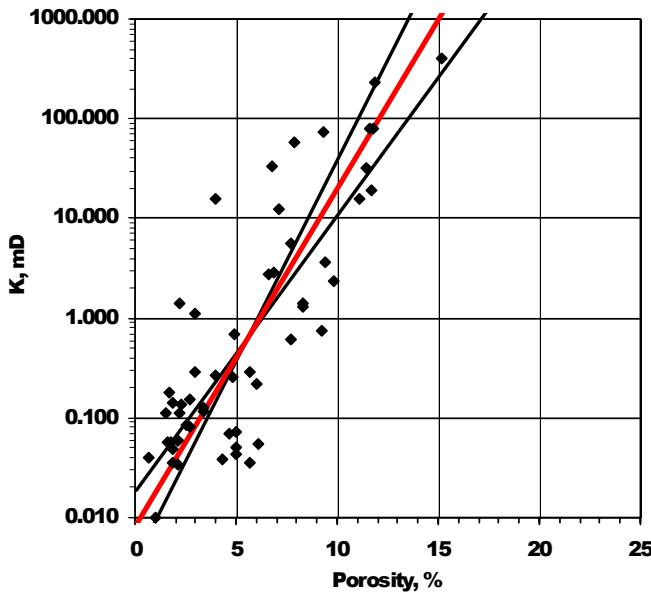


Figura 2.3 Exemplo de correlação entre porosidade e permeabilidade absoluta (retirado de Schneider (2003)).

umas sobre as outras.

Portanto, a permeabilidade de fato percebida por um fluido em um meio poroso será reduzida pela presença das outras fases, resultando portanto na necessidade da definição de um termo de permeabilidade relativa, representado neste trabalho por $k_{r\alpha}$, que irá descrever como uma fase escoa na presença simultânea de outras.

As permeabilidades absoluta K , efetiva K_α e relativa $K_{r\alpha}$ estão relacionadas através da seguinte expressão:

$$k_{r\alpha} = \frac{K_\alpha}{K} \leq 1 \quad (2.2)$$

Dada a complexidade envolvida na interação entre a rocha e as diferentes fases na escala dos poros durante o escoamento, os valores para a permeabilidade relativa em função das propriedades dos fluidos e dos meios porosos são difíceis de serem obtidos. De um modo geral, depende-se de modelos semi-empíricos para uso em simulações de reservatórios de petróleo. Um exemplo típico de curva de permeabilidade relativa tanto da fase água quanto da fase óleo em função da saturação da água usando o modelo de Corey é apresentado na Fig. (2.4).

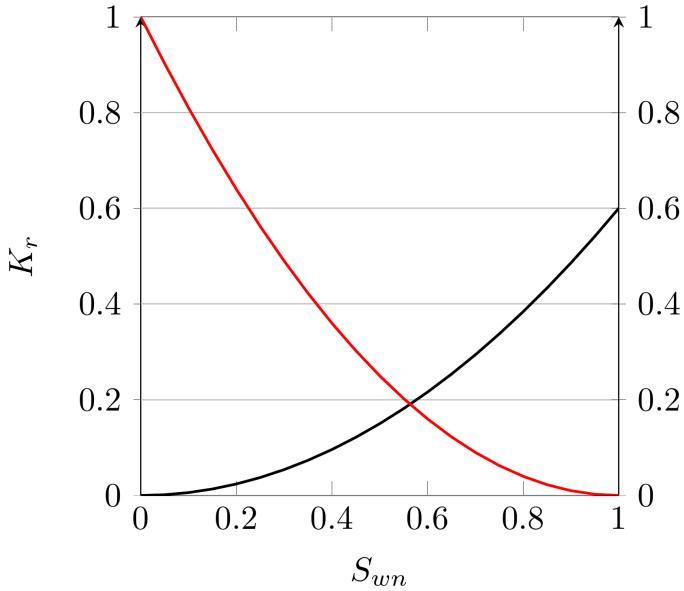


Figura 2.4 Exemplo de modelo de permeabilidade relativa para escoamentos bifásicos em meios porosos.

2.2 Equação da Conservação de Massa

Considerando a hipótese de meio poroso totalmente saturado e a existência de n fases, a equação constitutiva para a saturação pode ser descrita por:

$$\sum_{\alpha=1}^n S_{\alpha} = 1 \quad (2.3)$$

Onde α representa cada fase (neste trabalho, $\alpha = w$ e o para água e óleo, respectivamente) e S_{α} representa a saturação da fase α .

A lei de Darcy generalizada para a velocidade de cada fase é expressa através da seguinte equação:

$$\vec{v}_{\alpha} = -\frac{K_{\alpha}}{\mu_{\alpha}} \nabla (p_{\alpha} - \rho_{\alpha} g \nabla z) \quad (2.4)$$

com K_{α} , μ_{α} , p_{α} e ρ_{α} representando a permeabilidade efetiva, viscosidade, pressão e densidade da fase α , ao passo que g e z representam a aceleração da gravidade e o deslocamento na direção da mesma, respectivamente.

A equação de conservação para cada fase é descrita como em Peaceman (1977):

$$-\nabla \cdot (\rho_{\alpha} \vec{v}_{\alpha}) + q_{\alpha} = \frac{\partial (\phi \rho_{\alpha} S_{\alpha})}{\partial t} \quad (2.5)$$

2.3. EQUAÇÃO DE PRESSÃO

Os termos q_α , ϕ e t representam termos fontes/sumidores da fase α (por exemplo, poços), porosidade efetiva da rocha e o tempo, respectivamente. Além disso, se forem consideradas duas fases, uma molhante (água) e uma não-molhante (óleo), combinando as Eqs. (2.4) e (2.5), e usando a definição dada na Eq. (2.2), é obtido o seguinte sistema de equações diferenciais parciais que resolve o problema do escoamento bifásico dadas as hipóteses já mencionadas:

$$\nabla \cdot \left(\frac{\rho_w K k_{rw}}{\mu_w} \nabla (p_w - \rho_w g \nabla z) \right) + q_w = \frac{\partial (\phi \rho_w S_w)}{\partial t} \quad (2.6a)$$

$$\nabla \cdot \left(\frac{\rho_o K k_{ro}}{\mu_o} \nabla (p_o - \rho_o g \nabla z) \right) + q_o = \frac{\partial (\phi \rho_o S_o)}{\partial t} \quad (2.6b)$$

Nas Eqs. (2.6a) e (2.6b), os campos de pressão e saturação são acoplados em ambas as equações. As mesmas se assemelham à equação de condução de calor e por isso espera-se que tenham um comportamento essencialmente parabólico. Esta assertiva não é necessariamente verdadeira e pode ser avaliada através da obtenção de um par de equações que seja dependente ou da pressão ou da saturação. A derivação de tais equações é apresentada na seção seguinte.

2.3 Equação de Pressão

A abordagem utilizada para se obter a equação de pressão é eliminar a derivada temporal da saturação presente na Eq. (2.6). Primeiramente, as derivadas temporais são expandidas para se obter:

$$-\nabla \cdot (\rho_w \vec{v}_w) + q_w = \rho_w S_w \frac{\partial \phi}{\partial t} + \phi S_w \frac{d\rho_w}{dp_w} \frac{\partial p_w}{\partial t} + \phi \rho_w \frac{\partial S_w}{\partial t} \quad (2.7a)$$

$$-\nabla \cdot (\rho_o \vec{v}_o) + q_o = \rho_o S_o \frac{\partial \phi}{\partial t} + \phi S_o \frac{d\rho_o}{dp_o} \frac{\partial p_o}{\partial t} + \phi \rho_o \frac{\partial S_o}{\partial t} \quad (2.7b)$$

Então, dividindo a primeira equação por ρ_w e a segunda por ρ_o , levando em consideração as hipóteses feitas anteriormente de que fluidos e rochas são incompressíveis e finalmente adicionando as equações resultantes, obtém-se:

$$-\nabla \cdot \vec{v}_t + Q_t = \phi \frac{\partial (S_w + S_o)}{\partial t} \quad (2.8)$$

where $\vec{v}_t = \vec{v}_w + \vec{v}_o$ is the total velocity of the fluid and $Q_t = (q_w/\rho_w) + (q_o/\rho_o)$ is the

2.3. EQUAÇÃO DE PRESSÃO

total volumetric rate. Moreover, considering Eq. (2.3) and rearranging the terms, we obtain the pressure equation for the two-phase flow in porous media:

$$\nabla \cdot \vec{v}_t = Q_t \quad (2.9)$$

De modo a apresentar a Eq. (2.9) em relação a apenas uma única variável de pressão, pode-se definir uma pressão média por:

$$p_{avg} = \frac{p_w + p_o}{2} \quad (2.10)$$

Considerando a definição de pressão capilar como $p_c = p_o - p_w$, pode-se expressão as pressões das fases individuais como:

$$p_w = p_{avg} - \frac{p_c}{2} \quad (2.11a)$$

$$p_o = p_{avg} + \frac{p_c}{2} \quad (2.11b)$$

As mobilidades das fases são definidas como a relação entre a permeabilidade relativa e a viscosidade do fluido:

$$\lambda_\alpha = \frac{k_{r\alpha}}{\mu_\alpha} \quad (2.12)$$

Finalmente, reescrevendo a Eq. (2.9) utilizando pressões médias e capilares pode-se obter depois de um rearranjo dos termos a seguinte expressão:

$$\nabla \cdot \left(-K \left((\lambda_w + \lambda_o) \nabla p_{avg} + \frac{\lambda_w - \lambda_o}{2} \nabla p_c - (\lambda_w \rho_w + \lambda_o \rho_o) g \nabla z \right) \right) = Q_t \quad (2.13)$$

Logo, pode ser claramente observado que a Eq. (2.13), considerando as hipóteses feitas, tem uma natureza elíptica. A equação de saturação pode ser deduzida usando uma manipulação algébrica similar para completar o modelo acoplado de pressão-saturação para um escoamento bifásico água-óleo em meios porosos. O capítulo 3 apresenta métodos para resolver de forma aproximada este tipo de equação.

2.4 Equação de Saturação

Calculando a velocidade a partir do gradiente de pressão obtido no passo anterior, podemos então definir de modo explícito uma equação para a saturação.

$$\phi \frac{\partial S}{\partial t} + f_{,S} \mathbf{u} \cdot \nabla S + \nabla \cdot (\kappa \lambda_o f (\rho_w - \rho_o) \mathbf{g}) = 0 \quad (2.14)$$

a partir de (2.14) podemos derivar...

2.5 Sistema de Equações para Escoamento Bifásico

Capítulo 3

Formulação Numérica

A ideia básica de praticamente todos os métodos numéricos para a solução de EDP é a discretização de um problema contínuo com infinitos graus de liberdade (DOFs) e desse modo redução para um número finito de DOFs. Este problema discreto resulta em um sistema de equações com um número finito de variáveis, o qual normalmente pode ser resolvido utilizando um método adequado (ver seção 3.2 para detalhes) (Chen *et al.*, 2006).

No MDF, o qual ainda é o padrão na simulação numérica de reservatórios, as derivadas das equações originais são simplesmente substituídas por quocientes de diferenças. Os valores das variáveis são definidos apenas por um número específico de pontos, os quais podem estar localizados tanto nos cantos quanto no interior das células (Fortuna, 2000).

O processo de discretização no caso do MEF é diferente na medida em que ele requer a reformulação das EDPs em uma forma variacional equivalente. As variáveis desconhecidas, neste caso a pressão e a saturação, são descritas em cada ponto da região usando uma função de interpolação. Os pontos de suporte para estas funções são os nós da malha, os quais são conectados para formar os elementos.

Neste trabalho, foram experimentados tanto o clássico Método de Elementos Finitos de Galerkin quanto uma formulação dos Métodos dos Elementos Finitos Mistos para resolver a equação de pressão 2.3. Estas metodologias são discutidas nas seções subsequentes e referências são citadas para informações adicionais.

3.1 Método dos Elementos Finitos de Galerkin

Antes de especificar a formulação numérica utilizada, algumas definições a respeito das condições de contorno e hipóteses simplificadoras têm que ser discutidas.

3.1. MÉTODO DOS ELEMENTOS FINITOS DE GALERKIN

Neste trabalho, foi feita a consideração usual adotada em problemas de reservatório de petróleo (Peaceman, 1977; Ewing, 1983) de condição de fluxo zero em todas as fronteiras exteriores do domínio, isto é:

$$\lambda K \nabla p \cdot n = 0 \quad \text{in } \Gamma_N \quad (3.1)$$

Onde λ é a mobibilidade total, ou seja, $\lambda = \lambda_w + \lambda_o$, p é a pressão média ($p = p_{avg}$) e n é o vetor unitário normal aos contornos externos.

Os contornos são descritos como $\Gamma = \partial\Omega = \Gamma_I \cup \Gamma_P \cup \Gamma_D \cup \Gamma_N$, onde (Carvalho, 2005):

- Γ_I = Poços injetores;
- Γ_P = Poços produtores;
- Γ_D = Condição de contorno de Dirichlet (Variável prescrita);
- Γ_N = Condição de contorno de Neumann (Fluxo prescrito).

Usualmente, os poços são tratados como condições de contorno internas, sendo modelados através de métodos especiais para lidar com a velocidade alta nas adjacências de um poço (Peaceman, 1977) comparada com o resto do domínio. Apesar disso, foi adotada neste trabalho uma hipótese simplificador, considerando que os poços são termos fonte/sumidouros de produção ou injeção concentrados em um único nó da malha, isto é, uma função delta de Dirac neste ponto. Outrossim, foi considerada neste trabalho a hipótese de fluxo horizontal (sem influência da gravidade) e com pressão capilar negligenciável. Deste modo foi possível se concentrar nas características elípticas da equação de pressão.

A fim de resolver o problema utilizando o MEF, é necessário inicialmente expressar o mesmo na chamada *forma fraca*. O primeiro passo para obter esta nova formulação (também chamada de problema variacional) é multiplicar a EDP por uma função w , chamada de *função de teste*, em seguida o resultado é integrado no domínio Ω e finalmente uma integração por partes é efetuada nos termos com derivadas de segunda ordem (Hughes, 2000). A função desconhecida (pressão p , por exemplo) é chamada de *função tentativa*. Espaços de funções apropriados devem ser definidos tanto para função de teste quanto para a função tentativa.

O problema variacional para a Eq. (2.13) considerando todas as hipóteses mencionadas tem a seguinte forma:

3.1. MÉTODO DOS ELEMENTOS FINITOS DE GALERKIN

Problema 1 (Contínuo). Encontre $p \in P$ tal que:

$$A(p, w) = f(w) \quad \forall w \in W \quad (3.2)$$

com

$$P = \{p \in H^1(\Omega), p = p_D \text{ em } \Gamma\} \quad (3.3)$$

$$W = \{w \in H^1(\Omega), w = 0 \text{ em } \Gamma\} \quad (3.4)$$

$$A(p, w) = \int_{\Omega} \lambda K \nabla p \cdot \nabla w d\Omega \quad \forall p, w \in P, W \quad (3.5)$$

$$f(w) = \int_{\Omega} Q_t w d\Omega \quad \forall w \in W \quad (3.6)$$

onde p_D é o valor prescrito de p nos contornos de Dirichlet e H^1 é o Espaço de Sobolev de funções com derivadas de primeira ordem quadrado-integráveis. Isto significa que ao passo que a solução da EDP tem que existir em um espaço de funções com derivadas contínuas (*forma forte*), o Espaço de Sobolev exigido pela (*forma fraca*) permite funções com derivadas descontínuas (Langtangen, 2011). A existência e unicidade da solução deste problema é dada pelo lema de Lax e seu detalhamento completo pode ser encontrado em Garcia (1997).

Para a análise via elementos finitos deste problema variacional, é necessário transformar este formulção contínua definida pela Eq. (3.2) em um problema discreto. Considere o domínio Ω , o qual é discretizado por N elementos não sobrepostos E_i de modo que $\Omega_h = \bigcup_{i=1}^N E_i$ e $E_i \cap E_j = \emptyset, i \neq j$, onde o subíndice h denota um parâmetro de tamanho de malha que representa o problema discretizado.

O problema variacional discreto para a equação de pressão utilizando a abordagem clássica de Galerkin descrita em (Hughes, 2000) e adotada em (Wells *et al.*, 2008; Barbosa *et al.*, 2009) é:

Problema 2 (Discreto). Encontre $p_h \in P_h$ tal que:

$$A(p_h, w_h) = f(w_h) \quad \forall w_h \in W_h \quad (3.7)$$

com

$$P_h = \{p_h \in H^1(\Omega_h), p_h \in \mathbf{P}^k(E_i), p_h = p_d \text{ em } \Gamma\} \quad (3.8)$$

$$W_h = \{w_h \in H^1(\Omega_h), w_h \in \mathbf{P}^k(E_i), w_h = 0 \text{ em } \Gamma\} \quad (3.9)$$

$$A(p_h, w_h) = \int_{\Omega_h} \lambda K \nabla p_h \cdot \nabla w_h d\Omega \quad \forall p_h, w_h \in P_h, W_h \quad (3.10)$$

$$f(w_h) = \int_{\Omega_h} Q_t w_h d\Omega \quad \forall w_h \in W_h \quad (3.11)$$

onde $\mathbf{P}^k(E_i)$ define funções de forma de Lagrange com ordem k para o elemento finito E_i . A Eq. (3.7) aplicada a uma malha resulta em um conjunto de equações discretas na variável p , cuja solução é obtida usando os métodos descritos na seção 3.2.

3.2 Multigrid

A ideia básica dos métodos Multigrid é acelerar a solução de sistemas de equações em uma malha fina usando correções calculadas em uma malha grosseira. A motivação para esta abordagem vem da observação de que o comportamento do erro da solução numérica no domínio da frequência. Erros de alta frequência, os quais estão associados à variações locais da solução, são bem resolvidos por métodos iterativos convencionais (Gauss-Seidel, Gradientes Conjugados, etc.). Entretanto, erros de baixa frequência, associados à variação global das soluções, são muito mais insensíveis a esses métodos.

Um esquema Multigrid começa amortecendo os erros de alta frequência associados à solução inicial na malha fina, usando algum tipo de método de relaxação (usualmente um método iterativo que seria usado isoladamente). Uma vez atingido este objetivo inicial, efetuar mais iterações iria apenas resultar em uma pior convergência. Sendo assim, o resíduo é transferido para uma malha grosseira, onde os modos de baixa frequência da malha fina se comportam com de alta frequência, sendo portanto facilmente eliminados usando algum tipo de método direto, no caso da malha ser grosseira o suficiente, ou mesmo usando o mesmo método de relaxação aplicado na malha mais fina. As correções são então calculadas na malha grosseira e transferidos de volta para a malha fina de modo a atualizar a solução. Este procedimento pode ser aplicado recursivamente em uma sequência de malhas cada vez mais grosseiras, a fim de que cada nível nesta hierarquia seja responsável pela eliminação de uma faixa de frequência de erro (Briggs *et al.*, 2000).

Existem dois tipos principais de método Multigrid: o Multigrid Geométrico (GMG), o qual trabalha diretamente nas malhas que representam o domínio discreto, e o Multigrid Algébrico (AMG), o qual opera apenas no sistema de equações lineares resultantes da aplicação de uma formulação numérica, não tendo portanto uma interpretação geométrica direta.

A maior motivação para o desenvolvimento do AMG foi a necessidade de métodos robustos e flexíveis para acelerar a convergência sem a exigência de calibração fina para cada problema, como é frequentemente o caso para métodos GMG, o qual requer atenção

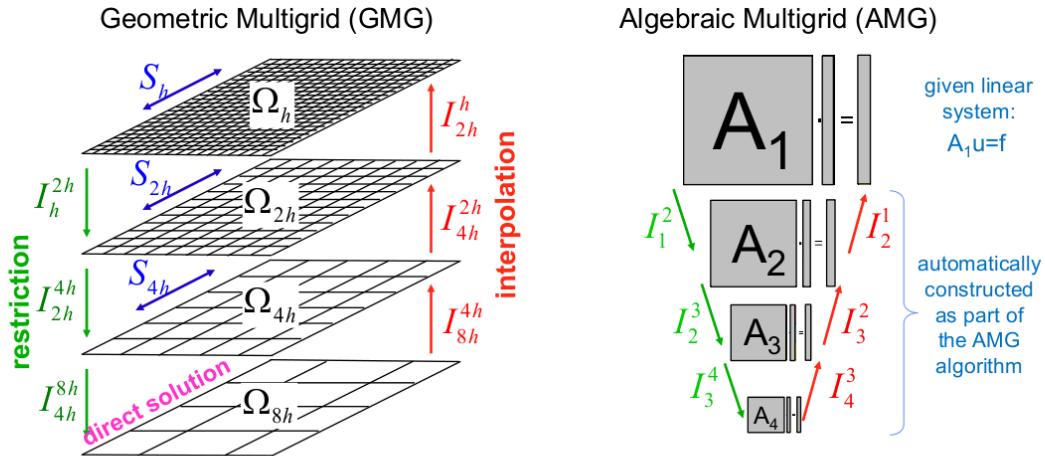


Figura 3.1 Comparação dos métodos Multigrid Geométrico e Algébrico (retirado de Trottenberg *et al.* (2001)).

especial na geração da sequência de malhas grosseiras, principalmente na presença de geometrias complexas (Trottenberg *et al.*, 2001). Esta robustez é alcançada porque o AMG conta com um processo totalmente automático de engrossamento que pode agir apenas nas direções nas quais a relaxação irá efetivamente suavizar o erro para o problema dado, enquanto o GMG requer uma hierarquia fixa pré-determinada antes de começar o ciclo. A Fig. 3.1 apresenta uma comparação gráfica destes dois métodos.

Matematicamente, ambas as abordagens podem ser descritas aproximadamente do mesmo modo, é apenas necessário considerar que malhas e nós no AMG são sistemas de equações lineares e variáveis destes sistemas, respectivamente. Portanto, o método Multigrid pode ser brevemente descrito como segue:

Considere o sistema de equações no nível mais refinado:

$$A_f p_f = f_f \quad (3.12)$$

onde A_f é a matriz resultante do procedimento descrito na seção 3.1 e o subíndice f representa valores no nível refinado. Depois de efetuadas algumas iterações com um método de relaxação, o erro estará suavizado e seu resíduo r_h será:

$$A_f \hat{p}_f - f_f = r_h \quad (3.13)$$

onde \hat{p}_f é a solução estimada para a variável de interesse. Subtraindo a Eq. (3.13) da Eq.

(3.12) resulta:

$$A_f p_f - A_f \hat{p}_f = -r_h \quad (3.14)$$

e se o operador A_h for linear, esta equação pode ser apresentada como a equação de correção:

$$A_f \Delta p_f = -r_h \quad (3.15)$$

Assumindo, como já mencionado, que os modos de alta frequência do erro foram amortecidos depois da relaxação no nível mais refinado, o erro reminiscente irá conter basicamente modos de baixa frequência que devem ser resolvidos, mas de modo a alcançar tal objetivo é necessário restringir tais modos ao nível grosso, onde os mesmos irão se comportar como modos de alta frequência, ou seja:

$$A_c \Delta p_c = -I_f^c r_f \quad (3.16)$$

Na Eq. (3.16) I_f^c representa o operador de restrição, o qual transfere valores do resíduo do nível refinado f para o nível grosso c . Uma vez que a Eq. 3.16 é resolvida adequadamente (frequentemente um resolvedor direto será suficiente, já que este nível apresenta menores exigências de processamento e armazenamento), sua solução pode ser interpolada de volta para o nível refinado usando um operador de prolongamento I_c^f e a aproximação original no nível refinado pode então ser corrigida:

$$\hat{p}_f^* = \hat{p}_f + I_c^f \Delta p_c \quad (3.17)$$

onde \hat{p}_f^* é a solução atualizada no nível refinado. É importante observar que este procedimento é intrinsecamente recursivo, podendo o sistema no nível grosso ser visto como um novo nível refinado, sendo suavizado novamente até que seja atingido um nível no qual as contribuições das correções nos níveis grossos seja trazidas de volta para os seus respectivos níveis refinados. Existem diversos tipos de esquemas para lidar com esta estratégia, entre eles estão os ciclos V, W e F. A Fig. 3.2 mostra uma representação gráfica destes ciclos em um esquema Multigrid de 4 níveis, ver Briggs *et al.* (2000) ou Trottenberg *et al.* (2001) para uma explicação mais detalhada de cada um destes ciclos.

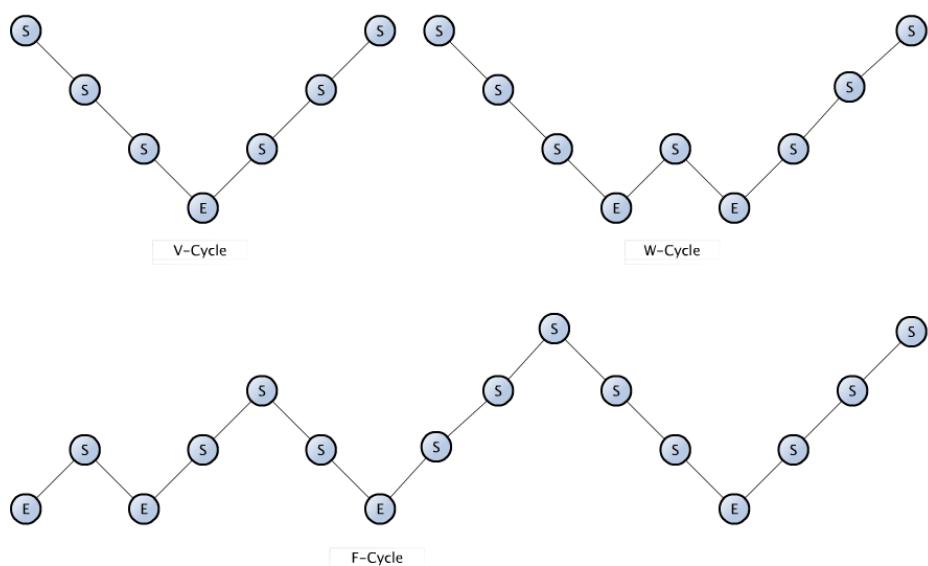


Figura 3.2 Ciclos V, W e F. S representa suavização e E é a solução no nível mais grosso.

Capítulo 4

Implementação Computacional

Todas as formulações, tanto matemática quanto numéricas, utilizadas neste trabalho apresentam grandes desafios, não apenas no que tange a compreensão teórica, mas também em relação ao desenvolvimento propriamente dito do software, já que é de fundamental importância uma implementação adequada destes métodos a fim de se atingir um compromisso vantajoso entre generalidade e performance computacional, pois este dois objetivos frequentemente tendem a conduzir as diretrivas de codificação do programa para direções diferentes.

Este capítulo se propõe a apresentar o modo como o programa de computador resultante deste trabalho foi elaborado, sendo que para isto foi adotada uma descrição *top-down*, ou seja, primeiramente são expostas as características gerais do programa e em seguida as divisões do mesmo em diferentes blocos que permitem uma melhor abordagem do problema. Cada um destes blocos é então subdividido em partes menores, as quais correspondem a tarefas específicas, que podem ter suas soluções diretamente codificadas utilizando uma linguagem de programação ou então serem resolvidas através de alguma biblioteca de software disponível na internet.

Outro aspecto a ser comentado a respeito da filosofia de desenvolvimento do programa é justamente que se buscou sempre utilizar soluções já prontas disponíveis publicamente, de modo a evitar o fenômeno popularmente conhecido como "reinventar a roda". Além disso, foi dada preferência explícita a pacotes publicados segundo licenças livres (e.g., GNU GPL, BSD License, etc.), as quais permitem o uso, estudo, adaptação e redistribuição de programas de modo bastante transparente. Logicamente, o resultado deste trabalho também está disponível através de uma licença livre. No Apêndice A se encontra um manual com instruções para a obtenção, instalação e uso do software desenvolvido.

4.1 Estrutura Geral do Programa

A resolução de um problema através de um método numérico pode normalmente ser dividida em três etapas distintas:

- *Pré-processamento*: Definição de uma geometria que aproxime o domínio real e geração de uma malha de pontos discretos para a mesma. Além disso, nesta etapa são fornecidas as condições de contorno e propriedades que buscam representar o problema real. Também nesta fase são definidos os parâmetros da análise efetuada no passo a seguir.
- *Processamento*: Cálculo propriamente dito dos valores para as variáveis de interesse. Para isto, é utilizada alguma formulação numérica específica que trate adequadamente a descrição matemática do problema. Os resultados obtidos nesta etapa são analisados criticamente na fase seguinte.
- *Pós-processamento*: Visualização e interpretação dos resultados fornecidos pela etapa de processamento. Caso a resposta obtida seja satisfatória, a resolução do problema é encerrada neste ponto. Caso contrário, alterações são feitas na etapa de pré-processamento e o ciclo é reiniciado até que se atinja o objetivo.

Esta divisão também foi considerada no desenvolvimento do *Delfine*, sendo este o nome do programa desenvolvido neste trabalho. Os blocos responsáveis por executar cada uma das tarefas descritas anteriormente serão apresentados na forma de fluxograma e detalhados nas seções seguintes.

4.2 Pré-Processamento

Na Fig. 4.1 pode ser visto um fluxograma representando os vários passos necessários para a obtenção, a partir de um problema real, de um problema discreto passível de ser analisado numericamente.

Inicialmente, é necessário por parte do usuário ter uma descrição o mais precisa possível do problema de interesse. De posse da mesma, parte-se para as etapas de discretização da geometria (i.e., geração da malha) e de definição de um arquivo de entrada de dados.

O *Delfine* oferece três caminhos para a obtenção da malha: através de um gerador interno do *FEniCS/Dolfin*, o qual é utilizado principalmente na parte de processamento,

porém disponibiliza algumas rotinas básicas de geração de malhas, com a vantagem de tornar o programa independente de qualquer programa externo para este fim, já que os parâmetros para definição de malha são definidos no próprio arquivo de entrada de dados da simulação. Porém, como grande desvantagem podemos citar o fato da limitação quanto às geometrias disponíveis, pois apenas formas primitivas como linhas, retângulos, círculos, paralelepípedos e esferas podem ser descritos usando esta ferramenta.

Uma segunda alternativa é o uso da ferramenta *Gmsh* (Geuzaine and Remacle, 2009), a qual apresenta como vantagem uma maior flexibilidade na definição da geometria, pois este gerador disponibiliza várias operações que podem ser executadas em formas primitivas para a obtenção de outras mais complexas. Entre estas operações podemos citar adição, subtração, extrusão, escalonamento, divisão, entre outras. Além disso, o *Gmsh* dispõe de uma meta-linguagem própria que possibilita a escrita de scripts para a automatização e parametrização da geração de malhas. Como desvantagem, temos a necessidade de conversão do arquivo no formato *.msh gerado pelo *Gmsh* para o padrão utilizado pelo *Delfine*, o qual é derivado diretamente do formato utilizado pelo *FEniCS/Dolfin*. Esta conversão é feita utilizando o script *delfine-convert*.

Como terceira e última alternativa temos o uso de outros gerador de malhas disponíveis publicamente ou não, como por exemplo o *Triangle*, o *Medit*, o *ExodusII* ou o pré-processador do *Abaqus*. Como vantagem desta alternativa podemos citar a liberdade em relação ao tipo de gerador de malhas a ser utilizado, já que o usuário pode escolher aquele com o qual tem mais familiaridade. Porém, os arquivos nos formatos de saída de qualquer um dos programas citados terá que ser convertido para o formato padrão do *Delfine* utilizando o script *dolfin-convert* (Logg and Wells, 2010).

A diferença básica entre a segunda e a terceira alternativas reside exatamente no tipo de script utilizado para a conversão dos arquivos de malha. O *dolfin-convert* é disponibilizado como parte da família de pacotes *FEniCS/Dolfin*, porém o mesmo faz apenas uma conversão das informações geométricas e topológicas da malha, ignorando às informações extras eventualmente presentes nos arquivos. Logo, não é possível importar *flags* de condições de contorno definidos no *Triangle* diretamente no *Delfine*, por exemplo. Sendo assim, tais informações têm que ser adicionadas manualmente aos arquivos de entrada do *Delfine*.

Já o *delfine-convert* é uma adaptação do *dolfin-convert* realizada durante este trabalho com o objetivo de importar todas as indicações de condições de contorno definidas no *Gmsh* e exportá-las no formato lido pelo *Delfine*. Esta funcionalidade é de fundamental importância para problemas de maior complexidade, pois permite agilizar bastante a

4.2. PRÉ-PROCESSAMENTO

etapa de pré-processamento. Por isto, dentre toda as citadas, a segunda alternativa foi a mais utilizada ao longo deste trabalho.

Uma vez obtida a geometria discretizada, é necessário ler o arquivo de entrada de dados, o qual contém informações a respeito das condições de contorno, propriedades de rochas e fluidos, parâmetros numéricos, etc. De um modo geral, tais informações podem ser fornecidas através de um arquivo de texto comum, desde que as mesmas estejam ordenadas de modo estruturado para serem processadas pelo programa.

Entretanto, tal abordagem não apresenta uma robustez adequada, pois permite que pequenos erros do usuário na confecção do arquivo de dados passem desapercebidos, o que pode acarretar tanto em demora para executar uma análise inicial, pois se torna necessário uma checagem manual de todos os parâmetros fornecidos até se encontrar a fonte de erro, como também se permite executar análises com valores não consistentes, os quais podem vir a gerar resultados totalmente não-físicos, podendo inclusive levar o usuário a interpretar o fenômeno de interesse de maneira errônea.

Sendo assim, de modo a aumentar a robustez da entrada de dados se optou pelo uso de arquivos estruturados no formato *Extensible Markup Language* (XML) (Bray *et al.*, 2000) com a utilização da linguagem de especificação de esquemas *Relax NG Compact Syntax* (RNC) (Clark, 2001). Esta linguagem permite definir um padrão lógico a ser seguido por qualquer arquivo XML gerado, caso contrário o mesmo não é considerado válido e o ponto exato onde o erro na entrada de dados foi encontrado é apresentado ao usuário antes mesmo de qualquer análise ter início. Este padrão é definido através de um arquivo chamado de *Schema Grammar* e todo arquivo de entrada será checado contra esta gramática através do programa *Relax NG Validator* (RNV) (Sheen, 2007). Este programa é sempre chamado automaticamente pelo *Delfine* antes do início da análise para verificar a entrada de dados do usuário, garantindo assim que a execução só será realizado caso exista uma consistência mínima nos dados fornecidos. Na Listagem 1 é apresentado um trecho de um arquivo de entrada típico. No mesmo pode ser observada a estrutura hierárquica utilizada para o armazenamento dos dados, os quais são tratados de modo completamente modular, ou seja, caso uma análise não precise de determinada informação, o bloco de dados referente a mesma pode simplesmente ser deixado de fora do arquivo de dados, sem prejuízo na etapa de pré-processamento.

XML

```

<define>
  <geometry>
    <mesh dimension="2" order ="1" type="gmsh">
      <filename>HomoIsoBCStruct.msh</filename>
    </mesh>
    <boundary-conditions>
      <well function="inject" id="301">.250</well>
    </boundary-conditions>
  </geometry>
  <physical>
    <rock-properties>
      <rock-type id="1">
        <porosity>1.0</porosity>
        <permeability type="per-domain">
          <Kxx>0.50</Kxx>
          <Kxy>0.0</Kxy>
          <Kxz>0.0</Kxz>
        </permeability>
      </rock-type>
    </rock-properties>
  </physical>
</define>
```

Listagem 1: Trecho de arquivo de entrada no formato *.xml.

4.3 Processamento

Uma vez lidos na etapa de pré-processamento os arquivos necessário para a execução da análise, tem início a etapa de processamento, a qual foi implementada neste trabalho de acordo com os fluxogramas apresentados nas Figs. 4.2 e 4.5.

O fluxograma da Fig. 4.2 representa os passos necessário para resolver a parte elíptica do problema de escoamentos multifásicos em meios porosos. As formulações matemática e numérica deste problema adotadas neste trabalho podem ser consultadas nas seções 2.3 e 3.1, respectivamente.

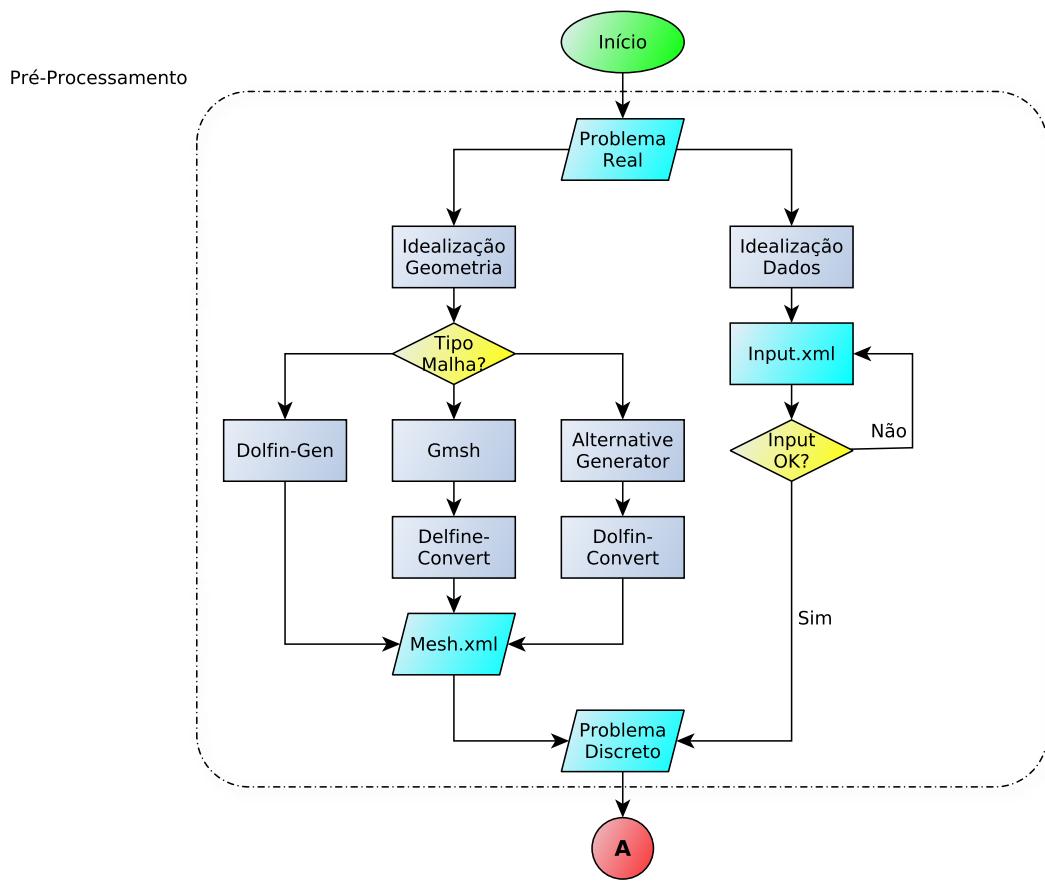


Figura 4.1 Fluxograma da etapa de pré-processamento.

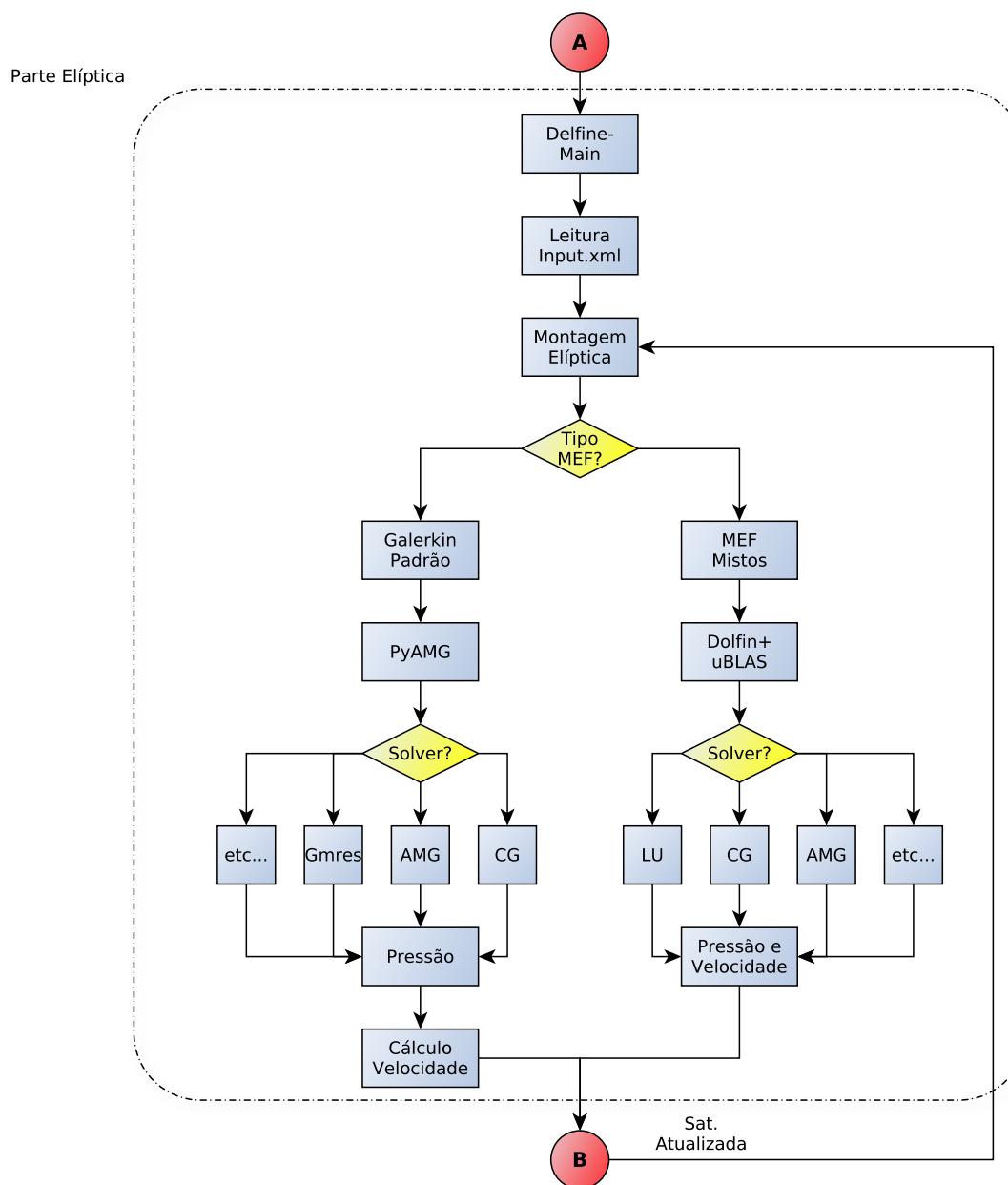


Figura 4.2 Fluxograma da resolução da parte elíptica.

O usuário tem a disposição duas alternativas de métodos numéricos para resolução do problema: o Método dos Elementos Finitos (MEF) e o Método dos Elementos Finitos Mistos (MEFM). As diferenças entre os dois métodos do ponto de vista da formulação numérica são discutidas no capítulo 3, sendo nesta seção discutidos apenas os aspectos de implementação. Ambos os métodos são codificados utilizando a interface na linguagem *Python* da ferramenta *FEniCS/DOLFIN*. A seguir faremos uma descrição geral deste pacote computacional e em seguida mostraremos como o mesmo foi utilizado neste trabalho.

4.3.1 FEniCS

O *FEniCS* é um projeto colaborativo em código aberto iniciado em 2003 com o objetivo automatizar a solução de modelos matemáticos baseados em equações diferenciais (Logg *et al.*, 2011), tendo todos os seus componentes sido desenvolvidos buscando generalidade, eficiência e simplicidade.

De um modo geral, o desenvolvedor tem acesso direto principalmente ao *DOLFIN*, o qual é uma biblioteca que permite a interface com o usuário através de diversas classes acessíveis via programas em *C++* ou em *Python*. Para facilitar a compreensão, a Fig. 4.3 apresenta de modo esquemático a sequência na qual os diversos componentes do projeto *FEniCS* são executados, e como eles interagem para permitir a resolução do problema.

Inicialmente, o problema tem que ser descrito matematicamente na sua forma variacional (ou fraca). Em seguida, esta deve ser implementada utilizando a *Unified Form Language* (UFL) (Logg *et al.*, 2011), a qual é uma linguagem específica de domínio para declaração da discretização via MEF de formas variacionais e funcionais. Para o caso de programas escritos em *Python*, a descrição via UFL é embutida dentro do próprio script, já no caso do *C++* é necessário criar um arquivo externo para definição da forma fraca do problema, sendo o mesmo importado para o programa principal.

Em seguida, tais formas são compiladas utilizando o *FEniCS Form Compiler* (FFC) (Logg *et al.*, 2011), o qual é o responsável de fato pela geração automática do código otimizado em linguagem de baixo-nível. Este código estará automaticamente conforme o padrão do *Unified Form-assembly Code* (UFC) e pode ser acessado de modo transparente através de classes da biblioteca *DOLFIN*, a qual será responsável pela montagem de todos os tensores necessários para a resolução numérica do problema dentro do programa definido pelo usuário (*Delfine* no caso deste trabalho, como representado na Fig. 4.3).

O *DOLFIN* (Logg and Wells, 2010) automatiza a montagem dos sistemas lineares ou não-lineares provenientes da discretização via MEF de EDPs expressas na forma

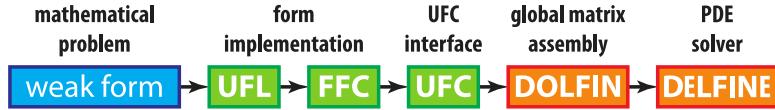


Figura 4.3 Interação entre os diversos componentes do projeto *FEniCS* para definição do problema, seguidos pela resolução no *Delfine* (adaptado de Rathgeber (2010)).

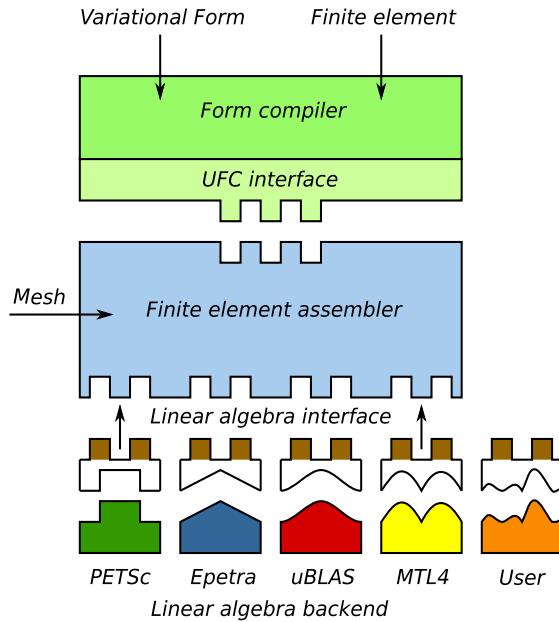


Figura 4.4 Estrutura modular do *DOLFIN* (retirado de Logg and Wells (2010)).

variacional. Na Fig. 4.4 é apresentada a estrutura modular da biblioteca, onde os dados de entrada para um problema específico são a malha, a forma variacional e os tipos de elementos finitos adotados.

```

Python
def Galerkin(self, delfineVar, parameter):
    .
    .
    # Define variational form
    a = inner(grad(v), K*mob*grad(u))*dx
    L = v*f*dx - g*v*ds
    
```

Listagem 2: Montagem do operador elíptico via MEF de Galerkin usando o *FEniCS-DOLFIN*.

```
    _____ Python _____  
def MixedFEM(self, delfineVar, parameter):  
    .  
    .  
    .  
    # Define variational form  
    a = (dot((invK/mob)*sigma, tau)  
          - div(tau)*u - div(sigma)*v)*dx  
    L = - f*v*dx
```

Listagem 3: Montagem do operador elíptico via MEFM usando o *FEniCS/DOLFIN*.

A parte parabólica/hiperbólica do problema pode ser descrita pelo fluxograma apresentado na Fig. 4.5.

4.4 Pós-Processamento

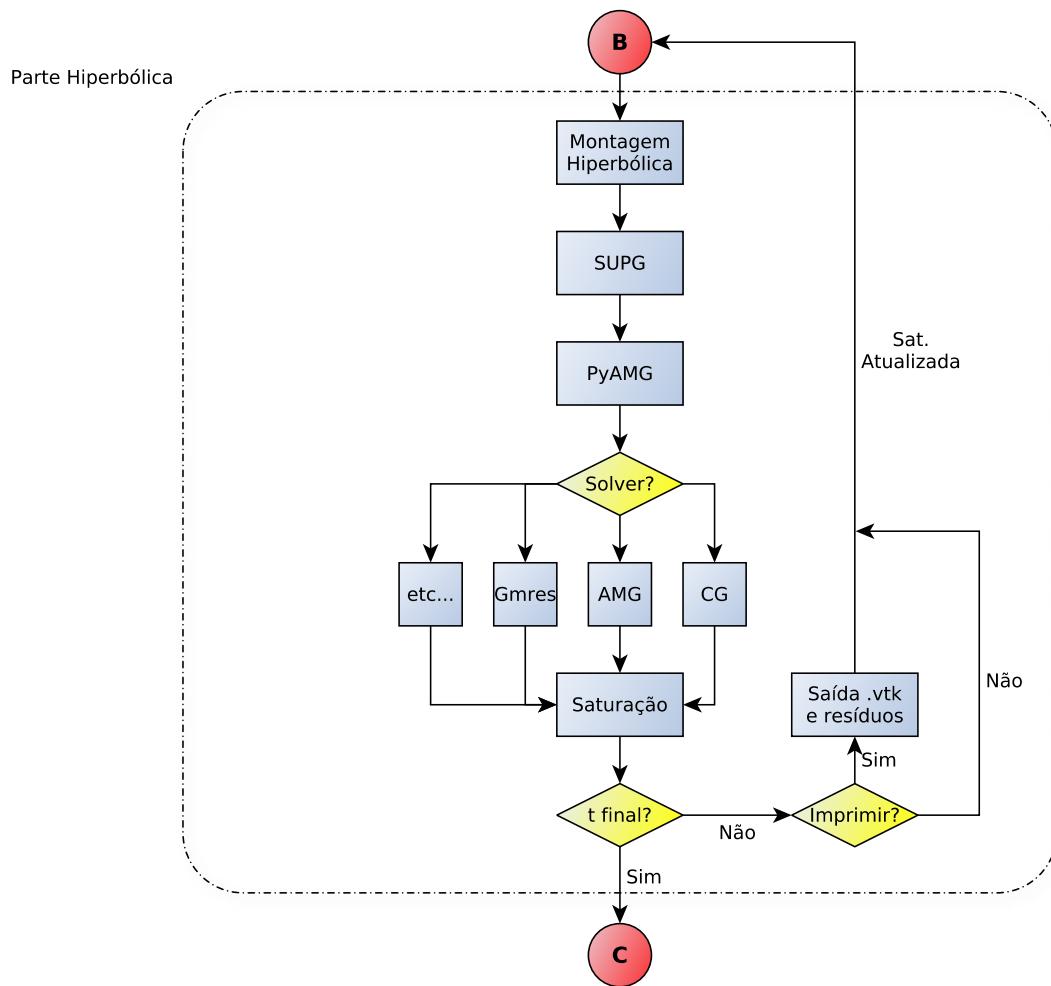


Figura 4.5 Fluxograma da resolução da parte hiperbólica.

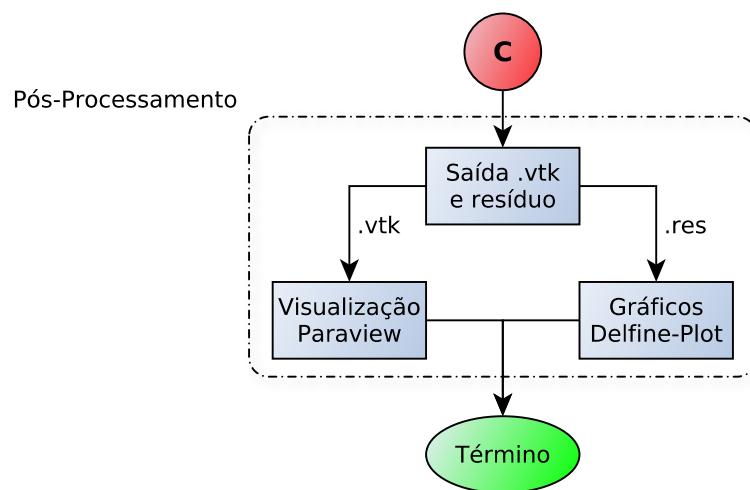


Figura 4.6 Fluxograma da etapa de pós-processamento.

Capítulo 5

Resultados

Neste capítulo serão apresentados exemplos e resultados obtidos utilizando a metodologia e o programa computacional descritos nos capítulos anteriores.

5.1 Problemas Elípticos

Esta seção tem por objetivo avaliar os métodos utilizados para a resolução da equação de pressão em meios porosos, a qual para o estado estacionário pode ser considerada uma típica equação elíptica. Devido a essa característica, a metodologia utilizada pode ser analisada utilizando-se de problemas modelos que compartilham das mesmas propriedades matemáticas. Uma particularidade comum aos problemas regidos por equações elípticas é que todo o domínio de interesse Ω é afetado por qualquer mudança no valor da variável em um ponto qualquer no interior de Ω , ou em sua fronteira $\delta\Omega$ (Fortuna, 2000). Isto significa que uma perturbação em um ponto irá ter influência sobre todo o domínio, sendo que a mesma diminui com o aumento da distância em relação ao ponto originador de tal perturbação. Sendo assim, tais problemas tendem em geral a apresentar soluções suaves ao longo do domínio (como contra-exemplo poderíamos citar casos com alta anisotropia). Além da equação de pressão, outros exemplos de aplicação deste tipo de equação são o cálculo do potencial elétrico, da difusão de calor em uma chapa metálica e de escoamentos incompressíveis, inviscidos e irrotacionais (também conhecidos como escoamentos potenciais).

A equação normalmente utilizada como modelo para testar metodologias de resolução de equações elípticas pode ser descrita como:

$$\nabla(\mathbf{K}\nabla u) = f \quad (5.1)$$

5.1. PROBLEMAS ELÍPTICOS

Em coordenadas cartesianas e considerando um coeficiente anisotrópico e variável no espaço $\mathbf{K}(x, y, z)$, esta equação pode ser representada por:

$$\frac{\partial}{\partial x} \left(\mathbf{K} \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left(\mathbf{K} \frac{\partial u}{\partial y} \right) + \frac{\partial}{\partial z} \left(\mathbf{K} \frac{\partial u}{\partial z} \right) = f(x, y, z) \quad (5.2)$$

Nas seções seguintes, iremos resolver equações do tipo (5.2) considerando diversas possibilidades para o coeficiente \mathbf{K} de modo a demonstrar a flexibilidade da metodologia utilizada para lidar com meios homogêneos, heterogêneos, iso- e anisotrópicos. Vários parâmetros de interesse serão avaliados a partir dos resultados obtidos, entre eles:

- Erro da solução numérica quando comparada com a solução analítica, a qual para problemas simples é facilmente encontrada.
- Taxas de convergências para uma sequência de malhas sucessivamente refinadas.
- Evolução do resíduo para diferentes estratégias de resolução do sistema de equações lineares.

5.1.1 Meio Homogêneo e Isotrópico

Este primeiro exemplo foi originalmente proposto em (Chen *et al.*, 2006), sendo resolvido utilizando dois métodos diferentes, o CVFA (*Control Volume Function Approximation*) e o CVFE (*Control Volume Finite Element*). Este problema também foi explorado em (Silva, 2008) utilizando duas variações do Método dos Volumes Finitos baseado em Arestas (EBFV1 e EBFV2). Este exemplo é um problema de valor de contorno que pode ser representado do modo a seguir:

$$\nabla(\mathbf{K}\nabla u) = 2\pi^2 \cos(\pi x) \cos(\pi y) \quad \text{em} \quad \Omega = \{(x, y) \mid 0 < x < 1 \text{ e } 0 < y < 1\} \quad (5.3)$$

Onde \mathbf{K} é uma matriz simétrica e diagonal representada por:

$$\mathbf{K} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad (5.4)$$

Este problema apresenta condições de contorno periódicas nas fronteiras inferior e superior, já as fronteiras laterais estão sujeitas a uma condição de fluxo zero. Matematicamente:

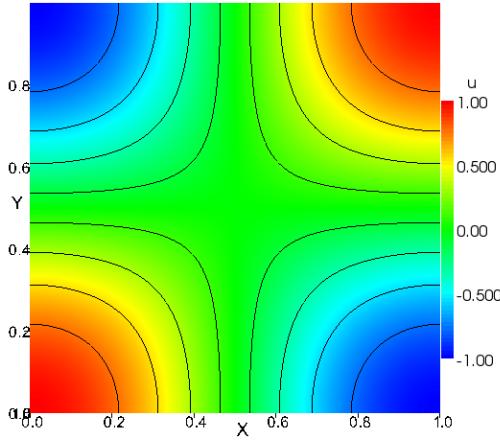


Figura 5.1 Campo escalar obtido utilizando o MEF com malha de 64x64 discretizada por triângulos lineares.

mente tais condições de contorno podem ser definidas do seguinte modo:

$$\begin{aligned} u &= \cos(\pi x) && \text{para } 0 < x < 1 \text{ e } y = 0 \\ u &= -\cos(\pi x) && \text{para } 0 < x < 1 \text{ e } y = 1 \\ \nabla u \cdot \vec{n} &= 0 && \text{para } 0 < y < 1 \text{ e } x = 0, 1 \end{aligned} \quad (5.5)$$

Resolvemos o problema elíptico descrito utilizando a formulação detalhada no capítulo 3, considerando diferentes malhas estruturadas sucessivamente refinadas e formadas por elementos triangulares de 1^a e 2^a ordem. O uso de malhas estruturadas se deu apenas para fins de comparação de erros e obtenção de ordens de convergência, já que a metodologia utilizada é perfeitamente capaz de lidar com malhas não-estruturadas bi- e tridimensionais. Para avaliação da acurácia do método utilizado, os resultados foram comparados com a solução analítica deste problema, a qual é representada pela função $u(x, y) = \cos(\pi x) \cos(\pi y)$. A Fig. 5.1 mostra o campo escalar u obtido para a malha de 64x64 elementos triangulares lineares, a qual já permite obter uma excelente concordância com a solução analítica.

Para uma sequência de malhas formadas por triângulos lineares, se espera uma taxa de convergência de segunda ordem, enquanto para uma sequência formada por triângulos quadráticos espera-se uma convergência de terceira ordem (Hughes, 2000). De modo a verificar o comportamento do método utilizado, resolvemos o problema proposto

Tabela 5.1 Erro e taxa de convergência obtidos neste trabalho para a solução da equação elíptica em meio homogêneo e isotrópico.

(a) Triângulo Linear

N	$\ E_{max}\ $	q_{max}
8	2.5e-02	—
16	6.4e-03	1.97
32	1.6e-03	1.99
64	4.0e-04	1.99
128	1.0e-04	2.00
256	2.5e-05	1.99

(b) Triângulo Quadrático

N	$\ E_{max}\ $	q_{max}
8	6.4e-04	—
16	8.7e-05	2.87
32	1.1e-05	2.94
64	1.4e-06	2.98
128	1.8e-07	2.98
256	2.2e-08	3.00

Tabela 5.2 Erro e taxa de convergência obtidos em Chen *et al.* (2006) e Silva (2008) para a solução da equação elíptica em meio homogêneo e isotrópico.

(a) CVFA(Chen *et al.*, 2006)

N	$\ E_{rms}\ $	q_{rms}
8	1.2e-02	—
16	3.0e-03	2.02
32	7.4e-03	2.01
64	1.8e-04	2.00

(b) EBFV1(Silva, 2008)

N	$\ E_{rms}\ $	q_{rms}
8	6.9-03	—
16	1.4e-03	2.29
32	3.2e-04	2.13
64	7.7e-05	2.06

utilizando elementos de 1^a (tabela 5.1(a)) e 2º ordem (tabela 5.1(b)), comparando entre si os resultados obtidos e também com outros disponíveis na literatura para elementos lineares utilizando outros tipos de formulação (tabelas 5.2(a) e 5.2(b)) (Chen *et al.*, 2006; Silva, 2008). De um modo geral, as taxas de convergências obtidas se aproximaram bastante das taxas teóricas tanto para o caso de elementos triangulares quanto quadráticos. A comparação com os resultados da literatura ficam um pouco prejudicadas por terem os erros sido calculados através de normas diferentes (norma do máximo neste trabalho, e RMS na literatura citada). Porém, é possível observar um comportamento coerente do erro obtido, pois com elementos quadráticos foram obtidos erros consideravelmente menores do que com os métodos de segunda ordem CVFA e EBFV1. Por outro lado, ao utilizar-se elementos lineares foram obtidos erros maiores porém na mesma ordem de grandeza daqueles obtidos na literatura.

Para a resolução do sistema de equações lineares provenientes da discretização do problema foram testadas três diferentes alternativas:

- Métodos dos Gradientes Conjugados (CG) aplicado isoladamente.
- Método do Multigrid Algébrico (AMG) aplicado isoladamente.

5.1. PROBLEMAS ELÍPTICOS

- Multigrid Algébrico aplicado como pré-condicionador para o método dos Gradientes Conjugados (AMG+CG).

Para a realização dos testes, foi utilizada uma malha de 32x32 elementos triangulares de 1^a ordem. Para todos os casos foi considerada como critério de parada uma tolerância $\epsilon = 10^{-10}$, com um número máximo de 200 iterações. Para os casos nos quais o método AMG foi utilizado como forma de acelerar a convergência, considerou-se ciclos do tipo V em uma hierarquia de 4 “malhas” sucessivamente mais grosseiras. É importante apenas lembrar que o método AMG não depende de malhas grosseiras propriamente ditas, apenas das matrizes que representam o sistema de equações, sendo tais matrizes manipuladas algebricamente de modo a se obter os níveis mais grosseiros (Trottenberg *et al.*, 2001).

Como pode ser observado na Fig. 5.2, os esquemas utilizando o método multigrid (AMG e AMG+CG) apresentaram resultados bastante superiores em relação ao método CG, o qual utiliza apenas um único sistema de equações lineares (ou seja, uma única “malha”). Enquanto o método AMG+CG reduz o resíduo em aproximadamente uma ordem de grandeza por iteração, o método CG necessita de aproximadamente 10 vezes mais iterações para reduzir o resíduo na mesma proporção. É interessante comparar também os resultados obtidos ao utilizar-se o AMG isoladamente e ao usá-lo como um pré-condicionador para o método dos Gradientes Conjugados. Percebe-se que nas primeiras iterações os dois métodos apresentam fatores de convergência para o resíduo praticamente idênticos, porém a medida que se prossegue com as iterações, o AMG usado isoladamente tende a apresentar uma piora na taxa de convergência, enquanto o método que utiliza a combinação AMG+CG mantém o excelente fator de convergência obtido inicialmente.

Isto se deve a um aspecto bastante observado na literatura (Trottenberg *et al.*, 2001; Oosteele and Washio, 1998; Mavriplis, 2001) no qual um único ou poucos autovalores da matriz representante do operador de iteração Multigrid estão com um valor muito acima daquele obtido para os demais autovalores. Sendo que o autovalor máximo (também chamados de raio espectral) do operador de iteração indicam em última análise o fator de convergência assimptótica para o problema (Briggs *et al.*, 2000), limitando portanto a taxa de convergência possível de ser obtida. Um modo de contornar tal problema é justamente utilizar algum método de subespaço de Krylov, como é o caso do método CG, os quais têm a característica de em poucas iterações amortecer os autovetores relacionados aos poucos autovalores elevados (Trottenberg *et al.*, 2001).

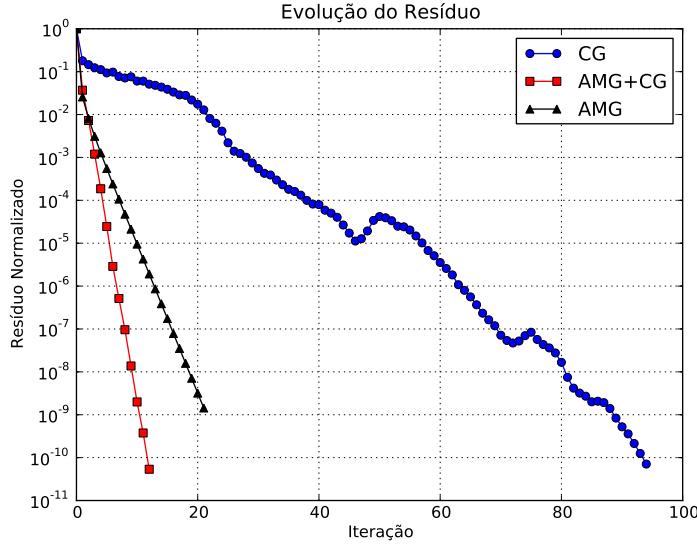


Figura 5.2 Caso homogêneo e isotrópico. Comparação da evolução dos resíduos para malha 32x32 discretizada por triângulos lineares.

5.1.2 Meio Homogêneo e Anisotrópico

Este problema foi originalmente proposto em Crumpton (1995), onde foi resolvido utilizando um método de volumes finitos centrado na célula (FCFV), tendo também sido estudado em Carvalho (2005) e Silva (2008) utilizando o já mencionado método dos volumes finitos baseado em arestas (EBFV).

Neste segundo exemplo, buscamos explorar a flexibilidade da metodologia utilizada, a qual permite considerar tensores diagonais ou cheios para representar o coeficiente \mathbf{K} . O problema proposto pode ser descrito através da equação (5.6) aplicada em um domínio $\Omega = (0, 1) \times (0, 1)$.

$$\nabla(\mathbf{K}\nabla u) = -2(1 + x^2 + xy + y^2)e^{xy} \quad (5.6)$$

Onde \mathbf{K} é uma matriz simétrica e não-diagonal representada por:

$$\mathbf{K} = \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \quad (5.7)$$

A definição deste problema está completa ao definir-se a condição de fronteira em todo o contorno $\delta\Omega$ como $u(x, y) = e^{xy}$.

Na Fig. 5.3 pode ser observado o campo escalar para a variável u , considerando uma discretização do domínio Ω com uma malha de 64x64 elementos triangulares de 1ª

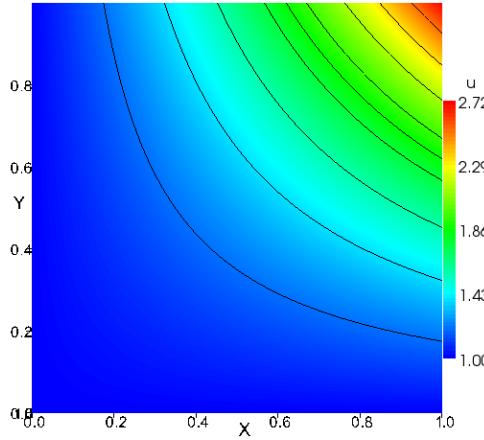


Figura 5.3 Caso homogêneo e anisotrópico. Campo escalar obtido utilizando o MEF com malha de 64x64 discretizada por triângulos lineares.

ordem.

De modo a comparar os resultados obtidos com aqueles da literatura, calculamos o erro segundo a norma do máximo e as taxas de convergências para o mesmo considerando uma sequência de malhas sucessivamente refinadas (ver tabela 5.3(a)). Tais resultados foram comparados com aqueles obtidos tanto pelo FCFV quanto pelo EBFV (ver tabelas 5.4(a) e 5.4(b)).

Pode-se observar pelo resultados obtidos que novamente o MEF se comportou como esperado, apresentando uma taxa de convergência de ordem 2 para os elementos lineares. Os valores absolutos do erro na norma do máximo se comparam de maneira ligeiramente

Tabela 5.3 Erro e taxa de convergência obtidos neste trabalho para a solução da equação elíptica em meio homogêneo e anisotrópico.

(a) Triângulo Linear

N	$\ E_{max}\ $	q_{max}
8	2.1e-03	—
16	5.3e-04	2.01
32	1.3e-04	2.00
64	3.3e-05	1.99
128	8.3e-06	2.00
256	2.1e-06	2.00

5.1. PROBLEMAS ELÍPTICOS

Tabela 5.4 Erro e taxa de convergência obtidos em Crumpton (1995) e Carvalho (2005) para a solução da equação elíptica em meio homogêneo e anisotrópico.

(a) FCFV(Crumpton, 1995)

N	$\ E_{rms}\ $	q_{rms}
8	1.2e-03	—
16	2.9e-04	2.00
32	7.3e-05	1.99
64	1.8e-05	1.99

(b) EBFV(Carvalho, 2005)

N	$\ E_{max}\ $	q_{max}
8	3.0e-03	—
16	1.0e-03	1.58
32	3.4e-04	1.80
64	8.9e-05	1.94

favorável aqueles apresentado em Carvalho (2005).

Assim como para o primeiro exemplo, também neste fizemos um estudo a respeito da performance de diferentes métodos iterativos para a resolução do sistema de equações lineares provenientes da discretização do problema. Cinco diferentes alternativas foram testadas:

- Métodos dos Gradientes Conjugados (CG) aplicado isoladamente.
- Métodos do Resíduo Mínimo Generalizado (GMRES) aplicado isoladamente.
- Método do Multigrid Algébrico (AMG) aplicado isoladamente.
- Multigrid Algébrico aplicado como pré-condicionador para o método dos Gradientes Conjugados (AMG+CG).
- Multigrid Algébrico aplicado como pré-condicionador para o método do Resíduo Mínimo Generalizado (AMG+GMRES).

É importante mencionar que o método GMRES é normalmente utilizado para problemas envolvendo matrizes não-simétricas (Saad, 2003), tendo sido avaliado apenas para verificar a flexibilidade do resolvelor de sistemas de equações lineares, já que a discretização deste problema resulta em uma matriz auto-adjunta (e portanto simétrica se considerarmos apenas termos reais).

Para a realização dos testes, foi utilizada uma malha de 64x64 elementos triangulares de 1^a ordem. Para todos os casos foi considerada como critério de parada uma tolerância $\epsilon = 10^{-10}$, com um número máximo de 200 iterações. Em todos os casos nos quais o método AMG foi utilizado como forma de acelerar a convergência, considerou-se ciclos do tipo V em uma hierarquia de 4 “malhas” sucessivamente mais grosseiras. Um gráfico mostrando a evolução do resíduo normalizado para todos os métodos testados pode ser observado na Fig. 5.4, onde fica claro novamente a grande superioridade dos métodos

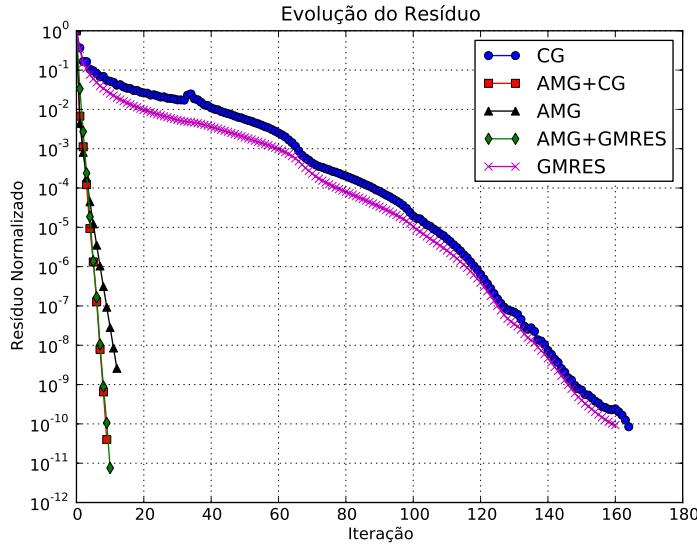


Figura 5.4 Caso homogêneo e anisotrópico. Comparação da evolução dos resíduos para malha 64x64 discretizada por triângulos lineares.

que utilizam aceleração de convergência via multigrid em comparação com os outros métodos. É interessante notar também que para este exemplo os métodos CG e GMRES tiveram desempenhos bastante próximos, tanto isoladamente quanto em conjunto com o AMG enquanto pré-condicionador.

5.1.3 Meio Heterogêneo e Anisotrópico

Neste exemplo foi considerado um meio heterogêneo e anisotrópico. Este caso foi proposto em (Crumpton, 1995), tendo sido também analisado em vários outros trabalhos (Hyman *et al.*, 1997; Carvalho, 2005; Silva, 2008). O domínio Ω é um quadrado $[-1, 1] \times [-1, 1]$ com condições de contornos de Dirichlet dadas pela solução exata. O coeficiente \mathbf{K} é definido por:

$$\mathbf{K} = \begin{cases} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} & \text{se } x \leq 0 \\ \alpha \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} & \text{se } x > 0 \end{cases} \quad (5.8)$$

onde o fator α é um parâmetro utilizado para definir a intensidade de descontinuidade em $x = 0$.

A solução analítica (e condição de Dirichlet para os contornos) é dada por:

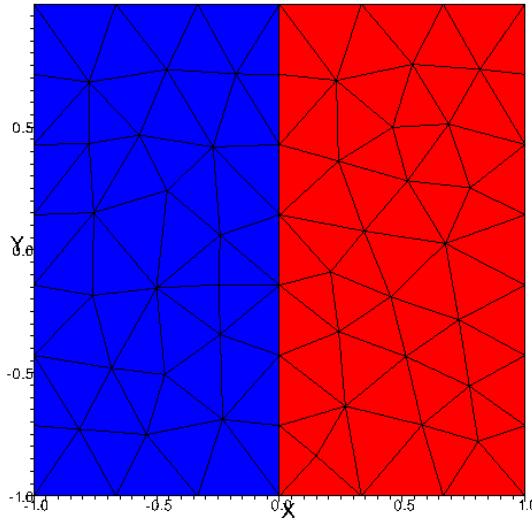


Figura 5.5 Domínio utilizado e malha não-estruturada com $N \simeq 8$ e heterogeneidade em $x = 0$.

$$u(x, y) = \begin{cases} [2 \sin(y) + \cos(y)]\alpha x + \sin(y) & \text{se } x \leq 0 \\ \alpha \exp(x) \sin(y) & \text{se } x > 0 \end{cases} \quad (5.9)$$

Por fim, podemos definir o também descontínuo termo fonte como:

$$f(x, y) = \begin{cases} [-2 \sin(y) - \cos(y)]\alpha x - \sin(y) & \text{se } x \leq 0 \\ 2\alpha \exp(x) \cos(y) & \text{se } x > 0 \end{cases} \quad (5.10)$$

Para a resolução deste problema e obtenção de taxas de convergência, consideramos uma sequência de malhas sucessivamente refinadas com dimensões aproximadas de $(8x8)$, $(16x16)$, $(32x32)$, $(64x64)$ e $(128x128)$. A malha não-estruturada utilizada para $N \simeq 8$ e a divisão dos subdomínios em $x = 0$ pode ser observada na Fig. 5.5.

Pode ser observado na Fig. 5.6 o efeito da descontinuidade na linha $x = 0$, efeito este que se torna mais pronunciado ao aumentarmos o valor do parâmetro α . Os valores para os erros da solução numérica quando comparada com a analítica para as normas do máximo e $L2$ são apresentados na tabela 5.5, onde pode ser visto que um aumento na descontinuidade do tensor \mathbf{K} para os diferentes sub-domínios causa um aumento nos valores absolutos dos erros obtidos.

Um fato interessante que pode ser constatado a partir dos valores obtidos para os erros

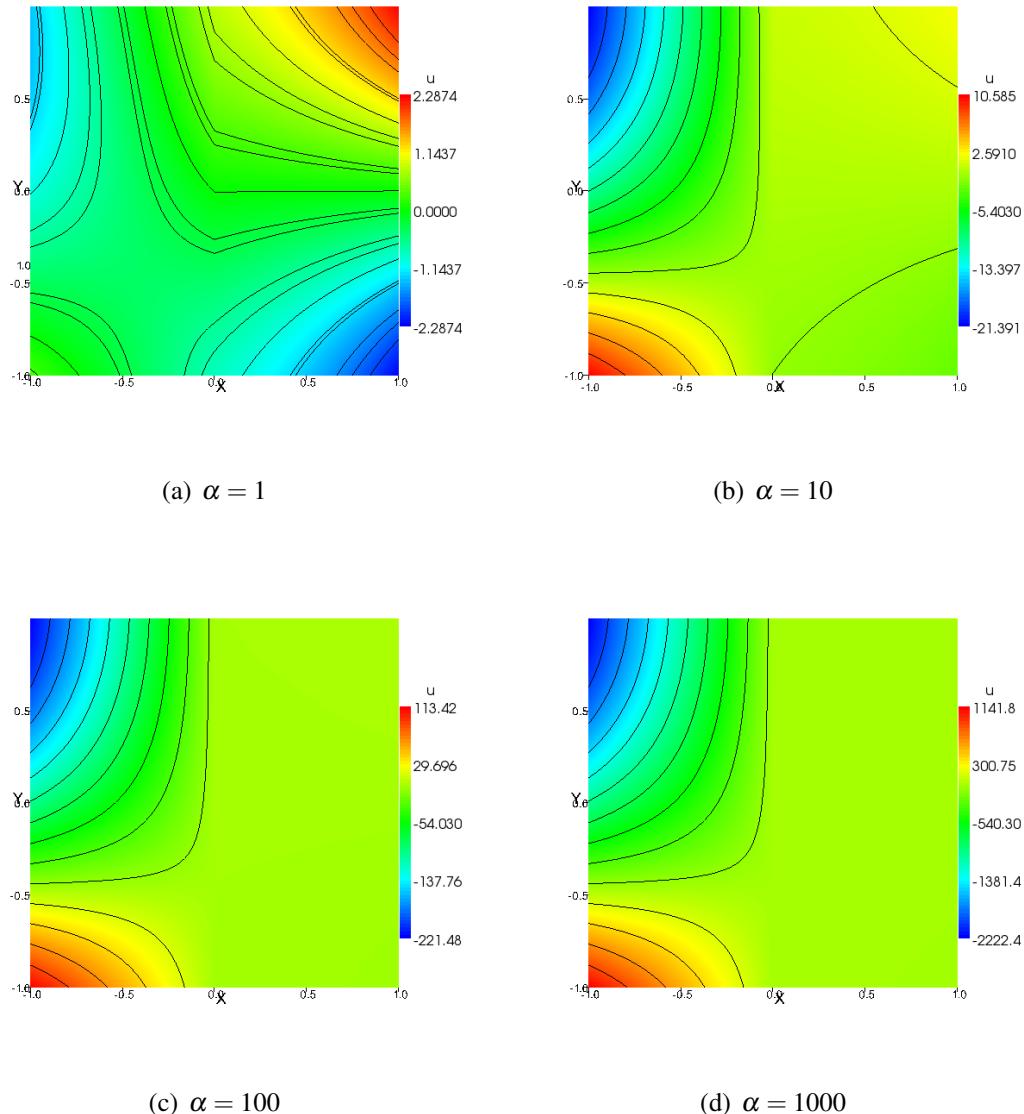


Figura 5.6 Campo escalar obtido para diferentes fatores de descontinuidade α utilizando o MEF com malha de 64x64 discretizada por triângulos lineares.

Tabela 5.5 Erro e taxa de convergência obtidos neste trabalho para a solução da equação elíptica em meio heterogêneo e anisotrópico.

(a) $\alpha = 1$

N	$ E_{max} $	q_{max}	$ E_{L2} $	q_{L2}
8	5.4e-02	—	4.5e-02	—
16	2.7e-02	0.99	2.0e-02	1.1
32	1.3e-02	1.02	9.8e-03	1.05
64	6.6e-03	1.02	4.8e-03	1.02
128	3.2e-03	1.02	2.4e-03	1.00

(b) $\alpha = 10$

N	$ E_{max} $	q_{max}	$ E_{L2} $	q_{L2}
8	9.2e-02	—	1.8e-01	—
16	3.8e-02	1.3	5.2e-02	1.78
32	1.9e-02	1.02	1.8e-02	1.52
64	9.2e-03	1.02	7.7e-03	1.24
128	4.6e-03	1.00	3.6e-03	1.08

(c) $\alpha = 100$

N	$ E_{max} $	q_{max}	$ E_{L2} $	q_{L2}
8	8.6e-01	—	1.7e-00	—
16	2.9e-01	1.59	4.3e-01	1.98
32	8.8e-02	1.70	1.1e-01	1.98
64	2.5e-02	1.79	2.8e-02	1.94
128	7.2e-03	1.81	7.9e-03	1.84

(d) $\alpha = 1000$

N	$ E_{max} $	q_{max}	$ E_{L2} $	q_{L2}
8	8.8e-00	—	1.7e+01	—
16	2.9e-00	1.58	4.3e-00	1.98
32	9.2e-01	1.68	1.1e-00	2.0
64	2.7e-01	1.74	2.7e-01	1.99
128	7.9e-02	1.79	6.8e-02	1.99

5.2. PROBLEMAS DE ESCOAMENTOS BIFÁSICOS EM MEIOS POROSOS

da solução numérica obtida é que os mesmos apresentam uma taxa de convergência em torno de 1 para $\alpha = 1$ e 10 , sendo que esta taxa se aproxima de 2 quando aumentamos o fator de descontinuidade ($\alpha = 100$ e 1000). Ao compararmos tais resultados com a literatura (Crumpton, 1995; Hyman *et al.*, 1997; Carvalho, 2005; Silva, 2008) percebemos que os métodos apresentados nos referidos trabalhos que fazem uma avaliação local dos fluxos tendem a obter uma convergência de ordem 2, enquanto métodos que trabalham apenas com a variável nodal (neste caso, $u(x,y)$) apresentam ordem 1 para o erro, tanto na norma do máximo quanto para a $L2$.

Um aspecto a ser notado que pode contribuir para a compreensão deste comportamento é a distribuição do erro ao longo do domínio para diferentes valores de α . Na Fig. 5.7 pode ser visto que o erro varia suavemente para $\alpha = 1$ e $\alpha = 10$, sendo que o mesmo está concentrado na região de descontinuidade em $x = 0$. Já para $\alpha = 100$ e $\alpha = 1000$, o erro para a malha de $N \simeq 64$ apresenta modos de frequência muito mais alta, estando também concentrado em um dos lados do domínio.

Este comportamento pode ser explicado pelo fato de que para maiores valores de α temos uma intensidade maior de descontinuidade para o mesmo tamanho de malha, o que pode ser visto como equivalente a manter o valor do coeficiente de descontinuidade e engrossar a malha, levando portanto os componentes do erro a se apresentarem como de mais alta frequência, independente da magnitude dos mesmos (Trottenberg *et al.*, 2001).

Para a resolução do sistema de equações lineares resultante da discretização foi novamente verificado o desempenho para diferentes tipos de métodos, mesmo sabendo de antemão que este problema é representado por uma matriz simétrica positiva-definida, sendo portanto um candidato natural a ser resolvido usando métodos do tipo gradiente conjugado, seja isoladamente ou com o multigrid algébrico como pré-condicionador.

Os resultados obtidos para a evolução dos resíduos podem ser verificados na Fig. 5.8, onde mais uma vez é constatada a superioridade dos métodos que utilizam multigrid, mesmo para este problema que apresenta sub-domínios com propriedades diferentes.

5.2 Problemas de Escoamentos Bifásicos em Meios Porosos

Ao se considerar também a equação de saturação, o problema do deslocamento de fluidos em meios porosos passa a ter também um componente altamente hiperbólico.

5.2. PROBLEMAS DE ESCOAMENTOS BIFÁSICOS EM MEIOS POROSOS

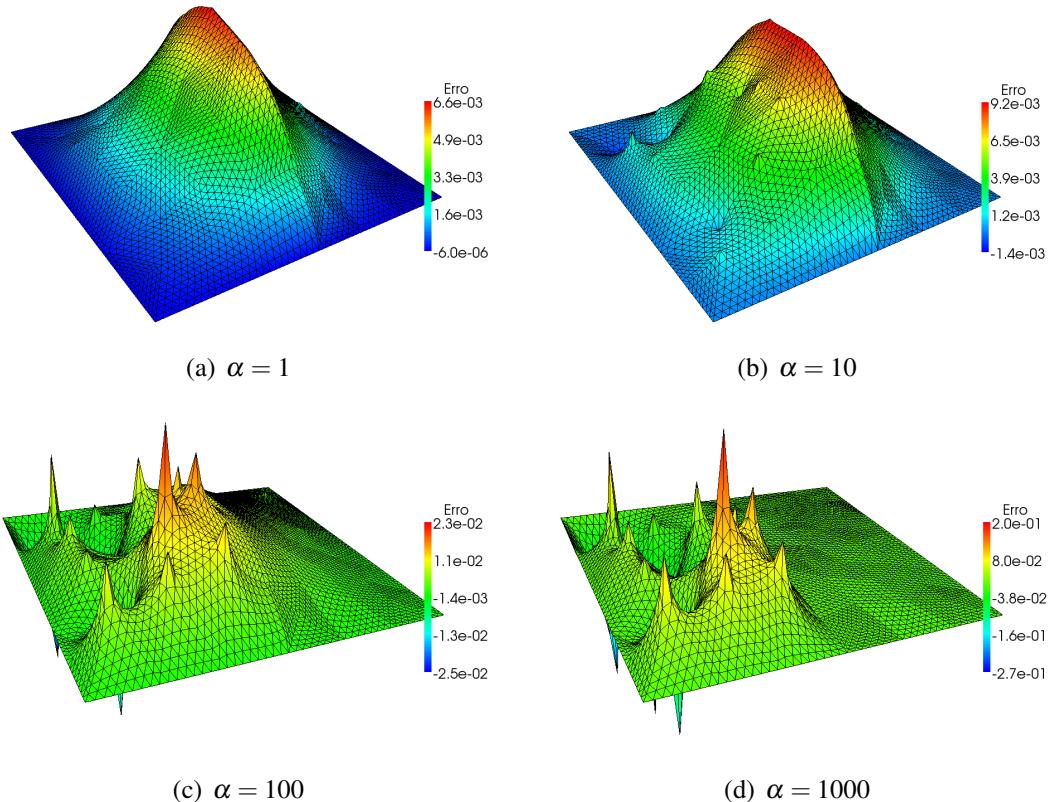


Figura 5.7 Caso heterogêneo e anisotrópico. Distribuição do erro obtido para malha com $N \simeq 64$ discretizada por triângulos lineares.

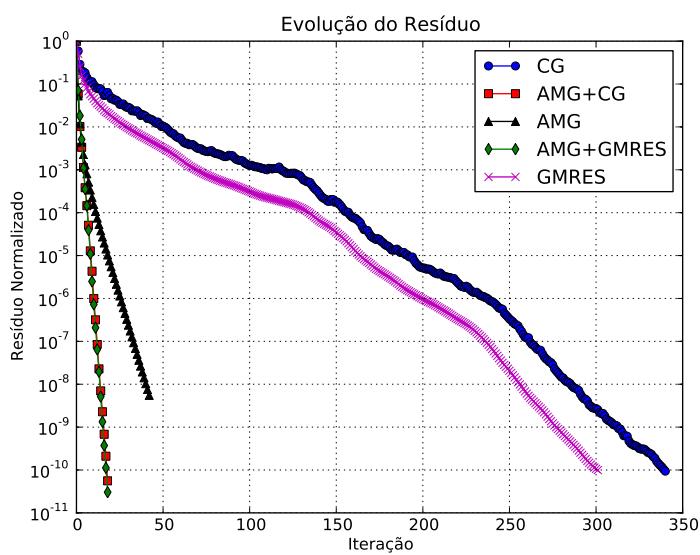


Figura 5.8 Caso heterogêneo e anisotrópico. Comparaçao da evolução dos resíduos para malha 64x64 discretizada por triângulos lineares e com $\alpha = 1$.

5.2. PROBLEMAS DE ESCOAMENTOS BIFÁSICOS EM MEIOS POROSOS

5.2.1 1/4 de Cinco Poços Homogêneo

5.2.2 1/4 de Cinco Poços Heterogêneo

Capítulo 6

Conclusões e Trabalhos Futuros

6.1 Conclusões

In this work, a procedure for the solution of the pressure equation in heterogeneous and anisotropic porous media was presented. The PDEs resultant from the mathematical modeling were solved using a Standard Galerkin Finite Element Method approach. In order to accelerate the convergence of the resultant system of equations, an Algebraic Multigrid method was employed, both as stand-alone and as a preconditioner to other solvers. As this kind of numerical framework normally imply in a great time-effort in developing the computational tools that implement them, this work used some available open source libraries in order to automate the generation of low-level code based on some high-level description of the problem. This showed to be a very efficient work methodology, as it allowed to experiment more with the problem in question in less time, because the developer spares time that would be used in "administrative" coding tasks. The results obtained for a model example showed the good performance of the proposed procedure, as the solution converged even for a heterogeneous and highly anisotropic problem.

6.2 Trabalhos Futuros

A future work, which is already in progress, will be the solution of the saturation equation associated with the pressure by means of the velocity. The pressure-velocity equation will be solved using a mixed FEM and the saturation equation using a Petrov-Galerkin FEM.

Referências Bibliográficas

- Aarnes, J., Gimse, T., and Lie, K.-A. (2007). *An Introduction to the Numerics of Flow in Porous Media using Matlab*, pages 265–306. Springer, Berlin.
- Aziz, K. and Settari, A. (1979). *Petroleum Reservoir Simulation*. Elsevier, Calgary.
- Barbosa, A., Catabriga, L., Valli, A., Malta, S., and Lima, L. (2009). Experiments using a finite element formulation of incompressible miscible displacement in porous media. In *32th Congresso Nacional de Matemática Aplicada e Computacional*, Cuiabá.
- Bear, J., Beljin, M., and Ross, R. (1992). Fundamentals of ground-water modeling. *EPA Ground Water Issue*, **13**(1), 1–11.
- Bell, W. N., Olson, L. N., and Schroder, J. (2008). Pyamg: Algebraic multigrid solvers in Python. Version 1.0.
- Bray, T., Paoli, J., Sperberg-McQueen, C., and Maler, E. (2000). Extensible Markup Language (XML). Version 1.0.
- Briggs, W., Henson, V. E., and McCormick, S. (2000). *A Multigrid Tutorial*. SIAM, Filadélfia.
- Carvalho, D. K. E. (2005). *Uma Formulação do Método dos Volumes Finitos com Estrutura de Dados por Aresta para a Simulação de Escoamentos em Meios Porosos*. Tese de Doutorado, Universidade Federal de Pernambuco (UFPE), Recife.
- Chavent, G. and Jaffre, J. (1986). *Mathematical Models and Finite Elements for Reservoir Simulation*. North Holland, Amsterdam.
- Chen, Z., Huan, G., and Ma, Y. (2006). *Computational Methods for Multiphase Flows in Porous Media*. SIAM, Filadélfia.
- Clark, J. (2001). Relax NG Specification. Version 1.0.

REFERÊNCIAS BIBLIOGRÁFICAS

- Cordazzo, J. (2006). *Simulação de Reservatórios de Petróleo Utilizando o Método EbFVM e Multigrid Algébrico*. Tese de Doutorado, Universidade Federal de Santa Catarina (UFSC), Florianópolis.
- Crumpton, P. (1995). Discretization and multigrid solution of elliptic equations with mixed derivative terms and strongly discontinuous coefficients. *Journal of Computational Physics*, **116**(1), 343–358.
- de Souza Menezes, D. (2009). *Utilização de Métodos de Transferência de Escala na Simulação de Recuperação de Hidrocarbonetos com Aplicação de Computação Distribuída*. Dissertação de Mestrado, Universidade Federal de Pernambuco (UFPE), Recife.
- Ewing, R. (1983). *The Mathematics of Reservoir Simulation*. SIAM, Philadelphia.
- Fortuna, A. (2000). *Técnicas Computacionais para Dinâmica dos Fluidos: Conceitos Básicos e Aplicações*. Edusp, São Paulo.
- Garcia, E. L. M. (1997). *Formulações de Elementos Finitos Bi e Tridimensionais para Simulação em Paralelo de Escoamentos em Reservatórios de Petróleo*. Tese de Doutorado, Universidade Federal do Rio de Janeiro (UFRJ), Rio de Janeiro.
- Geuzaine, C. and Remacle, J.-F. (2009). Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, **79**(11), 1309–1331.
- Hughes, T. (2000). *The Finite Element Method: Linear Static and Dynamic Finite Element Analysis*. Dover Publications, Nova Iorque.
- Hurtado, F. S. V. (2005). *Uma Formulação de Volumes Finitos Baseada em Elementos para a Simulação do Deslocamento Bifásico Imiscível em Meios Porosos*. Dissertação de Mestrado, Universidade Federal de Santa Catarina (UFSC), Florianópolis.
- Hyman, J., Shashkov, M., and Steinberg, S. (1997). The numerical solution of diffusion problems in strongly heterogeneous non-isotropic materials. *Journal of Computational Physics*, **132**(1), 130–148.
- Langtangen, H. (2011). *A FEniCS Tutorial*. Springer, Berlin.
- Logg, A. and Wells, G. (2010). Dolfin: Automated finite element computing. *ACM Transactions on Mathematical Software*, **37**(2), 417–444.

REFERÊNCIAS BIBLIOGRÁFICAS

- Logg, A., Mardal, K.-A., and Wells, G. (2011). *Automated Solution of Differential Equations by the Finite Element Method: The FEniCS Book*. Springer, Berlin.
- Marle, C. (1981). *Multiphase Flow in Porous Media*. Gulf Publishing, Paris.
- Matamoros, L. and Brueggemann, D. (2006). Simulation of the water and heat management in proton exchange membrane fuel cells. *Journal of Power Sources*, **161**(1), 203–213.
- Mavriplis, D. (2001). An assessment of linear versus non-linear multigrid methods for unstructured mesh solvers. *NASA/ICASE Technical Report*, **2001**(12).
- Oosteele, C. and Washio, T. (1998). On the use of multigrid as a preconditioner. In *Ninth International Conference on Domain Decomposition Methods*, Bergen.
- Peaceman, D. (1977). *Fundamentals of Numerical Reservoir Simulation*. Elsevier, Amsterdam.
- Peaceman, D. and Nash, S. (1990). *A History of Scientific Computing*. ACM Press, New York.
- Rathgeber, F. (2010). *Automated Finite Element Computation in the FEniCS Framework using General Purpose Graphics Processing Units*. Dissertação de Mestrado, Royal Institute of Technology (KTH), Estocolmo.
- Saad, Y. (2003). *Iterative Methods for Sparse Linear Systems*. SIAM, Philadelphia.
- Schneider, J. (2003). New least squares model used for development of permeability-porosity correlation. Technical report, Schneider Reservoir Engineering Consultant, Poteet.
- Sheen, D. (2007). Relax NG validator. Version 1.7.
- Silva, R. S. (2008). *Simulação de Escoamento Bifásico Oléo-Água em Reservatórios de Petróleo Usando Computadores Paralelos de Memória Distribuída*. Tese de Doutorado, Universidade Federal de Pernambuco (UFPE), Recife.
- Thomas, J. E. (2001). *Fundamentos de Engenharia de Petróleo*. Ed. Interciência, Rio de Janeiro.
- Trottenberg, U., Oosterlee, C., and Schüller, A. (2001). *Multigrid*. Elsevier, Oxford.

REFERÊNCIAS BIBLIOGRÁFICAS

Wells, G., Hooijkaas, T., and Shan, X. (2008). Modelling temperature effects on multi-phase flow through porous media. *Philosophical Magazine*, **88**(28), 3265–3279.

Apêndice A

Delfine - Manual do usuário

Manual do usuário do Delfine.

A.1 Instalação do Programa

Mostrar onde o programa pode ser baixado. Descrever dependências do mesmo. Focar em uma instalação do Ubuntu. Descrição da estrutura geral do programa.

A.2 Dados de Entrada

Descrição detalhada do arquivo .xml de entrada. Também descrever as possibilidades de geração de malha(gmsh com conversão, funções próprias do dolfin, uso de arquivo .xml do dolfin convertido a partir de outros formatos utilizando o convert-mesh, etc.)

A.3 Exemplo Detalhado

Mostrar como rodar um exemplo simples utilizado na tese.

Apêndice B

Delfine - Trechos de Código Selecionados

Trechos de código selecionados para exemplificar as formulações e métodos utilizados neste trabalho.

B.1 Montagem da Parte Elíptica

B.2 Cálculo das Velocidades

B.3 Montagem da Parte Hiperbólica

B.4 Montagem da Sistema de Equações Lineares