



COMPARAÇÃO ENTRE DUAS ABORDAGENS PARA SOLUÇÃO DE
SISTEMAS DE EQUAÇÕES RESULTANTES DA DISCRETIZAÇÃO PELO
MÉTODO IMPLÍCITO ADAPTATIVO

Jean Philippe François

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

Orientadores: Nelson Maculan Filho
Luiz Mariano Paes de Carvalho
Filho

Rio de Janeiro
Abril de 2017

COMPARAÇÃO ENTRE DUAS ABORDAGENS PARA SOLUÇÃO DE
SISTEMAS DE EQUAÇÕES RESULTANTES DA DISCRETIZAÇÃO PELO
MÉTODO IMPLÍCITO ADAPTATIVO

Jean Philippe François

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

Examinada por:

Prof. Nelson Maculan Filho, D.Sc.

Prof. Luiz Mariano Paes de Carvalho Filho, Ph.D.

Prof. Franklin de Lima Marquezino, D.Sc.

Prof. Paulo Goldfeld, D.Sc.

RIO DE JANEIRO, RJ – BRASIL

ABRIL DE 2017

François, Jean Philippe

Comparação entre duas abordagens para solução de sistemas de equações resultantes da discretização pelo método implícito adaptativo /Jean Philippe François. – Rio de Janeiro: UFRJ/COPPE, 2017.

XII, 48 p.: il.; 29, 7cm.

Orientadores: Nelson Maculan Filho

Luiz Mariano Paes de Carvalho Filho

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia de Sistemas e Computação, 2017.

Referências Bibliográficas: p. 46 – 48.

1. Simulação de reservatórios. 2. Método implícito adaptativo. 3. Pré-condicionadores. 4. Fatoração incompleta. 5. Ordenação natural. I. Maculan Filho, Nelson *et al.* II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia de Sistemas e Computação. III. Título.

*À minha esposa Luana e minha
filha Larissa, os grandes amores
da minha vida.*

Agradecimentos

Em primeiro lugar gostaria de agradecer a Deus pela minha vida, minha saúde e minha família.

À minha esposa e à minha filha que por muitos momentos relevaram e entenderam minha indisponibilidade para que eu pudesse me dedicar a esse trabalho.

Ao meu orientador Luiz Mariano que com sua insistência e perseverança nesta orientação fizeram com que esse trabalho pudesse ser executado com sucesso.

Aos meus gerentes e colegas da Petrobras que me incentivaram muito a concluir com sucesso essa etapa da minha vida.

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

COMPARAÇÃO ENTRE DUAS ABORDAGENS PARA SOLUÇÃO DE
SISTEMAS DE EQUAÇÕES RESULTANTES DA DISCRETIZAÇÃO PELO
MÉTODO IMPLÍCITO ADAPTATIVO

Jean Philippe François

Abril/2017

Orientadores: Nelson Maculan Filho

Luiz Mariano Paes de Carvalho Filho

Programa: Engenharia de Sistemas e Computação

Um dos pontos críticos de uma simulação de reservatório é o tempo de simulação, que, a depender do modelo, pode durar dias. Após séries de discretizações os sistemas lineares a serem resolvidos utilizam a fatoração incompleta como pré-condicionador. É comum a aplicação de um ordenamento à matriz a ser resolvida para melhorar a convergência e diminuir a quantidade de preenchimentos. Esse trabalho apresenta a comparação de dois pré-condicionadores utilizando a ordenação natural e a induzida aplicadas em simulações completas de campos reais utilizando o método implícito adaptativo. Os testes realizados apontaram que o pré-condicionador que utiliza a ordenação natural obteve speed-up de 2,46 quando comparado com as simulações totalmente implícitas para os mesmos campos enquanto o pré-condicionador que utiliza a ordenação induzida alcançou um speed-up de 1,77 em relação ao mesmo caso.

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

COMPARING TWO APPROACHES FOR SOLUTION SYSTEMS OF THE
EQUATIONS RESULTING FROM THE ADAPTATIVE IMPLICIT METHOD
DISCRETIZATION

Jean Philippe François

April/2017

Advisors: Nelson Maculan Filho

Luiz Mariano Paes de Carvalho Filho

Department: Systems Engineering and Computer Science

One critical point of a reservoir simulation is the total simulation time, that, depending on the model, may be days. After a series of discretizations the linear systems to be solved use incomplete factorization as preconditioning method. It is usual the application of matrix ordering to help convergence and decrease the amount of fill-in. In this work, we present the comparison between two preconditioners using the natural and induced orderings when applied to complete simulations of real cases using the adaptative implicit method. The results showed that, when compared to fully implicit simulation cases, the natural ordering preconditioner had a 2.46 speed-up, whereas the induced preconditioner had a 1.77 speed-up.

Sumário

Lista de Figuras	ix
Lista de Tabelas	xi
Lista de Símbolos	xii
1 Introdução	1
2 Métodos de Solução	4
2.1 FIM - Fully Implicit Method	5
2.1.1 Formulação	5
2.2 IMPES - Implicit Pressure, Explicit Saturation	9
2.2.1 Formulação	10
2.3 AIM - Adaptative Implicit Method	12
3 Ordenamento em um Pré-Condicionador Incompleto de Cholesky	14
3.1 Análise para ICHOL(0)	16
3.2 Análise para ICHOL(1)	20
3.3 Observações adicionais	22
3.4 Experimentos numéricos	24
4 Implementação	28
4.1 Blocos de Tamanhos Híbridos	28
4.2 Implementação das Estruturas de Blocos de Tamanhos Híbridos	32
5 Testes e Resultados para a Estrutura Híbrida	35
5.1 Resultados Numéricos	36
5.2 Resultados de Tempo de Execução	41
6 Conclusões	44
Referências Bibliográficas	46

Lista de Figuras

1.1	Malhas 4 x 4 ordenadas com a ordenação natural e a ordenação induzida. As células em verde foram marcadas como FIM.	2
1.2	Estrutura da matriz utilizando a ordenação natural e a ordenação induzida.	2
3.1	Grafo correspondente e matriz de esparsidade para uma malha 5 x 5 com ordenação natural.	16
3.2	Grafo correspondente e matriz de esparsidade para uma malha 5 x 5 com ordenação induzida.	16
3.3	Grafo e padrão de esparsidade associados à parte L da fatoração ICHOL(0) de uma malha 5 x 5 com ordenação natural.	17
3.4	Grafo e padrão de esparsidade associados à parte L da fatoração ICHOL(0) de uma malha 5 x 5 com ordenação induzida.	18
3.5	Padrão de esparsidade dos elementos não-zero para a inversa verdadeira.	19
3.6	Padrão de esparsidade dos elementos não-zero para a inversa utilizando a ordenação natural.	19
3.7	Padrão de esparsidade dos elementos não-zero para a inversa utilizando a ordenação induzida.	20
3.8	Representação de um grafo de uma malha regular mostrando o caminho entre dois nós causado por um preenchimento.	21
3.9	Grafo e padrão de esparsidade associados à parte L da fatoração ICHOL(1) de uma malha 5 x 5 com ordenação natural.	21
3.10	Grafo e padrão de esparsidade associados à parte L da fatoração ICHOL(1) de uma malha 5 x 5 com ordenação induzida.	22
3.11	Padrão de esparsidade dos elementos não-zero para a inversa utilizando a ordenação induzida utilizando preenchimento de nível 1.	23
3.12	Grafos de uma malha 6 x 6 com um grupo de células FIM cercadas por células IMPES.	23
3.13	Padrão de distribuição das células FIM e IMPES utilizada nos experimentos numéricos.	26

4.1	Matriz esparsa 3×3 com blocos de tamanho 2×2 e os respectivos vetores A, COLIDX e ROWIDX.	29
4.2	Matriz esparsa com seu formato estendido do BCRS para tratar blocos de tamanhos híbridos.	30

Lista de Tabelas

3.1	Número de iterações realizadas pelo <i>pcg</i> para $\text{ICHOL}(0)$	25
3.2	Número de iterações realizadas pelo <i>pcg</i> para $\text{ICHOL}(1)$	26
3.3	Número de elementos não-zero para o fator L do $\text{ICHOL}(1)$	27
5.1	Campos utilizados para executar os testes computacionais e o resultado das simulações.	37
5.2	Dados numéricos para os casos executados com sucesso.	37
5.3	Médias utilizadas para se comparar os dois pré-condicionadores.	38
5.4	Dados numéricos para os mesmos casos da Tabela 5.2 para as simulações FIM.	39
5.5	Médias das iterações não-lineares por passo de tempo e iterações lineares por iterações não-lineares para as simulações FIM.	39
5.6	Dados numéricos para o caso C5 EOS.	40
5.7	Percentual utilizado pela fatoração simbólica no tempo do solver linear em simulações com pré-condicionador padrão e simulações FIM.	41
5.8	Tempos totais de simulação e solver linear para os pré-condicionadores híbrido e padrão.	42
5.9	Razão entre os tempos dos pré-condicionadores padrão e híbrido.	42
5.10	Informação de tempo das simulações FIM.	43

Lista de Símbolos

A	Área da seção transversal para fluxo
B	Fator volume de formação
K	Permeabilidade absoluta
K_{ro}	Permeabilidade relativa
P	Pressão
P_{cog}	Pressão capilar óleo-gás
P_{cow}	Pressão capilar óleo-água
R_s	Razão de solubilidade
S	Saturação
V_b	Volume da célula
Z	Elevação
Δ	Dimensão de uma célula
Φ	Potencial
γ	Peso específico
μ	Viscosidade
ϕ	Porosidade
q	Vazão volumétrica

Capítulo 1

Introdução

No ramo da engenharia de reservatórios de petróleo, a simulação de reservatórios é uma ferramenta importante para auxiliar as tomadas de decisão dos campos explorados, como pode ser visto em ROSA *et al.* [1]. Um dos pontos críticos de uma simulação de reservatório é o tempo de simulação, que, a depender do modelo, pode durar dias. Em conjunto com o fato que o custo computacional aumenta conforme o tamanho dos modelos, novas técnicas devem ser encontradas com o objetivo de melhorar o tempo de simulação e com isso diminuir seu custo.

Esses modelos são baseados em equações diferenciais parciais não-lineares que são discretizadas, por exemplo, via método de volume finitos. Alguns dos métodos de solução mais comuns encontrados na literatura serão apresentados. O primeiro é o IMPES que trata a pressão implicitamente e a saturação explicitamente, o segundo é o FIM que trata todas as equações como sendo implícitas e o terceiro é o método AIM que realiza uma combinação dos dois métodos, IMPES e FIM, durante uma simulação. Essas equações discretizadas dão origem a um sistema de equações algébricas não-lineares que, por sua vez, são resolvidas através do método de Newton. Nesse método é necessário resolver sistemas lineares em cada passo de tempo. No processo de solução dos sistemas lineares $Ax = b$ usam-se métodos iterativos de Krylov, já que métodos diretos são inviáveis dado o tamanho dos problemas. Métodos de Krylov precisam de pré-condicionadores, e um dos mais utilizados na área de simulação de reservatórios é a fatoração incompleta. É comum que se aplique um ordenamento na matriz A para que o pré-condicionador baseado em uma fatoração incompleta tanto ajude ao método ter uma melhor convergência, i.e. menos iterações para se resolver o sistema, quanto tenha o menor preenchimento possível, visando economia de memória e de operações de ponto flutuante. A aplicação do ordenamento das equações e variáveis é um assunto bem difundido na literatura.

Em uma simulação AIM utilizando a ordenação natural, a matriz do sistema fica com blocos de tamanhos diferentes espalhados dentro dela como pode ser observado na Figura 1.2 à esquerda. Essa matriz foi gerada a partir da malha à esquerda da

1	2	3
4	5	6
7	8	9

1	2	5
3	6	7
8	9	4

Figura 1.1: Malhas 4 x 4 ordenadas com a ordenação natural e a ordenação induzida. As células em verde foram marcadas como FIM.

Figura 1.1. Com isso, foi desenvolvida uma estrutura de dados capaz de armazenar as entradas dessa matriz. Por sua vez, quando a ordenação induzida é utilizada, a malha é ordenada de tal que suas células FIM sejam ordenadas em primeiro, como pode ser observado na malha à direita da Figura 1.1, com isso a matriz do sistema é montada tal que suas primeiras linhas são referentes às células FIM, como pode ser observado na Figura 1.2 à direita. Com isso, a estrutura de dados necessária para tratar esses dados é mais simples. A fatoração incompleta pode ser separada em duas etapas: a fatoração simbólica e a fatoração numérica. Uma grande vantagem da ordenação natural é que a fatoração simbólica é executada somente uma vez durante a simulação. Enquanto para a ordenação induzida a fatoração simbólica deve ser executada a cada vez que se altere a distribuição das células FIM e IMPES. O percentual da fatoração simbólica no tempo do solver linear em uma simulação com a ordenação induzida pode chegar a 52% enquanto esse valor pode cair para 1% quando executada por uma simulação com a ordenação natural.

x	x	x	x		x	x	x		x			
x	x	x	x		x	x	x		x			
x	x	x	x	x			x					
x	x	x	x	x			x					
		x	x	x			x					
x	x				x	x	x		x			
x	x				x	x	x		x			
x	x	x	x		x	x	x	x		x		
				x			x	x			x	x
x	x				x	x			x	x		
						x	x	x	x	x		
							x	x	x	x		
								x	x		x	x
								x	x		x	x

x	x	x	x	x	x								
x	x	x	x	x	x								
x	x	x	x					x	x				
x	x	x	x					x	x				
x	x				x	x				x			
x	x				x	x				x			
							x	x			x		
							x	x			x		
									x				
									x				
										x			
											x	x	
												x	x

Figura 1.2: Estrutura da matriz utilizando a ordenação natural e a ordenação induzida.

Este trabalho propõe a comparação de dois pré-condicionadores utilizando a ordenação natural e induzida aplicados a simulações utilizando o método AIM em um

simulador desenvolvido pela Petrobras, onde foram observados algumas dificuldades durante as simulações que utilizavam o método AIM. Portanto, baseado na ideia de BRIDSON e TANG [2] foi realizada uma discussão onde é avaliada a estrutura dos grafos das matrizes geradas durante o processo de simulação. Os testes realizados em simulações completas de campos reais apontaram que o pré-condicionador que utiliza a ordenação natural obteve speed-up de 2,46 quando comparado com as simulações FIM para os mesmos campos enquanto o pré-condicionador que utiliza a ordenação induzida alcançou um speed-up de 1,77 em relação ao mesmo caso.

O restante desse trabalho está dividido da seguinte forma: *(i)* o Capítulo 2 apresenta alguns conceitos relativos às formulações IMPES, FIM e AIM; *(ii)* o Capítulo 3 mostra os efeitos do ordenamento em um pré-condicionador baseado na fatoração de Cholesky incompleta; *(iii)* o Capítulo 4 descreve a implementação das estruturas de dados capazes de tratar os blocos de tamanhos híbridos; *(iv)* o Capítulo 5 apresenta os testes e resultados para os dois pré-condicionadores com as duas ordenações; *(v)* a conclusão desse trabalho.

Capítulo 2

Métodos de Solução

Este capítulo apresenta conceitos relativos às formulações, métodos de solução e técnicas de manipulação de equações encontradas na literatura para problemas de fluxo em reservatórios. Considera-se que o reservatório tenha passado por um processo de discretização do meio poroso fazendo com que ele tenha células com propriedades definidas pelo modelo geológico.

Segundo MATTAX *et al.* [3], o termo *métodos de solução* é constantemente utilizado em uma maneira geral para cobrir todas as decisões de formulação e opções de solução que são feitas pelo simulador ou pelos usuários. Por *formulação* entende-se como sendo o tipo e estrutura das equações a serem tratadas, isto é, como os problemas de fluxo em reservatório são reduzidos e expressados através das equações. Uma vez que se tenha tomado a decisão de qual formulação utilizar, deve-se escolher o método de discretização temporal para se resolver as equações geradas de forma a minimizar os recursos computacionais para se obter uma solução ao problema.

Formulações explícitas podem ser expressas diretamente a partir de outros valores conhecidos, onde nesse caso há somente uma variável desconhecida. Essas formulações podem ter sérios problemas de convergência e com isso tendem a levar mais passos de tempo para completar uma simulação. Contudo, devido ao fato que somente uma variável é resolvida, o custo computacional por passo de tempo é baixo para essas formulações. Por sua vez, formulações implícitas são aquelas onde há mais de uma variável desconhecida e todas elas serão avaliadas no mesmo passo de tempo, o que leva as formulações implícitas a terem um custo computacional maior por passo de tempo quando comparadas com as formulações explícitas. Porém, o custo total da simulação pode ser menor já que é possível realizar uma simulação com menos passos de tempo. Com isso, deve-se balancear o custo por passo de tempo e a estabilidade.

O simulador também deve escolher como as equações geradas serão manipuladas quando forem resolvidas. As duas técnicas mais conhecidas para se manipular as equações são separadas em duas categorias: simultâneas e sequenciais. A técnica

simultânea junta todas as variáveis dependentes no mesmo conjunto de equações. Por sua vez, na abordagem sequencial as equações são manipuladas para separar a solução da equação da pressão com a solução da saturação.

2.1 FIM - Fully Implicit Method

O método FIM (*Fully Implicit Method*), proposto por DOUGLAS JR *et al.* [4], resolve simultaneamente as variáveis de pressão e saturação em um passo de tempo. BLAIR *et al.* [5] foram os primeiros a utilizar um simulador totalmente implícito onde acharam necessário utilizar um método de Newton a fim de se obter passos de tempo de maior tamanho. Porém, inerente à utilização de um método iterativo como o método de Newton, espera-se, naturalmente, que o esforço computacional do método FIM seja maior que o do método IMPES. Porém, segundo MATTAX *et al.* [3], esse fato é mais do que compensado pela possibilidade de se obter soluções estáveis para problemas relacionados à cones de água, percolação de gás e outros problemas onde há grande variação da saturação.

A análise realizada por AZIZ *et al.* [6] para a estabilidade do método FIM mostra que, como as variáveis primárias são tratadas implicitamente, ele é incondicionalmente estável nesse aspecto. Entretanto, segundo SETTARI e AZIZ [7], essa estabilidade está relacionada diretamente com o fato de se obter um maior erro na solução do problema. Com isso, chegou-se à seguinte observação: conforme a implicitude do método aumenta, sua estabilidade aumenta, mas os erros também aumentam.

As formulações listadas tanto para o método FIM quanto para o método IMPES foram retiradas de MACHADO [8] e AZIZ *et al.* [6].

2.1.1 Formulação

Para um sistema trifásico, as equações de conservação de massa para a fase óleo é representada pela Equação 2.1, para a fase água pela Equação 2.2 e para a fase gás pela Equação 2.3. Além dessas três equações, necessita-se da Equação 2.4 relativa à soma das saturações e a Equação 2.5 e Equação 2.6 relativas à pressão capilar.

$$\begin{aligned} & \frac{\partial}{\partial x} \left(\frac{A_x k_x k_{ro}}{\mu_o B_o} \frac{\partial \Phi_o}{\partial x} \right) \Delta x + \\ & \frac{\partial}{\partial y} \left(\frac{A_y k_y k_{ro}}{\mu_o B_o} \frac{\partial \Phi_o}{\partial y} \right) \Delta y + \\ & \frac{\partial}{\partial z} \left(\frac{A_z k_z k_{ro}}{\mu_o B_o} \frac{\partial \Phi_o}{\partial z} \right) \Delta z + q_o = V_b \frac{\partial}{\partial t} \left(\frac{\phi S_o}{B_o} \right) \end{aligned} \quad (2.1)$$

$$\begin{aligned}
& \frac{\partial}{\partial x} \left(\frac{A_x k_x k_{rw}}{\mu_w B_w} \frac{\partial \Phi_w}{\partial x} \right) \Delta x + \\
& \frac{\partial}{\partial y} \left(\frac{A_y k_y k_{rw}}{\mu_w B_w} \frac{\partial \Phi_w}{\partial y} \right) \Delta y + \\
& \frac{\partial}{\partial z} \left(\frac{A_z k_z k_{rw}}{\mu_w B_w} \frac{\partial \Phi_w}{\partial z} \right) \Delta z + q_o = V_b \frac{\partial}{\partial t} \left(\frac{\phi S_w}{B_w} \right)
\end{aligned} \tag{2.2}$$

$$\begin{aligned}
& \frac{\partial}{\partial x} \left(\frac{A_x k_x k_{rg}}{\mu_g B_g} \frac{\partial \Phi_g}{\partial x} + \frac{A_x k_x k_{ro} R_s}{\mu_o B_o} \frac{\partial \Phi_o}{\partial x} \right) \Delta x + \\
& \frac{\partial}{\partial y} \left(\frac{A_y k_y k_{rg}}{\mu_g B_g} \frac{\partial \Phi_g}{\partial y} + \frac{A_y k_y k_{ro} R_s}{\mu_o B_o} \frac{\partial \Phi_o}{\partial y} \right) \Delta y + \\
& \frac{\partial}{\partial z} \left(\frac{A_z k_z k_{rg}}{\mu_g B_g} \frac{\partial \Phi_g}{\partial z} + \frac{A_z k_z k_{ro} R_s}{\mu_o B_o} \frac{\partial \Phi_o}{\partial z} \right) \Delta z + q_g = V_b \frac{\partial}{\partial t} \left(\frac{\phi S_g}{B_g} + \frac{\phi R_s S_o}{B_o} \right)
\end{aligned} \tag{2.3}$$

$$S_o + S_w + S_g = 1 \tag{2.4}$$

$$P_{cow} = P_o - P_w \tag{2.5}$$

$$P_{cog} = P_o - P_g \tag{2.6}$$

Substituindo as pressões da fases água e gás e a saturação de gás nas Equações 2.1, 2.2 e 2.3, levando em consideração as equações 2.7–2.15, a discretização geométrica através do método de diferenças finitas em conjunto com a discretização temporal para um sistema unidimensional, ou seja, considerando somente o fluxo em x , obtém-se as equações 2.16, 2.17 e 2.18 para as fases óleo, água e gás.

$$X_o = \frac{\partial}{\partial x} \left(\frac{A_x k_x k_{ro}}{\mu_o B_o} \gamma_o \frac{\partial Z}{\partial x} \right) \Delta x + \frac{\partial}{\partial y} \left(\frac{A_y k_y k_{ro}}{\mu_o B_o} \gamma_o \frac{\partial Z}{\partial y} \right) \Delta y + \frac{\partial}{\partial z} \left(\frac{A_z k_z k_{ro}}{\mu_o B_o} \gamma_o \frac{\partial Z}{\partial z} \right) \Delta z \tag{2.7}$$

$$X_w = \frac{\partial}{\partial x} \left(\frac{A_x k_x k_{rw}}{\mu_w B_w} \gamma_w \frac{\partial Z}{\partial x} \right) \Delta x + \frac{\partial}{\partial y} \left(\frac{A_y k_y k_{rw}}{\mu_w B_w} \gamma_w \frac{\partial Z}{\partial y} \right) \Delta y + \frac{\partial}{\partial z} \left(\frac{A_z k_z k_{rw}}{\mu_w B_w} \gamma_w \frac{\partial Z}{\partial z} \right) \Delta z \tag{2.8}$$

$$X_g = \frac{\partial}{\partial x} \left(\frac{A_x k_x k_{rg}}{\mu_g B_g} \gamma_g \frac{\partial Z}{\partial x} \right) \Delta x + \frac{\partial}{\partial y} \left(\frac{A_y k_y k_{rg}}{\mu_g B_g} \gamma_g \frac{\partial Z}{\partial y} \right) \Delta y + \frac{\partial}{\partial z} \left(\frac{A_z k_z k_{rg}}{\mu_g B_g} \gamma_g \frac{\partial Z}{\partial z} \right) \Delta z \tag{2.9}$$

$$X_{dg} = \frac{\partial}{\partial x} \left(\frac{A_x k_x k_{rg} R_s}{\mu_g B_g} \gamma_g \frac{\partial Z}{\partial x} \right) \Delta x + \frac{\partial}{\partial y} \left(\frac{A_y k_y k_{rg} R_s}{\mu_g B_g} \gamma_g \frac{\partial Z}{\partial y} \right) \Delta y + \frac{\partial}{\partial z} \left(\frac{A_z k_z k_{rg} R_s}{\mu_g B_g} \gamma_g \frac{\partial Z}{\partial z} \right) \Delta z \tag{2.10}$$

$$A_w = \frac{\partial}{\partial x} \left(\frac{A_x k_x k_{rw}}{\mu_w B_w} \gamma_w \frac{\partial P_{cow}}{\partial x} \right) \Delta x + \frac{\partial}{\partial y} \left(\frac{A_y k_y k_{rw}}{\mu_w B_w} \gamma_w \frac{\partial P_{cow}}{\partial y} \right) \Delta y + \frac{\partial}{\partial z} \left(\frac{A_z k_z k_{rw}}{\mu_w B_w} \gamma_w \frac{\partial P_{cow}}{\partial z} \right) \Delta z \tag{2.11}$$

$$A_g = \frac{\partial}{\partial x} \left(\frac{A_x k_x k_{rg}}{\mu_g B_g} \gamma_g \frac{\partial P_{cog}}{\partial x} \right) \Delta x + \frac{\partial}{\partial y} \left(\frac{A_y k_y k_{rg}}{\mu_g B_g} \gamma_g \frac{\partial P_{cog}}{\partial y} \right) \Delta y + \frac{\partial}{\partial z} \left(\frac{A_z k_z k_{rg}}{\mu_g B_g} \gamma_g \frac{\partial P_{cog}}{\partial z} \right) \Delta z \tag{2.12}$$

$$T_{ox} = \frac{A_x k_x k_{ro}}{\mu_o B_o \Delta x}; \quad T_{oy} = \frac{A_y k_y k_{ro}}{\mu_o B_o \Delta y}; \quad T_{oz} = \frac{A_z k_z k_{ro}}{\mu_o B_o \Delta z} \quad (2.13)$$

$$T_{wx} = \frac{A_x k_x k_{rw}}{\mu_w B_w \Delta x}; \quad T_{wy} = \frac{A_y k_y k_{rw}}{\mu_w B_w \Delta y}; \quad T_{wz} = \frac{A_z k_z k_{rw}}{\mu_w B_w \Delta z} \quad (2.14)$$

$$T_{gx} = \frac{A_x k_x k_{rg}}{\mu_g B_g \Delta x}; \quad T_{gy} = \frac{A_y k_y k_{rg}}{\mu_g B_g \Delta y}; \quad T_{gz} = \frac{A_z k_z k_{rg}}{\mu_g B_g \Delta z} \quad (2.15)$$

Discretização para a fase óleo:

$$\begin{aligned} & T_{ox_{i+\frac{1}{2}}}^{n+1} (p_{o_{i+1}} - p_{o_i})^{n+1} - \\ & T_{ox_{i-\frac{1}{2}}}^{n+1} (p_{o_i} - p_{o_{i-1}})^{n+1} + \\ & q_{o_i}^{n+1} = \frac{V_{b_i}}{\Delta t} \left[\left(\frac{\phi'}{B_o^n} + \phi^{n+1} \left(\frac{1}{B_o} \right)' \right) \right. \\ & \quad \times (1 - S_w^n - S_g^n) \\ & \quad \times (p_o^{n+1} - p_o^n) \\ & \quad - \left(\frac{\phi}{B_o} \right)^{n+1} (S_w^{n+1} - S_w^n) \\ & \quad \left. - \left(\frac{\phi}{B_o} \right)^{n+1} (S_g^{n+1} - S_g^n) \right] + X_o^{n+1} \end{aligned} \quad (2.16)$$

Discretização para a fase água:

$$\begin{aligned} & T_{wx_{i+\frac{1}{2}}}^{n+1} (p_{o_{i+1}} - p_{o_i})^{n+1} - \\ & T_{wx_{i-\frac{1}{2}}}^{n+1} (p_{o_i} - p_{o_{i-1}})^{n+1} + \\ & q_{w_i}^{n+1} = \frac{V_{b_i}}{\Delta t} \left[\left(\frac{\phi'}{B_w^n} + \phi^{n+1} \left(\frac{1}{B_w} \right)' \right) \right. \\ & \quad \times S_w^n \\ & \quad \times (p_o^{n+1} - p_o^n) \\ & \quad - \left(\frac{\phi}{B_w} \right)^{n+1} (S_w^{n+1} - S_w^n) \\ & \quad \left. - \left(\frac{\phi}{B_w} \right)^{n+1} (S_g^{n+1} - S_g^n) \right] + X_w^{n+1} + A_w^{n+1} \end{aligned} \quad (2.17)$$

Discretização para a fase gás:

$$\begin{aligned}
& T_{gx_{i+\frac{1}{2}}}^{n+1} (p_{o_{i+1}} - p_{o_i})^{n+1} - \\
& T_{gx_{i-\frac{1}{2}}}^{n+1} (p_{o_i} - p_{o_{i-1}})^{n+1} + \\
& T_{ox_{i+\frac{1}{2}}}^{n+1} R_s^{n+1} (p_{o_{i+1}} - p_{o_i})^{n+1} + \\
& T_{ox_{i-\frac{1}{2}}}^{n+1} R_s^{n+1} (p_{o_i} - p_{o_{i-1}})^{n+1} + \\
& q_{g_i}^{n+1} = \frac{V_{b_i}}{\Delta t} \left\{ \left[\left(\frac{\phi'}{B_g^n} + \phi^{n+1} \left(\frac{1}{B_g} \right)' \right) S_g^n \right. \right. \\
& \quad \left. \left. + \left[\left(\frac{\phi'}{B_g^n} + \phi^{n+1} \left(\frac{1}{B_g} \right)' \right) R_s^n + \left(\frac{\phi}{B_o} \right)^{n+1} R_s' \right] \right. \right. \\
& \quad \left. \left. \times \left(1 - S_w^n - S_g^n \right) \right] \right. \\
& \quad \left. \times \left(p_o^{n+1} - p_o^n \right) \right. \\
& \quad \left. - \left(\frac{\phi}{B_o} \right)^{n+1} R_s^{n+1} (S_w^{n+1} - S_w^n) \right. \\
& \quad \left. + \left[\left(\frac{\phi}{B_g} \right)^{n+1} - \left(\frac{\phi}{B_o} \right)^{n+1} R_s^{n+1} \right] (S_g^{n+1} - S_g^n) \right\} \\
& \quad + X_g^{n+1} + X_{dg}^{n+1} + A_g^{n+1}
\end{aligned} \tag{2.18}$$

Onde

$$\begin{aligned}
\phi' &= \frac{\phi^{n+1} - \phi^n}{p_o^{n+1} - p_o^n} \\
R_s' &= \frac{R_s^{n+1} - R_s^n}{p_o^{n+1} - p_o^n} \\
\left(\frac{1}{B_l} \right)' &= \frac{\frac{1}{B_l^{n+1}} - \frac{1}{B_l^n}}{p_o^{n+1} - p_o^n}, \quad \text{para } l = o, w, g
\end{aligned}$$

As equações 2.16, 2.17 e 2.18 representam a discretização espacial e temporal de um sistema trifásico unidimensional, porém, deve-se lembrar que as equações apresentadas são não-lineares. As transmissibilidades dessas equações são dependentes das permeabilidades relativas que por sua vez são dependentes das saturações, o que obriga a utilização de algum método de linearização para a solução do sistema, como por exemplo o Método de Newton.

2.2 IMPES - Implicit Pressure, Explicit Saturation

O método IMPES (*Implicit Pressure, Explicit Saturation*), proposto originalmente por SHELDON *et al.* [9] e STONE *et al.* [10], realiza o cálculo das saturações de forma explícita enquanto o cálculo da pressão é mantido de forma implícita. No início do passo de tempo, assumindo que a pressão capilar não irá mudar durante o passo de tempo, as saturações são eliminadas através da adição das equações de balanço de material das fases individuais. A equação resultante tem somente a pressão como variável desconhecida, que é obtida implicitamente através da solução simultânea do conjunto de equações.

Segundo MATTAX *et al.* [3], as aproximações feitas pelo método IMPES em relação à avaliação dos coeficientes de pressão no passo de tempo antigo e da saturação no passo de tempo atual, podem colocar severas restrições na solução do sistema. Os problemas surgem nos parâmetros dependentes da saturação, permeabilidade relativa e pressão capilar, os quais mudam rapidamente. Com isso, o método IMPES pode não ser adequado para problemas cujas propriedades sofram rápidas variações durante a simulação. Por exemplo, a combinação de células de tamanho pequeno do modelo discreto do reservatório em conjunto com uma alta taxa de transferência dos fluidos entre as células devido à uma alta permeabilidade entre elas resulta em rápidas mudanças de saturação do reservatório. Além desses, há os problemas de cones de água e da percolação de gás, o qual pode fluir verticalmente. Nesses dois últimos problemas, o passo de tempo deve ser pequeno suficiente para evitar oscilações nas saturações calculadas nas regiões onde o fluxo passa.

Uma análise sobre a estabilidade do método IMPES foi realizada por AZIZ *et al.* [6] onde são listadas duas possíveis limitações de estabilidade do método. A primeira vem do tratamento explícito das variáveis primárias, a segunda vem do tratamento explícito das transmissibilidades onde há um alto grau de não linearidade. Em relação às variáveis primárias, o método IMPES trata a pressão capilar explicitamente e por isso a sua estabilidade depende diretamente da magnitude da variação da pressão capilar em relação à saturação. Em relação à não linearidade do tratamento explícito das transmissibilidades, o limite de estabilidade do método IMPES pode ser calculado através da Equação 2.19, que é conhecida como condição CFL (*Courant-Friedrichs-Lewy*), descrita por RUSSELL *et al.* [11],

$$c = \frac{v\Delta t}{\Delta x} \quad (2.19)$$

onde Δt é o passo de tempo máximo, Δx é o tamanho da célula e v é a velocidade da frente de avanço da onda. Segundo RUSSELL *et al.* [11], o método IMPES

precisa que $c \leq 1$ para ser estável e não ter oscilações, com isso, seu passo de tempo está restrito por essa condição, a qual pode ser bastante severa em casos reais de simulação de reservatório. Portanto, com o objetivo de se obter soluções estáveis e precisas, o método IMPES precisa, obrigatoriamente, utilizar passos de tempo pequenos.

2.2.1 Formulação

A formulação IMPES consiste em resolver o sistema implicitamente para as pressões e, a partir dos valores de pressões calculados para o passo de tempo $n + 1$, obter o valor das saturações em $n + 1$ explicitamente. As Equações 2.16, 2.17 e 2.18 possuem como incógnitas a pressão da fase óleo e as saturações da fase gás e fase água no passo de tempo $n + 1$. Considerando que se dispõe de 3 equações e 3 incógnitas, pode-se combiná-las de modo a eliminar duas incógnitas. Na formulação IMPES combinam-se as equações de forma a eliminar as saturações já que apenas a pressão é resolvida implicitamente e as saturações são resolvidas explicitamente.

Esquemáticamente, para cada fase, dispõe-se das seguintes funções com as respectivas incógnitas entre parênteses:

- Óleo: $f_o(p_o^{n+1}, S_w^{n+1}, S_g^{n+1}) = 0$
- Água: $f_w(p_o^{n+1}, S_w^{n+1}, S_g^{n+1}) = 0$
- Gás: $f_g(p_o^{n+1}, S_w^{n+1}, S_g^{n+1}) = 0$, onde se pode explicitar S_g^{n+1} para obter $S_g^{n+1} = f_{g_2}(p_o^{n+1}, S_w^{n+1})$

Com uma expressão para S_g^{n+1} , as funções f_o e f_w podem ser reescritas, a partir da substituição de S_g^{n+1} em suas expressões, de modo que ficam restando somente 2 incógnitas.

- Óleo: $f_{o_2}(p_o^{n+1}, S_w^{n+1}) = 0$
- Água: $f_{w_2}(p_o^{n+1}, S_w^{n+1}) = 0$, de onde se pode explicitar S_w^{n+1} para obter $S_w^{n+1} = f_{w_3}(p_o^{n+1})$.

Esta última expressão pode ser substituída na equação da fase óleo, reduzindo, assim, nosso problema a uma única incógnita, p_o^{n+1} . Assim, o sistema pode ser resolvido da seguinte forma:

$$\begin{aligned} f_{w_3}(p_o^{n+1}) = 0 &\Rightarrow p_o^{n+1} \Rightarrow S_w^{n+1} = f_{w_3}(p_o^{n+1}) \\ &\Rightarrow S_g^{n+1} = f_{g_2}(p_o^{n+1}, S_w^{n+1}) \Rightarrow S_o^{n+1} = 1 - S_g^{n+1} - S_w^{n+1} \quad (2.20) \end{aligned}$$

Combinando então as duas equações acima de forma a eliminar os termos da saturação, obtém-se a seguinte equação

$$\begin{aligned}
& \left(P1 \times p_{o_{i+1}}^{n+1} \right) + \left(P2 \times p_{o_{i-1}}^{n+1} \right) - \\
& \left[(P3 + P4 + P5 + P6) \times p_{o_i}^{n+1} \right] = \\
& \qquad \qquad \qquad - \left[(P7 + P8)_i \times p_{o_i}^n \right] \\
& \qquad \qquad \qquad - P9 \\
& \qquad \qquad \qquad + [P10 \times ((Z_{i+1} - Z_i))] \\
& \qquad \qquad \qquad - [P11 \times ((Z_i - Z_{i-1}))]
\end{aligned} \tag{2.21}$$

Onde

$$f(y) = \left(\frac{\phi'}{B_y^n} + \phi^{n+1} \left(\frac{1}{B_y} \right)' \right), \quad \text{para } y = o, w, g$$

$$G = \left(B_o - R_s B_g \right)^{n+1}$$

$$T_f^s = T_{f_{x_1 s \frac{1}{2}}}^n \quad \text{para } s = [+,-]$$

$$\gamma_f^s = \gamma_{f_1 s \frac{1}{2}}^n \quad \text{para } s = [+,-]$$

$$P1 = G_i T_o^+ + B_{w_i}^{n+1} T_w^+ + B_{g_i}^{n+1} \left(T_g^+ + T_o^+ R_{s_{1+\frac{1}{2}}}^n \right)$$

$$P2 = G_i T_o^- + B_{w_i}^{n+1} T_w^- + B_{g_i}^{n+1} \left(T_g^- + T_o^- R_{s_{1-\frac{1}{2}}}^n \right)$$

$$P3 = G_i \left[\frac{V_b}{\Delta t} f(o) (1 - S_w^n - S_g^n) \right]_i + B_{w_i}^{n+1} \left[\frac{V_b}{\Delta t} f(w) S_w^n \right]_i$$

$$P4 = B_{g_i}^{n+1} \left\{ \frac{V_b}{\Delta t} \left[\left(f(o) R_s^n + \left(\frac{\phi}{B_o} \right)^{n+1} R'_s \right) (1 - S_w^n - S_g^n) + f(g) S_g^n \right] \right\}_i$$

$$P5 = G_i T_o^+ + B_{w_i}^{n+1} T_w^+ + B_{g_i}^{n+1} \left(T_g^+ + T_o^+ R_{s_{1+\frac{1}{2}}}^n \right)$$

$$P6 = G_i T_o^- + B_{w_i}^{n+1} T_w^- + B_{g_i}^{n+1} \left(T_g^- + T_o^- R_{s_{1-\frac{1}{2}}}^n \right)$$

$$P7 = G \left\{ \frac{V_b}{\Delta t} f(o) (1 - S_w^n - S_g^n) \right\} + B_w^{n+1} \left\{ \frac{V_b}{\Delta t} f(w) S_w^n \right\}_i$$

$$P8 = B_g^{n+1} \left\{ \frac{V_b}{\Delta t} \left[\left(f(o)R_s^n + \left(\frac{\phi}{B_o} \right)^{n+1} R'_s \right) (1 - S_w^n - S_g^n) + f(g)S_g^n \right] \right\}$$

$$P9 = \left[G \times q_o^n + B_w^{n+1} q_w^n + B_g^{n+1} q_g^n \right]_i$$

$$+ B_{w_i}^{n+1} T_w^+ (P_{cow_{i+1}}^n - P_{cow_i}^n)$$

$$- B_{w_i}^{n+1} T_w^- (P_{cow_i}^n - P_{cow_{i-1}}^n)$$

$$- B_{g_i}^{n+1} T_g^+ (P_{cow_{i+1}}^n - P_{cow_i}^n)$$

$$- B_{g_i}^{n+1} T_g^- (P_{cgo_i}^n - P_{cgo_{i-1}}^n)$$

$$P10 = G_i T_o^+ \gamma_o^+ + B_w^{n+1} T_w^+ \gamma_w^+ + B_g^{n+1} \left(T_g^+ \gamma_g^+ + T_o^+ R_{s_{1+\frac{1}{2}}}^n \gamma_o^+ \right)$$

$$P11 = G_i T_o^- \gamma_o^- + B_w^{n+1} T_w^- \gamma_w^- + B_g^{n+1} \left(T_g^- \gamma_g^- + T_o^- R_{s_{1-\frac{1}{2}}}^n \gamma_o^- \right)$$

O processo de cálculo implícito da pressão e explícito da saturação é dado pelos seguintes passos:

- Selecione um valor de passo de tempo Δt ;
- Calcule os termos de produção no passo de tempo escolhido;
- Calcule, explicitamente, os coeficientes de transmissibilidade da Equação 2.21 de todos os blocos;
- Resolva a Equação 2.21; depois da solução p_o^{n+1} torna-se conhecido;
- Substitua a pressão calculada na equação de fluxo de água, determinando assim S_w^{n+1} ;
- Calcule a saturação de gás na equação de fluxo de gás, obtendo, então, S_g^{n+1} ;
- Pela diferença, determine $S_o^{n+1} = 1 - S_g^{n+1} - S_w^{n+1}$;
- Controle o erro de balanço de materiais;

2.3 AIM - Adaptive Implicit Method

O método AIM (*Adaptive Implicit Method*), proposto por THOMAS *et al.* [12] e melhorado por FORSYTH e SAMMON [13] tem como objetivo utilizar tanto a pressão como a saturação de forma implícita nas regiões onde há uma grande variação de saturação, como por exemplo, nas proximidades de um poço. Em outras regiões, somente a pressão é considerada implícita. Deve-se observar que o custo da solução

aumenta conforme a quantidade de células implícitas aumenta. Com isso, deve-se aperfeiçoar o grau de implicitude baseado no tempo de simulação e na localização das células.

Uma das grandes dificuldades do método AIM é o critério de troca para se alterar entre o cálculo da célula como implícita ou explícita. Uma descrição sobre o histórico do desenvolvimento dos critérios disponíveis na literatura pode ser encontrado em MARCONDES *et al.* [14]. O critério proposto por THOMAS *et al.* [12] especificava um limite de saturação a partir do passo de tempo anterior. Se houvesse uma variação da saturação maior que esse limite, o volume da célula era calculado utilizando o método FIM. Como foi observado por FORSYTH e SAMMON [13], esse critério não permitia que uma célula voltasse a se tornar IMPES uma vez que ela tenha se tornado FIM. A abordagem proposta por RUSSELL *et al.* [11] sugere utilizar um critério baseado no número CFL para determinar qual célula se torna FIM ou IMPES, fazendo com que o método AIM produza respostas mais precisas e estáveis que o método FIM, a partir do momento que há redução no erro numérico. Por sua vez, o critério proposto por YOUNG *et al.* [15] sugere que o número CFL seja utilizado com o cálculo da variação máxima da saturação. O critério proposto por FUNG *et al.* [16] se baseia numa análise local das equações aproximadas onde, em cada célula, os autovalores calculados servirão como critério para decidir se a célula deverá ser FIM ou IMPES. Por fim, o trabalho realizado por MARCONDES *et al.* [14] adaptou o critério criado por FUNG *et al.* [16] com uma variação específica no campo de saturação para malhas não-estruturadas.

Capítulo 3

Ordenamento em um Pré-Condicionador Incompleto de Cholesky

No processo de solução de sistemas $n \times n$ simétricos, positivo-definidos na forma de $Ax = b$ onde A é uma matriz esparsa e n é grande, é comum aplicar um reordenamento na matriz A antes de se fazer sua fatoração incompleta. O objetivo desse reordenamento é melhorar a convergência do sistema e diminuir o número de operações de ponto flutuante.

Segundo YANNAKAKIS [17], esse problema é NP-Completo e portanto algumas heurísticas são utilizadas para se obter essa matriz de permutação. Alguns exemplos de heurísticas que obtêm ordenamentos através de conjuntos de níveis conhecidas na literatura são o ordenamento de *Minimum Degree*, que geram fatorações com um baixo número de preenchimentos, os ordenamentos de *CutHill-McKee* e sua versão reversa que reduzem a banda e o profile da matriz, além do *Nested Dissection* que utiliza técnicas recursivas para achar um ordenamento. Há também ordenamentos que utilizam coloração de grafos, onde um dos mais conhecidos e difundidos na literatura é o *red-black* onde seu objetivo é determinar uma coloração dos nós do grafo de uma matriz tal que todos os nós adjacentes tenham cores diferentes.

A importância do ordenamento de variáveis e equações no desempenho de pré-condicionadores baseados na fatoração de Cholesky incompleta (ICHOL) está bem difundida na literatura e tem se tornado tópico padrão em livros de métodos iterativos como SAAD [18] e MEURANT [19]. O efeito do ordenamento das variáveis para fatoração incompleta foi estudado experimentalmente por DUFF e MEURANT [20].

Nesta seção, baseado na ideia proposta por BRIDSON e TANG [2], será desenvolvida uma discussão que irá mostrar que há situações relevantes onde a ordenação aplicada ao método AIM pode resultar em pré-condicionadores ICHOL fracos. As

ordenações comparadas aqui são a ordenação natural e uma ordenação onde são priorizadas as células FIM para serem ordenadas antes das células IMPES que aqui será chamada de ordenação induzida. A simulação de reservatórios produz matrizes que são em geral estruturalmente simétricas quando observadas do ponto de vista de blocos, portanto, para essa discussão, somente matrizes simétricas serão consideradas. Note-se que todos os argumentos são baseados em propriedades da estrutura da matriz e sua representação gráfica associada, logo não há perda de generalidade em se restringir a discussão ao caso simétrico.

Seja A uma matriz simétrica, positivo-definida e

$$A \approx LL^T \quad (3.1)$$

é uma fatoração incompleta de A onde L é triangular inferior. Como foi mostrado em BRIDSON e TANG [2], uma condição desejável para que uma fatoração ICHOL seja boa é que seu padrão de elementos não-nulos da inversa da matriz $(LL^T)^{-1} = L^{-T}L^{-1}$ deve ter o mesmo padrão encontrado na inversa verdadeira A^{-1} . Para matrizes irredutíveis, como é o caso das matrizes encontradas na simulação de reservatórios, é desejável que a inversa da fatoração ICHOL tenha a mesma matriz de esparsidade da A^{-1} . Caso essa condição não seja satisfeita, significa que a aproximação pode não ser boa.

Uma observação feita por BRIDSON e TANG [2] é que sua análise é puramente estrutural e, então, os resultados podem ser aplicados a matrizes que sejam estruturalmente simétricas.

Um conceito importante é o de caminho monotonicamente decrescente entre dois nós de um grafo de uma matriz, o qual é um caminho tal que todos os nós intermediários são sempre menores que o nó anterior. Por exemplo, um caminho entre i e j que passe por k , ou seja, $i \rightarrow k \rightarrow j$ é monotonicamente decrescente se e somente se $i > k$ e $k > j$. Portanto, para uma matriz triangular L , $L_{ij}^{-1} \neq 0$ implica que existe um caminho monotonicamente decrescente entre i e j no grafo que representa a estrutura de não-zeros de L .

Uma vez que,

$$(LL^T)_{ij}^{-1} = \sum_{k=1}^n L_{ik}^{-T} L_{kj}^{-1} \quad (3.2)$$

para que $(LL^T)_{ij}^{-1} \neq 0$ deve existir pelo menos um $k \geq i$ com um caminho monotonicamente decrescente entre k e i e entre k e j .

Os exemplos desse capítulo serão baseados em uma malha 5 x 5 com células FIM distribuídas no canto inferior direito da malha. Os grafos à esquerda da Figura 3.1 e da Figura 3.2 representam, respectivamente, essa malha utilizando a ordenação natural e a ordenação induzida. Nesses grafos, as células coloridas em verde são as

células FIM. Note que essas células na Figura 3.2 são as primeiras a serem ordenadas ao contrário do que pode ser observado na Figura 3.1. Já as matrizes de esparsidade à direita da Figura 3.1 e da Figura 3.2 mostram como a esparsidade muda conforme a escolha da ordenação. No caso das matrizes de simulação de reservatórios deve-se realizar a fatoração ICHOL por blocos de elementos uma vez que a fatoração ICHOL por elemento pode gerar pré-condicionadores quase singulares. Portanto, cada ponto da matriz gerada pode representar um bloco de elementos ou um único elemento.

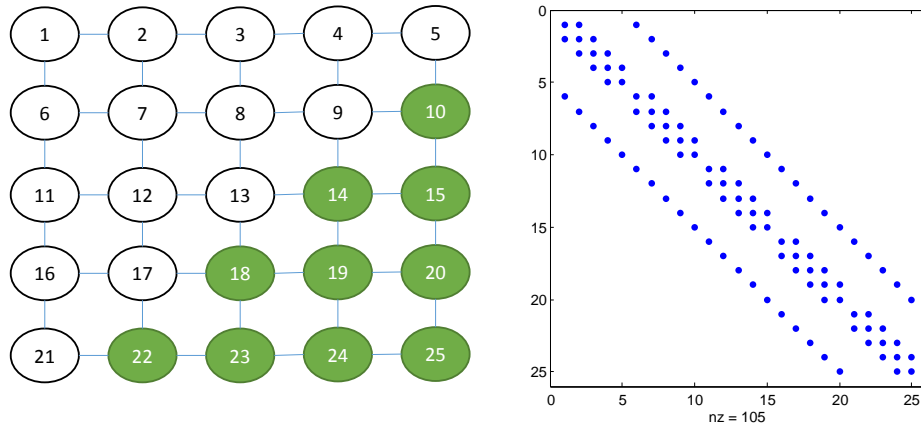


Figura 3.1: Grafo correspondente e matriz de esparsidade para uma malha 5 x 5 com ordenação natural.

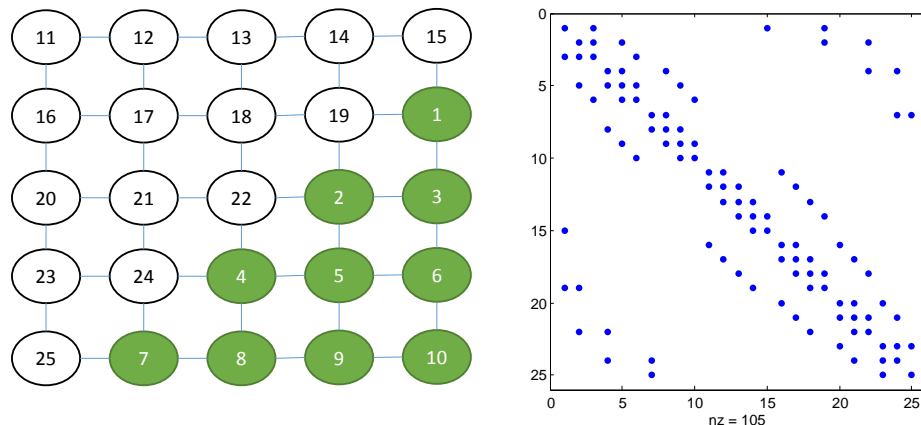


Figura 3.2: Grafo correspondente e matriz de esparsidade para uma malha 5 x 5 com ordenação induzida.

3.1 Análise para ICHOL(0)

Com o intuito de se caracterizar a estrutura de não-zeros do fator L a fim de analisar as ordenações natural e ordenação induzida, vamos considerar primeiramente o nível zero da fatoração ICHOL, ou seja, ICHOL(0). Nela, a estrutura dos elementos não-nulos de L é a mesma que a estrutura triangular inferior de A . A Figura 3.3 mostra

o grafo induzido pelo fator L da fatoração ICHOL(0) para uma malha 5 x 5 junto com seu padrão de esparsidade quando a ordenação natural é aplicada.

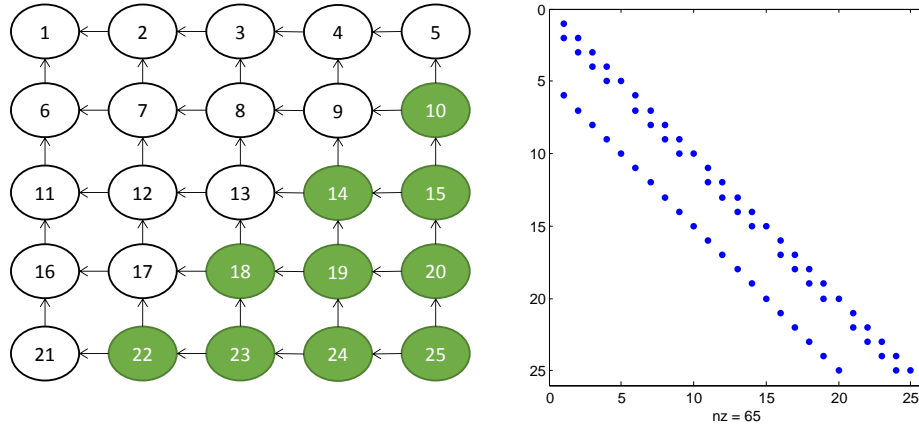


Figura 3.3: Grafo e padrão de esparsidade associados à parte L da fatoração ICHOL(0) de uma malha 5 x 5 com ordenação natural.

Para um problema 2D geral, dados quaisquer nós i e j com $i > j$, algumas situações podem acontecer:

1. i e j estão na mesma linha. Neste caso, i deve estar à direita de j se estiver na mesma linha ou abaixo de j caso esteja na mesma coluna. De qualquer forma, há um caminho monotonicamente decrescente a partir de i até j através da linha do grafo juntando i e j , o que implica que $(LL^T)_{ij}^{-1} \neq 0$.
2. i está em uma linha abaixo de j e à direita de j . Neste caso, para ir de i até j através das linhas do grafo resultará em um caminho monotonicamente decrescente fazendo com que $(LL^T)_{ij}^{-1} \neq 0$.
3. i está em uma linha abaixo de j e à esquerda de j . Neste caso, não há um caminho monotonicamente decrescente entre i e j mas, obtendo um k para ser um nó de interseção entre a linha i e a coluna j , existe um caminho monotonicamente decrescente entre k e i e de k até j através das linhas do grafo e, portanto, $(LL^T)_{ij}^{-1} \neq 0$.

Alguns exemplos podem ser dados para as condições acima baseados no grafo da Figura 3.3. Para a situação do item 1 pode-se considerar os nós $i = 14$ e $j = 12$ que resulta em um caminho $14 \rightarrow 13 \rightarrow 12$. Já a situação do item 2 pode-se considerar os nós $i = 19$ e $j = 7$ resultando em um caminho $19 \rightarrow 18 \rightarrow 17 \rightarrow 12 \rightarrow 7$. Por fim, para exemplificarmos a situação do item 3 pode-se escolher os nós $i = 17$ e $j = 9$. Para obter um caminho entre esses nós, temos que escolher, por exemplo, $k = 19$ que faz interseção tanto com i quanto com j e com isso os caminhos gerados são $19 \rightarrow 18 \rightarrow 17$ para ligar k a i e $19 \rightarrow 14 \rightarrow 9$ para ligar k a j .

As condições acima também podem ser aplicadas a casos 3D se planos forem considerados. A ideia proposta por BRIDSON e TANG [2] mostra que para $(LL^T)^{-1}$ ser uma matriz cheia basta observar que existe um caminho monotonicamente decrescente entre um nó n para qualquer nó i . Contudo, para se compreender onde a ordenação induzida falha nos itens acima serão utilizados argumentos mais explícitos entre quaisquer dois nós. Para uma malha regular 3D utilizando estêncil de 7 pontos ou 2D utilizando estêncil de 5 pontos, pode-se derivar o Teorema 2.1 do BRIDSON e TANG [2] para obter o seguinte resultado:

Corolário 3.1. *Seja A uma matriz proveniente de uma malha regular 3D (2D) utilizando estêncil de 7 pontos (5 pontos) ordenados naturalmente e L a matriz triangular inferior resultante da fatoração $ICHOL(0)$ de A . Então, desprezando quaisquer cancelamentos numéricos casuais, $(LL^T)^{-1} = L^{-T}L^{-1}$ é uma matriz cheia.*

Entretanto, as situações mencionadas acima não são verdadeiras para os casos onde a ordenação induzida é aplicada. A Figura 3.4 mostra o grafo gerado pelo fator L da fatoração $ICHOL(0)$ para uma malha 5×5 junto com seu padrão de esparsidade utilizando a ordenação induzida. Não há caminho algum entre um nó IMPES e um nó FIM fora do limite entre as regiões e, portanto, se i é um nó IMPES e j é um nó FIM que não esteja no limite das regiões, então $(LL^T)^{-1} = 0$. Pode-se observar como exemplo dessa situação os nós $i = 22$ e $j = 5$.

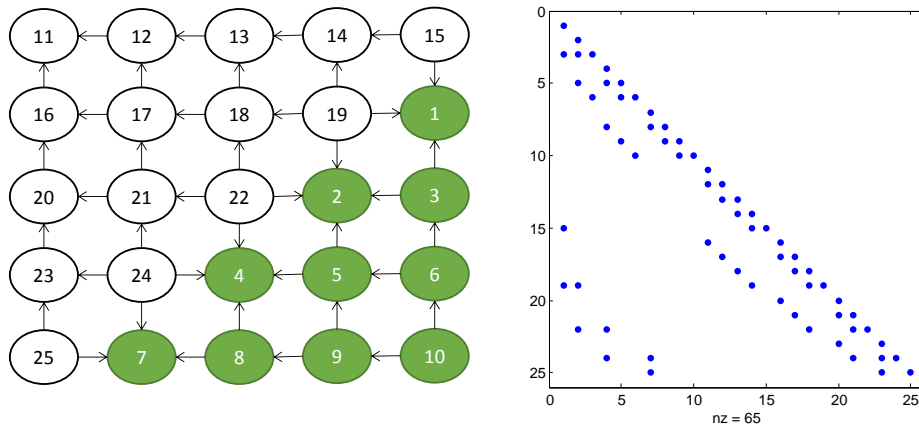


Figura 3.4: Grafo e padrão de esparsidade associados à parte L da fatoração $ICHOL(0)$ de uma malha 5×5 com ordenação induzida.

Inclusive, até conexões entre dois nós IMPES podem resultar em um respectivo valor zero na inversa de LL^T . Por exemplo, ao se tentar achar um caminho entre os nós 22 e 19, precisa-se achar um $k > 22$ pois o nó 22 está abaixo e à esquerda do nó 19. Entretanto, uma vez que o nó 22 tem 3 conexões menores que o nó 19 e uma conexão com o nó 21 mas que a partir deste não se consegue achar um caminho até o nó 19 torna-se impossível achar um $k > 22$ que faria com que houvesse caminhos monotonicamente decrescentes entre 22 e 19. Em geral, se i é um nó IMPES na

região limítrofe com a região FIM e j é qualquer nó IMPES à direita de i então $(LL^T)_{ij}^{-1} = 0$. Portanto, a situação considerada pelo item 3 não pode ser aplicada com essa ordenação uma vez que as possíveis interseções estariam na região FIM onde todos os nós são numerados com grau menor que os nós IMPES.

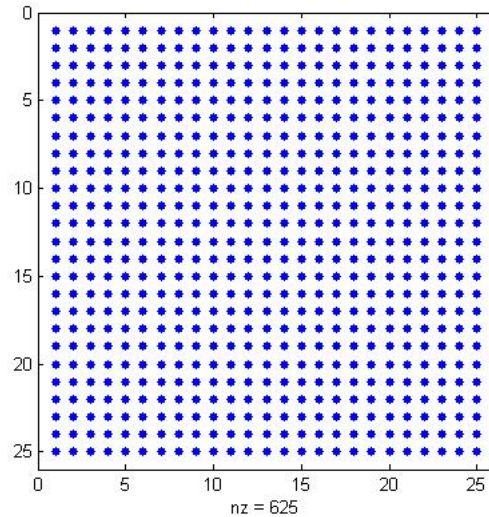


Figura 3.5: Padrão de esparsidade dos elementos não-zero para a inversa verdadeira.

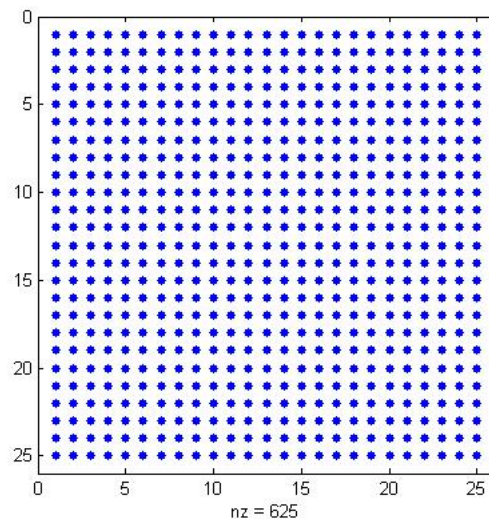


Figura 3.6: Padrão de esparsidade dos elementos não-zero para a inversa utilizando a ordenação natural.

Foram geradas as inversas para o operador de Laplace com condições de contorno de Dirichlet aplicados à malha referente à Figura 3.1. O padrão de esparsidade da inversa verdadeira pode ser observado na Figura 3.5. Por sua vez, os padrões de esparsidade das inversas aproximadas utilizando a ordenação natural e a induzida podem ser observados na Figura 3.6 e Figura 3.7 respectivamente. A função do MATLAB que gera esse padrão de esparsidade coloca um ponto azul para cada elemento

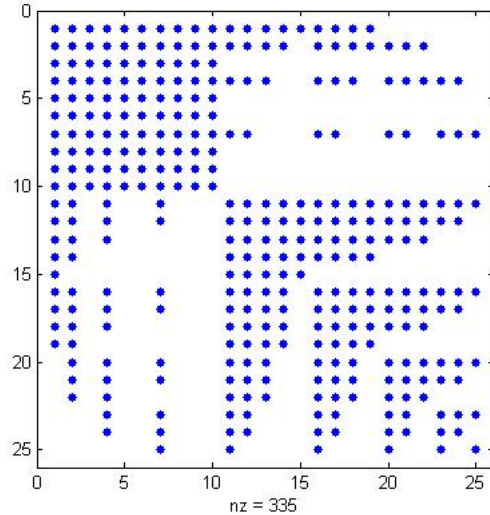


Figura 3.7: Padrão de esparsidade dos elementos não-zero para a inversa utilizando a ordenação induzida.

não-zero na matriz. Com isso, a comparação do resultado das inversas aproximadas com a inversa verdadeira mostra que todos os valores da inversa aproximada utilizando a ordenação natural são não-zero. Entretanto, para a ordenação induzida, todas as conexões com os nós FIM que não estejam na região limítrofe e algumas conexões com nós IMPES tem valor zero e por isso não aparecem no seu respectivo gráfico. Levando em consideração as condições necessárias especificadas por BRIDSON e TANG [2], pode ser que a inversa aproximada utilizando a ordenação induzida seja pior que a inversa aproximada utilizando a ordenação natural.

3.2 Análise para ICHOL(1)

Na fatoração ICHOL(1), preenchimentos são permitidos quando eles correspondem a uma posição que surge a partir da operação entre duas entradas originais da matriz. Na representação do grafo da matriz, segundo SAAD [18], no processo da fatoração ICHOL, o valor do nível de um preenchimento na posição (i, j) é p se, e somente se, existe um caminho de tamanho $p + 1$ entre i e j de tal forma que todos os vértices nesse caminho, exceto os pontos finais i e j , são menores que i e j . Portanto, no caso de ICHOL(1), vértices adicionais que ligam dois nós i e j são adicionados sempre que houver um caminho monotonicamente decrescente de tamanho dois entre i e j .

A Figura 3.8 mostra uma parte de um grafo para uma malha regular 2D onde mostra que há um caminho entre os nós i e $i - nx + 1$ através do nó $i - nx$. Este caminho introduzirá um preenchimento de nível 1 através da uma diagonal na direção nordeste do grafo (representado através de uma linha pontilhada na figura). A Figura 3.9 à esquerda mostra o grafo correspondente para o fator L resultante da

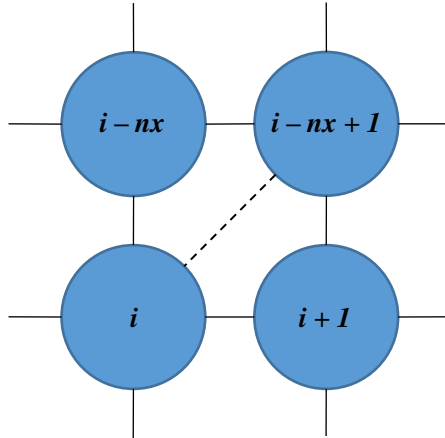


Figura 3.8: Representação de um grafo de uma malha regular mostrando o caminho entre dois nós causado por um preenchimento.

fatoração ICHOL(1) do exemplo da Figura 3.1 onde a ordenação natural foi aplicada. A diagonal extra que pode ser observada na matriz de esparsidade da Figura 3.9 à direita se dá pelo fato de que um nível de preenchimento é permitido ao se utilizar ICHOL(1). Em uma malha 3D, existiriam outros dois níveis de preenchimento os quais ligariam o nó i com os nós no plano de cima, o que faria com que aparecessem três diagonais extras na matriz de esparsidade. A partir do ponto de vista da estrutura de não-zeros de $(LL^T)^{-1}$, o Corolário 3.1 ainda se aplica e, com isso, $(LL^T)^{-1}$ será uma matriz cheia, tendo em vista que o grafo do fator L da fatoração ICHOL(1) vai conter todos os vértices do grafo do fator L da fatoração ICHOL(0).

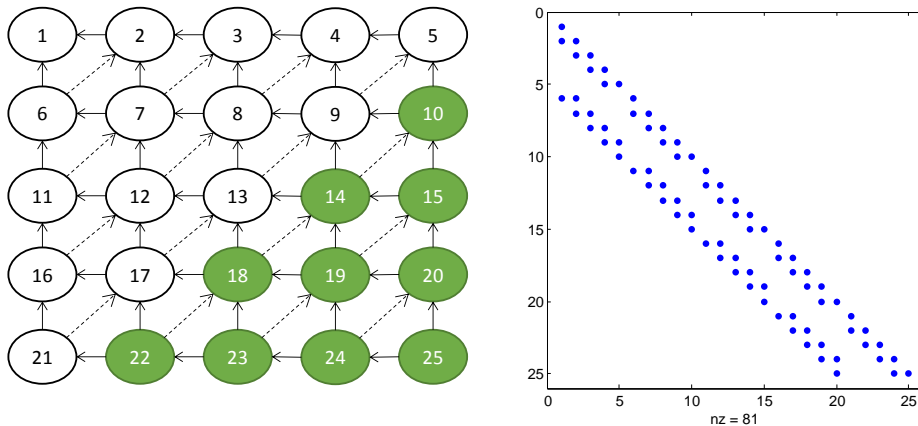


Figura 3.9: Grafo e padrão de esparsidade associados à parte L da fatoração ICHOL(1) de uma malha 5×5 com ordenação natural.

Para a ordenação induzida não se obtém um padrão de preenchimento tão regular quando o padrão obtido pela ordenação natural devido ao fato de que alguns preenchimentos aparecem entre os limites das regiões FIM e IMPES. A Figura 3.10 mostra o grafo e o padrão de esparsidade obtidos do fator L da fatoração ICHOL(1) da malha utilizada na Figura 3.2. Fora da interface entre as regiões FIM e IMPES, o

mesmo padrão de preenchimentos é encontrado, porém, com a direção invertida dentro da região FIM. Na região limítrofe, além de haver alguns preenchimentos entre nós FIM com nós IMPES, há preenchimentos extras de nós IMPES para um nó FIM dentro da região FIM. Esses preenchimentos são introduzidos pelos preenchimentos de nível 1 e podem ser observados, por exemplo, entre os nós 19 e 5 onde há um caminho $19 \rightarrow 2 \rightarrow 5$ ou entre os nós 25 e 8, onde há um caminho $25 \rightarrow 7 \rightarrow 8$.

Apesar dos novos preenchimentos, introduzidos na interface entre as regiões FIM e IMPES pela fatoração ICHOL(1), gerarem entradas não-zero para $(LL^T)^{-1}$ em lugares onde zeros existiram para a fatoração ICHOL(0), por exemplo $i = 22$ e $j = 5$, não é suficiente para fazer com que $(LL^T)^{-1}$ seja uma matriz cheia, devido ao fato que os nós com valores zero persistirão dentro da região FIM.

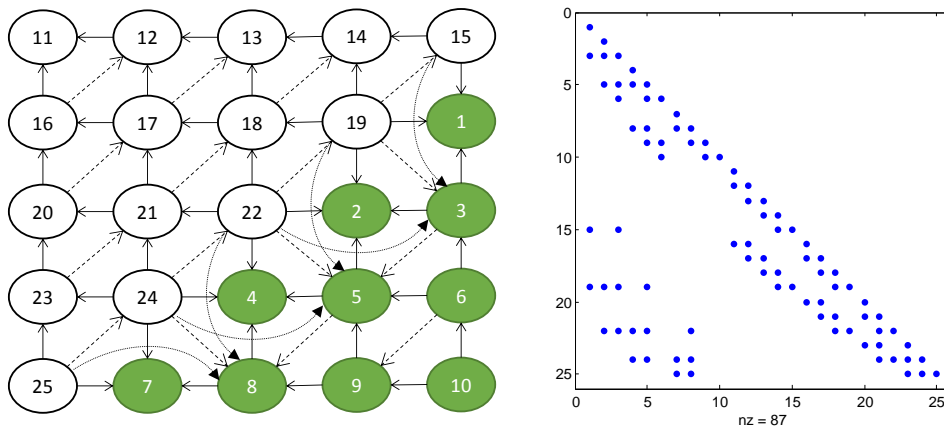


Figura 3.10: Grafo e padrão de esparsidade associados à parte L da fatoração ICHOL(1) de uma malha 5×5 com ordenação induzida.

Foi realizada uma comparação com ICHOL(1) para o problema apresentado na seção anterior. A Figura 3.11 mostra o padrão de esparsidade obtido para a ordenação induzida com preenchimento de nível 1.

Observa-se que apesar da fatoração ICHOL(1) para a ordenação induzida introduzir alguns preenchimentos ela ainda se observam que existem vários elementos nulos e, portanto, não atende à condição necessária especificada por BRIDSON e TANG [2]. Com isso, há indícios de que a inversa aproximada utilizando a ordenação induzida seja pior do que a inversa aproximada utilizando a ordenação natural.

3.3 Observações adicionais

Pode-se argumentar que os problemas encontrados nas análises feitas com ICHOL(0) e ICHOL(1) são concentrados em uma região perto da interface entre as células FIM e IMPES e, portanto, se essa região limítrofe fosse mantida longe das regiões com uma atividade maior, não haveria impacto significativo. Entretanto, a escolha da distribuição das células FIM/IMPES é baseada no comportamento da parte hiperbólica

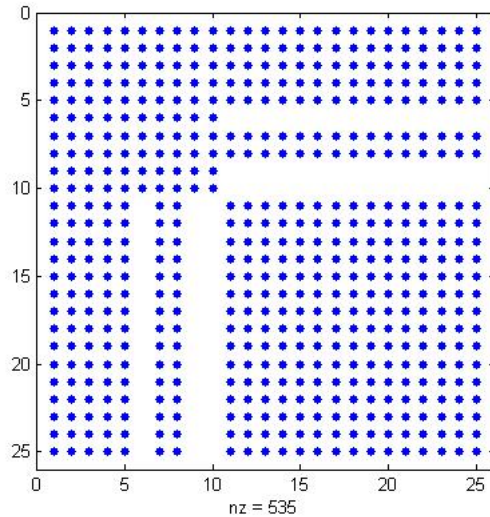


Figura 3.11: Padrão de esparsidade dos elementos não-zero para a inversa utilizando a ordenação induzida utilizando preenchimento de nível 1.

das equações governantes, enquanto os problemas encontrados são relacionados com a solução do sistema linear, onde alguns componentes da parte elíptica do problema, tais como os componentes de erro de baixa frequência da pressão, são os que determinam a taxa de convergência.

As situações onde aparecem zeros na inversa aproximada utilizando a fatoração ICHOL com a ordenação induzida não são restritas ao tipo de padrão introduzido pela Figura 3.2. Por exemplo, a conexão entre qualquer célula pertencente a um grupo de células IMPES cercadas por células FIM com qualquer outra célula IMPES fora desse grupo vai resultar em um 0 na inversa aproximada.

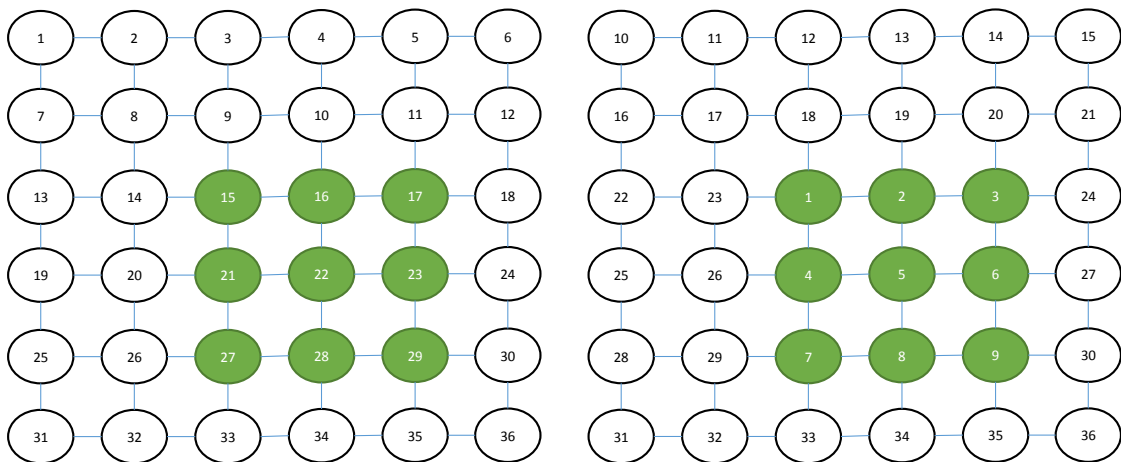


Figura 3.12: Grafos de uma malha 6 x 6 com um grupo de células FIM cercadas por células IMPES.

Mesmo quando a inversa aproximada da ordenação induzida tiver o mesmo padrão de esparsidade, há razões para se acreditar que ela resulta em um pré-condicionamento pior quando comparado com a inversa aproximada com ordenação

natural. Considerando a malha mostrada na Figura 3.12, onde as células FIM estão no meio da malha, temos que $(LL^T)^{-1}$ tem o mesmo padrão de esparsidade para ambas as ordenações, uma vez que é possível alcançar qualquer nó do grafo a partir do nó 36 utilizando um caminho monotonicamente decrescente. Entretanto, para a ordenação induzida, há alguns caminhos que são maiores que os caminhos obtidos pela ordenação natural. Por exemplo, seja $i = 20$ e $j = 22$ no grafo à esquerda e $i = 26$ e $j = 5$ para o grafo à direita. Os caminhos encontrados são $22 \rightarrow 21 \rightarrow 20$ para a ordenação natural enquanto para a ordenação induzida é necessário ir até o nó 34, obtendo-se os caminhos $34 \rightarrow 33 \rightarrow 32 \rightarrow 29 \rightarrow 26$ e $34 \rightarrow 8 \rightarrow 5$. Esses caminhos maiores podem resultar em uma inversa aproximada mais fraca, como pode-se esperar, em algumas circunstâncias como diagonal dominância, que valores pequenos seriam obtidos depois de cada passo do cálculo da inversa da matriz.

3.4 Experimentos numéricos

Foram rodados alguns testes para se avaliar o desempenho do pré-condicionador ICHOL quando aplicado a um gradiente conjugado utilizando ordenação natural e a ordenação induzida. Os testes consistem em se resolver e avaliar o número de iterações obtidas para um operador de Laplace com condições de contorno de Dirichlet em uma malha 3D uniforme. A estação onde foram executadas as simulações foi uma HP-Z820 com 128GB de memória RAM, 2 processadores Intel Xeon ES-2687W a uma frequência de 3.10GHz totalizando 16 cores em um sistema operacional Windows 7 Enterprise com Service Pack 1. A versão do MATLAB utilizada para rodar os testes foi a R2012b. A distribuição das células FIM e IMPES foi baseada em uma malha 3D onde o mesmo padrão 2D de células foi repetido em todos os planos. Conforme pode-se observar na Figura 3.13, três padrões foram estabelecidos: 1/4 de 5-spot, centralizado e 5-spot. No primeiro, as células FIM foram distribuídas entre o canto superior esquerdo e inferior direito. No segundo, as células FIM foram concentradas no centro da malha, e, finalmente, no último padrão, as células FIM são distribuídas entre cinco grupos, entre os quatro cantos e o centro. Para os testes com a ordenação induzida, a quantidade de células FIM a ser distribuída variou entre 1% e 50%.

O lado direito do sistema a ser resolvido foi calculado de tal maneira que um vetor unitário fosse a solução do sistema. O chute inicial foi um vetor de zeros e a tolerância para a convergência foi de 10^{-6} uma vez que o método *pcg* usa como critério de convergência o resíduo relativo.

A Tabela 3.1 mostra o número de iterações utilizando ICHOL(0) realizadas pelo método *pcg* com a ordenação natural e induzida para malhas variando entre 8 mil células até 2 milhões de células. A coluna “Size” mostra o tamanho da malha em

Malha	Size	N Iter	% FIM	1/4 5-spot		Centralizado		5-spot	
				N Iter	Variação	N Iter	Variação	N Iter	Variação
20 x 20 x 20	8	20	1%	21	5,0%	20	0,0%	21	5,0%
			5%	21	5,0%	20	0,0%	21	5,0%
			10%	22	10,0%	20	0,0%	21	5,0%
			20%	22	10,0%	20	0,0%	22	10,0%
			35%	23	15,0%	20	0,0%	22	10,0%
			50%	23	15,0%	20	0,0%	22	10,0%
40 x 40 x 20	32	29	1%	30	3,4%	30	3,4%	30	3,4%
			5%	33	13,8%	33	13,8%	33	13,8%
			10%	34	17,2%	33	13,8%	33	13,8%
			20%	33	13,8%	34	17,2%	34	17,2%
			35%	34	17,2%	34	17,2%	34	17,2%
			50%	34	17,2%	34	17,2%	34	17,2%
80 x 80 x 20	128	39	1%	41	5,1%	41	5,1%	41	5,1%
			5%	41	5,1%	41	5,1%	41	5,1%
			10%	42	7,7%	41	5,1%	41	5,1%
			20%	42	7,7%	42	7,7%	42	7,7%
			35%	45	15,4%	43	10,3%	43	10,3%
			50%	45	15,4%	43	10,3%	43	10,3%
160 x 160 x 20	512	47	1%	48	2,1%	48	2,1%	48	2,1%
			5%	49	4,3%	48	2,1%	48	2,1%
			10%	49	4,3%	49	4,3%	49	4,3%
			20%	49	4,3%	49	4,3%	49	4,3%
			35%	49	4,3%	49	4,3%	49	4,3%
			50%	49	4,3%	49	4,3%	49	4,3%
320 x 320 x 20	2048	47	1%	49	4,3%	49	4,3%	49	4,3%
			5%	49	4,3%	49	4,3%	49	4,3%
			10%	49	4,3%	50	6,4%	50	6,4%
			20%	50	6,4%	50	6,4%	50	6,4%
			35%	50	6,4%	50	6,4%	50	6,4%
			50%	50	6,4%	50	6,4%	50	6,4%

Tabela 3.1: Número de iterações realizadas pelo *pcg* para $ICHOL(0)$.



Figura 3.13: Padrão de distribuição das células FIM e IMPES utilizada nos experimentos numéricos.

Malha	Size	N Iter	% FIM	1/4 5-spot		Centralizado		5-spot	
				N Iter	Variação	N Iter	Variação	N Iter	Variação
20 x 20 x 20	8	15	1%	15	0,0%	15	0,0%	15	0,0%
			5%	15	0,0%	15	0,0%	15	0,0%
			10%	15	0,0%	15	0,0%	15	0,0%
			20%	15	0,0%	15	0,0%	15	0,0%
			35%	15	0,0%	15	0,0%	15	0,0%
			50%	15	0,0%	15	0,0%	15	0,0%
40 x 40 x 20	32	20	1%	20	0,0%	21	5,0%	20	0,0%
			5%	21	5,0%	21	5,0%	21	5,0%
			10%	22	10,0%	21	5,0%	22	10,0%
			20%	22	10,0%	22	10,0%	22	10,0%
			35%	23	15,0%	22	10,0%	22	10,0%
			50%	23	15,0%	22	10,0%	22	10,0%
80 x 80 x 20	128	26	1%	27	3,8%	27	3,8%	27	3,8%
			5%	27	3,8%	28	7,7%	27	3,8%
			10%	27	3,8%	29	11,5%	27	3,8%
			20%	27	3,8%	29	11,5%	28	7,7%
			35%	29	11,5%	29	11,5%	29	11,5%
			50%	29	11,5%	27	3,8%	29	11,5%

Tabela 3.2: Número de iterações realizadas pelo *pcg* para ICHOL(1).

milhares de células. A primeira coluna “N Iter” exibe o número de iterações realizadas com a ordenação natural e as outras colunas “N Iter” mostram o número de iterações para a ordenação induzida. Para essa última ordenação e para cada tamanho de malha há um percentual de implicitude aplicado, ou seja, a quantidade de células FIM que será distribuída conforme o padrão escolhido. Além disso, para cada padrão, há uma coluna que indica a variação no número de iterações entre as duas ordenações. Em todas as configurações, exceto pela configuração “Centralizado” e 8 mil células, o número das iterações foi maior para a ordenação induzida. Esse aumento chegou até a faixa dos 17% e foi proporcional conforme o aumento do número de células FIM.

A Tabela 3.2 mostra o resultado das iterações realizadas pelo método *pcg* utilizando ICHOL(1) para a ordenação natural e para com a ordenação induzida. Não foi possível realizar testes com os tamanhos de malhas 160 x 160 x 20 e 320 x 320

Malha	Size	NNZ	% FIM	1/4 5-spot		Centralizado		5-spot	
				NNZ	Aumento	NNZ	Aumento	NNZ	Aumento
20 x 20 x 20	8	52460	1%	52480	0,0%	52540	0,2%	52494	0,1%
			5%	52580	0,2%	52640	0,3%	52620	0,3%
			10%	52660	0,4%	52720	0,5%	52780	0,6%
			20%	52760	0,6%	52820	0,7%	52940	0,9%
			35%	52880	0,8%	52940	0,9%	53160	1,3%
			50%	52960	1,0%	53040	1,1%	53320	1,6%
40 x 40 x 20	32	214500	1%	214600	0,0%	214660	0,1%	214662	0,1%
			5%	214800	0,1%	214860	0,2%	214980	0,2%
			10%	214960	0,2%	215020	0,2%	215260	0,4%
			20%	215160	0,3%	215220	0,3%	215580	0,5%
			35%	215400	0,4%	215460	0,4%	216020	0,7%
			50%	215560	0,5%	215640	0,5%	216340	0,9%
80 x 80 x 20	128	867380	1%	867640	0,0%	867700	0,0%	867818	0,1%
			5%	868040	0,1%	868100	0,1%	868460	0,1%
			10%	868320	0,1%	868400	0,1%	869000	0,2%
			20%	868760	0,2%	868820	0,2%	869740	0,3%
			35%	869200	0,2%	869280	0,2%	870520	0,4%
			50%	869600	0,3%	869660	0,3%	871160	0,4%

Tabela 3.3: Número de elementos não-zero para o fator L do ICHOL(1).

x 20 devido ao longo tempo necessário para se realizar a fatoração. Para os outros tamanhos de malha, foi observado que o aumento no número de iterações não é tão grande quanto o observado para o ICHOL(0), mas a ordenação induzida realiza mais iterações em 2/3 dos casos e nunca realiza menos iterações. Para a menor malha, o número de iterações é o mesmo para ambas as ordenações em todos os padrões e porcentagem de células FIM, mas para as outras duas, o número de iterações cresce entre 10% e 15%. Assim como no ICHOL(0), em geral o número de iterações com a ordenação induzida cresce com a porcentagem de células FIM e não é tão sensível com o padrão de distribuição das células FIM e IMPES. A Tabela 3.3 apresenta o número de elementos não-zeros no fator L para cada configuração proposta. Assim como discutido na Seção 3.2, a ordenação induzida introduz preenchimentos adicionais na região limite entre as células FIM e IMPES, os quais não são presentes quando a ordenação natural é utilizada. O número de elementos não-zero adicionais cresce conforme a porcentagem de células FIM cresce, como esperado, mas esse aumento é muito pequeno sendo menos que 1% na maioria dos casos.

Capítulo 4

Implementação

Nesse capítulo vamos discutir a estrutura de dados e alguns aspectos da fatoração incompleta de um simulador de nova geração que utiliza a ordenação induzida. Vamos comparar com os simuladores da CMG (*Computer Modelling Groud LTD*), empresa canadense líder no mercado de simuladores de reservatórios de petróleo, que utilizam o ordenamento natural, nesse caso são necessárias estruturas de dados mais complexas para se armazenar as entradas das matrizes.

4.1 Blocos de Tamanhos Híbridos

Na discretização AIM, como vimos, células com tratamentos FIM e IMPES são utilizadas. A seleção de quais células recebem um tratamento ou outro é realizada dinamicamente no espaço e tempo. Em termos da matriz do sistema, a discretização AIM resulta em uma matriz com blocos de diferentes tamanhos na parte do reservatório, bem como nas conexões entre o reservatório e outras partes do sistema, como por exemplo, poços. Seja n_c o número de componentes, as conexões entre duas células FIM resulta em um bloco de tamanho $(n_c + 1) \times (n_c + 1)$, conexões entre duas células IMPES, por sua vez, resulta em blocos 1×1 , enquanto conexões entre uma célula FIM e outra IMPES resulta em um bloco retangular com $(n_c + 1) \times 1$ componentes para as derivadas das equações FIM em relação às variáveis IMPES e $1 \times (n_c + 1)$ para as derivadas das equações IMPES em relação às variáveis FIM. Note que o componente extra que é somado ao n_c é relativo à variável de pressão. A presença de blocos de tamanhos híbridos requer um tratamento especial em termos de estrutura de dados para se armazenar as entradas das matrizes.

Essas estruturas introduzem ponteiros adicionais no formato BCSR (*Blocked Compressed Sparse Row*) para cuidar da localização de cada entrada da matriz dentro do vetor que contém os valores não-nulos da matriz. A definição do BCSR se dá por três vetores: o vetor A contém as entradas de cada bloco da matriz armazenadas consecutivamente linha a linha, o vetor de inteiros $COLIDX$ que armazena,

para cada bloco, os índices das colunas dos blocos e o vetor *ROWIDX* que armazena o início de cada bloco em *A* e *COLIDX*. Os vetores *ROWIDX* e *COLIDX* usam numeração que começam pelo valor 1 (similar ao Fortran). Uma vez que o tamanho do bloco *b* é igual para todos os blocos, a posição inicial das entradas de qualquer bloco em *A* é facilmente calculada a partir dos índices em *ROWIDX* e multiplicando-os por b^2 . A Figura 4.1 mostra um exemplo do formato BCRS aplicado a uma matriz simples.

$$\begin{array}{c}
 \mathbf{M} = \\
 \begin{array}{|c|c|c|c|}
 \hline
 1 & 2 & & 3 & 4 \\
 5 & 6 & & 7 & 8 \\
 \hline
 & & 9 & 10 & \\
 & & 11 & 12 & \\
 \hline
 & & 13 & 14 & 15 & 16 \\
 & & 17 & 18 & 19 & 20 \\
 \hline
 \end{array} \\
 \\
 \mathbf{A} = [1 \ 2 \ 5 \ 6 \mid 3 \ 4 \ 7 \ 8 \mid 9 \ 10 \ 11 \ 12 \mid 13 \ 14 \ 17 \ 18 \mid 15 \ 16 \ 19 \ 20] \\
 \mathbf{COLIDX} = [1 \ 3 \mid 2 \mid 2 \ 3] \\
 \mathbf{ROWIDX} = [1 \ 3 \ 4 \ 6]
 \end{array}$$

Figura 4.1: Matriz esparsa 3×3 com blocos de tamanho 2×2 e os respectivos vetores *A*, *COLIDX* e *ROWIDX*.

Entretanto, quando os blocos tem tamanhos diferentes não é possível se obter a mesma informação a partir de um simples cálculo, com isso, o vetor adicional *ABLKIDX* é introduzido para fornecer a posição inicial dos valores de cada bloco em *A*. Outro vetor adicional *XBLKIDX* é utilizado para indicar a posição inicial dos valores correspondentes a cada linha em qualquer vetor *X* seguindo a mesma estrutura em blocos do vetor *A*. Além desses, se faz necessário manter um vetor *NEQ* com o tamanho do bloco para cada linha.

A Figura 4.2 mostra um exemplo de uma matriz com blocos de tamanhos híbridos e como o formato BCRS é estendido para tratar esse problema. No BCRS estendido, os vetores *ROWIDX* e *COLIDX*, os quais também aparecem no BCRS padrão, dependem das conexões da malha, mas não do tamanho do bloco. Os vetores de ponteiros adicionais *ABLKIDX*, *XBLKIDX* e *NEQ* são os únicos que dependem do tamanho do bloco. Uma vez que a ordenação é fixa, mudanças na implicitude do problema somente fazem com que esses ponteiros adicionais sejam recalculados e, portanto, não há necessidade de se mudar o padrão de esparsidade e, logo, ter que refazer a fatoração simbólica. Esses cálculos, executados antes da fatoração numérica, geram uma perda de desempenho pequena, uma vez que os ponteiros adicionais podem ser recalculados com um algoritmo barato.

O Código 4.1 mostra como, a partir de um vetor booleano chamado *IMPMAP*, que indica quais células são FIM ou IMPES, pode-se montar os vetores *NEQ* e

M =	1	2	3	4		5			
	6	7	8	9		10			
	11	12	13	14	15		16		
	17	18	19	20	21		22		
			23	24	25				
	26	27				28	29	30	31
			32	33		34	35		36
						37		38	39
						41		42	43
							45	46	47

49
50
51
52
53
54
55
56
57
58

A = [1 2 6 7 | 3 4 8 9 | 5 10 | 11 12 17 18 | 13 14 19 20 | 15 21 | 16 22 | 23 24 | 25 | 26 27
| 28 | 29 | 30 31 | 32 33 | 34 | 35 | 36 | 37 41 | 38 39 42 43 | 40 44 | 45 | 46 47 | 48]

COLIDX = [1 2 4 | 1 2 3 5 | 2 3 | 1 4 5 6 | 2 4 5 7 | 4 6 7 | 5 6 7]

ROWIDX = [1 4 8 10 14 18 21 24]

ABLKIDX = [1 5 9 | 11 15 19 21 | 23 25 | 26 28 29 30 | 32 34 35 36 | 37 39 43 | 45 46 48]

NEQ = [2 2 1 1 1 2 1]

X = [49 50 | 51 52 | 53 | 54 | 55 | 56 57 | 58]

XBLKIDX = [1 3 5 6 7 8 10 11]

Figura 4.2: Matriz esparsa com seu formato estendido do BCRS para tratar blocos de tamanhos híbridos.

XBLKIDX. Inicialmente atribui-se à primeira posição do vetor *XBLKIDX* o valor 1, para que então, comece-se um laço em todas as linhas da matriz. Caso a linha *i* seja uma linha que contenha células implícitas, então, o tamanho do bloco é atribuído ao vetor *NEQ* na posição *i*, além de se atualizar a posição *i + 1* do vetor *XBLKIDX* com o valor do tamanho do bloco somado ao valor da posição *i* do vetor *XBLKIDX*. Caso contrário, ou seja, quando a linha *i* não contém células implícitas, o valor 1 é atribuído à posição *i* do vetor *NEQ*. Por sua vez, a posição *i + 1* do vetor *XBLKIDX* recebe o valor obtido na posição *i* do vetor *XBLKIDX* adicionado a 1.

Uma vez que se tenha o vetor *NEQ* calculado, o vetor *ABLKIDX* pode ser calculado com um simples laço em todos os blocos da matriz, como pode ser visto no Código 4.2. O código inicia atribuindo-se 0 ao contador de blocos chamado *ib* que será utilizado como índice de *ABLKIDX*, além de iniciar a primeira posição do vetor *ABLKIDX* como 0. Dois laços são realizados, o primeiro para percorrer as linhas da matriz e outro, interno ao primeiro, para percorrer as colunas da matriz para uma determinada linha *i*. Para cada coluna *j*, o vetor *ABLKIDX* é atualizado na posição *ib + 1* com o valor da posição *ib* do vetor *ABLKIDX* somado à multiplicação do tamanho dos blocos da linha *i* com a coluna *j*. Incrementa-se o contador de blocos e continua-se o laço até o final.

```

1  XBLKIDX[0] = 1;
2  // loop sobre as bloco-linhas
3  for ( int i = 0; i < n; i++) {
4      if (IMPMAP[i]) { // linha com celulas implicitas
5          NEQ[i] = blockSize;
6          XBLKIDX[i+1] = XBLKIDX[i] + blockSize;
7      }
8      else {
9          NEQ[i] = 1;
10         XBLKIDX[i+1] = XBLKIDX[i] + 1;
11     }
12 }

```

Código 4.1: Código para se atribuir NEQ e XBLKIDX.

```

1  int ib = 0;
2  ABLKIDX[0] = 1;
3  // loop sobre todas as bloco-linhas
4  for(int i = 0; i < n; i++) {
5      // loop sobre todas as bloco-colunas
6      // para o bloco-linha i
7      for(int jx = ROWIDX[i]; jx < ROWIDX[i+1]; jx++) {
8          int j = COLIDX[jx];
9          ABLKIDX[ib + 1] = ABLKIDX[ib] + NEQ[i] * NEQ[j];
10         ib++;
11     }
12 }

```

Código 4.2: Código para se atribuir ABLKIDX.

Outro aspecto relacionado à eficiência dos pré-condicionadores ILU é o fato de que aritmética de blocos pode ser estendida naturalmente quando se utilizam blocos de tamanho híbridos, com isso, essa abordagem é muito mais adequada para generalizações da discretização AIM, onde, ao invés de tratar ou todas as variáveis ou somente a pressão implicitamente, há a possibilidade de se escolher quais variáveis vão receber o tratamento explícito ou implícito, como proposto por RUSSELL *et al.* [11]. Nesse caso, blocos de qualquer tamanho entre 1 e $n_c + 1$ apareceriam e a abordagem de se separar as equações e variáveis em grupos de blocos do mesmo tamanho seria um trabalho árduo, sendo inviável na prática. Até mesmo uma simples abordagem de se ter células tratadas como FIM ou IMPES em uma simulação composicional seria tratada naturalmente no contexto de se ter blocos de tamanho híbrido, enquanto, no contexto da ordenação induzida, esse problema se torna mais complicado. Entretanto, ao se trabalhar com estrutura de dados com tamanhos de blocos híbridos faz com que se tenham endereçamentos de memória extras, o que se

faz inferir que haverá penalidades em relação ao desempenho. Algumas otimizações de laços, as quais são muito importantes a fim de se obter bom desempenho nos núcleos de álgebra linear tais como multiplicação matriz-vetor, fatoração ILU e o solver triangular, incluindo a utilização de laços desenrolados e a omissão de laços internos quando se sabe que o tamanho do bloco é igual a 1, não são facilmente realizáveis quando os tamanhos dos blocos são híbridos.

Os benefícios em desempenho adquiridos quando se utilizam matrizes esparsas em blocos no contexto de multiplicação matriz-vetor é amplamente discutido por IM [21].

Apesar de ser possível utilizar aritmética de blocos com o armazenamento utilizado com a ordenação induzida, seria necessário ou se misturar blocos de diferentes tamanhos ou seria feito através de uma abordagem utilizando complemento de Schur, o que, provavelmente, seria complexo para se estender para níveis de preenchimento maiores que 0. As entradas da matriz são armazenadas, pelo simulador escolhido, individualmente através de uma estrutura de dados homogênea ignorando a estrutura de blocos nela existente. As desvantagens de se utilizar o tratamento individual estão relacionadas aos potenciais problemas de estabilidade que podem acontecer, a um número maior de indireções de endereços de memória, o que torna essa implementação pouco eficiente, e, por fim, à perda da habilidade de se utilizar laços desenrolados. Entretanto, a uniformidade fornecida pelo uso de estruturas de dados homogêneas pode ser útil para tirar proveito de vetorização, o que pode ser crucial em algumas arquiteturas tais como paralelismo em placas gráficas.

4.2 Implementação das Estruturas de Blocos de Tamanhos Híbridos

Um protótipo de um solver linear baseado na estrutura de blocos de tamanho híbrido foi desenvolvido e integrado com o código de montagem da matriz de um simulador de nova geração, que internamente utiliza a ordenação induzida para simulações AIM. Esse novo solver linear permitiu rodar simulações completas utilizando o método AIM com a ordenação natural. Novas estruturas de dados foram criadas baseadas nas ideias apresentadas na Seção 4.1.

A estrutura de dados híbrida somente será utilizada para o pré-condicionador ILU, portanto, será necessário passar um vetor de permutação que mapeará a ordenação induzida para a ordenação natural para ser utilizada pela fatoração ILU e pelo solver triangular. Uma vez que o resto do solver iterativo vai utilizar um ordenamento diferente das equações e das variáveis, a aplicação do pré-condicionador ILU vai precisar de um mapeamento do lado direito do sistema para nova a ordenação

antes de se executar o solver triangular e mapear o vetor de solução de volta para a ordenação original do simulador. Note que a utilização de um vetor de permutação na fatoração ILU e no solver triangular permite que quaisquer outras ordenações sejam utilizadas, desde que haja um mapeamento entre a ordenação do simulador para a ordenação desejada além de um mapeamento contrário. Com isso, apesar do impacto de se ter mais indireções para tratar esse mapeamento, o simulador se torna mais flexível quando permite que outras ordenações possam ser aplicadas.

Formalmente, seja A uma matriz na ordenação induzida, P uma matriz de permutação que mapeia a ordenação induzida para a ordenação natural que será utilizada no pré-condicionador ILU e L e U os fatores ILU de PAP^T . Então, o pré-condicionador $M \sim A$ é dado por

$$M = P^T L U P \quad (4.1)$$

e, portanto, aplicar o pré-condicionador envolve resolver um sistema $Mu = v$, o qual se dá em três passos:

1. $v' = Pv$ (reordena o lado direito para a ordenação natural)
2. Resolver $LUy = v'$
3. $u = P^T y$ (reordenar a solução de volta para a ordenação do simulador)

Uma importante observação relacionada à fatoração ILU é que não necessariamente será preciso explicitamente se formar PAP^T para calcular os fatores ILU pois o simulador de nova geração utilizado para esses testes utiliza uma implementação de leitura da matriz baseada em linhas. Com isso, a ordenação pode ser aplicada durante a da execução da fatoração ILU.

Seja o vetor com o mapeamento da ordenação induzida para a ordenação natural chamado de *induzidaParaNat*, o vetor com a ordenação inversa chamado de *natParaInduzida* e seja ii o índice da linha da matriz que está sendo percorrida. Note que ii vai ser percorrido na ordenação natural, porém, para acessar qual é o índice da matriz na ordenação induzida, faz-se o mapeamento utilizando o vetor *natParaInduzida*. Essa instrução pode ser vista na linha 4 do Código 4.3. A partir do momento que se tem a *iiOrig*, deve-se saber quantas entradas naquela linha da matriz existem para serem armazenadas no vetor *alu* onde os fatores ILU são calculados e armazenados para futura utilização. Observe através da linha 5 do Código 4.3 que o método *numEnt* retorna um índice relativo à coluna do bloco, onde este está na ordenação induzida. Com isso, para acessar corretamente a coluna do bloco na ordenação natural, deve-se realizar o mapeamento utilizando o vetor *induzidaParaNat*, onde tal instrução pode ser observada na linha 8 do Código 4.3. A partir desse momento, copia-se o bloco correto para o vetor *alu* e passa-se para o próximo bloco da linha.

Nota-se que a introdução dos vetores de ordenação no Código 4.3 não afeta o desempenho da fatoração simbólica quanto em relação à quantidade de memória utilizada, pois, a cópia dos dados para o vetor *alu* é realizada mesmo sem os vetores de ordenação.

```
1 // loop sobre todas as linhas da matriz
2 for( int ii = 0; ii < n; ii++)
3     int col = 0;
4     int iiOrig = natParaInduzida[ ii ];
5     int numEntradasLinha = A.numEnt( iiOrig, col );
6     for( int i = 0; i < numEntradasLinha; i++)
7     {
8         int colNat = induzidaParaNat[ col ];
9         // copia o bloco (iiOrig,colNat) de A para o alu
10        // passa para o proximo bloco da linha
11    }
12 }
```

Código 4.3: Utilização dos vetores de ordenamento na fatoração ILU.

Algum problema de localidade de acesso aos dados pode acontecer a partir do momento que as entradas da matriz não são acessadas consecutivamente. Contudo, ao tirar proveito do fato que o acesso aos elementos da matriz são realizados linha a linha e todos os elementos de uma linha são acessados dentro de um laço interno, espera-se que haja alguma localidade dos dados. Se o mapeamento entre a ordenação do simulador para a ordenação natural seguir próximo das linhas da ordenação da matriz original, o que acontece quando a ordenação natural é utilizada em conjunto com um baixo número de células implícitas, o problema de localidade de dados não deve ser impactante no desempenho do solver linear.

A utilização da ordenação natural permite que a fatoração simbólica seja realizada somente uma vez ao início do processo de simulação pelo motivo que a estrutura da matriz a ser resolvida não será alterada durante a simulação. Contudo, a utilização da ordenação induzida faz com que haja a necessidade de se recalcular a fatoração simbólica sempre que houver a redistribuição das células FIM e IMPES, o que, no simulador utilizado, acontece no início de cada passo de tempo.

Capítulo 5

Testes e Resultados para a Estrutura Híbrida

Com o intuito de se comparar o desempenho das simulações utilizando o método AIM utilizando pré-condicionador com a ordenação natural e a ordenação induzida, foram executadas várias simulações com alguns casos reais. Uma simulação *black-oil* (*BO*) tem seu fluido modelado com somente 3 componentes: água, óleo e gás. Entretanto, alguns campos necessitam de uma modelagem de seu fluido com uma maior fidelidade em relação aos componentes do óleo. Por essa razão, utiliza-se uma modelagem chamada composicional (*EOS*) onde se consegue modelar a quantidade dos componentes do óleo de forma mais refinada. Uma característica dos campos do pré-sal é que eles contém CO₂ presente nos reservatórios e com isso a simulação tem que ser realizada utilizando uma modelagem EOS onde tipicamente podem ser realizadas com mais de 9 componentes. Para as simulações desse capítulo foram utilizados modelos de campos reais e modelos *proxy*, que são modelos que foram descaracterizados para esconder as características reais do reservatório por se tratarem de modelos sigilosos. Para os casos descritos, o tempo de simulação para C1 foi de 9 anos. Para os casos C2, C3 e C4, o simulador foi configurado para simular 13 anos. Para os casos C5 o tempo de simulação, por se tratar de um modelo com muitas células ativas, foi limitado para 3 anos. O caso C1 também foi simulado a partir de um modelo refinado. Nesse capítulo, o termo “padrão” refere-se ao pré-condicionador que utiliza a ordenação induzida. Por sua vez, a coluna que aparece o termo “híbrido” ao pré-condicionador que utiliza a ordenação natural.

Essas simulações foram executadas em um simulador de nova geração onde foram compilados dois binários diferentes. No primeiro binário não há qualquer alteração de código e foi utilizado para executar as simulações utilizando o pré-condicionador com a ordenação induzida. O segundo, por sua vez, foi compilado com as alterações necessárias para se implementar a estrutura de dados híbrida (ver 4.2), além da alteração nos núcleos da fatoração ILU e do solver triangular. Apesar do simulador

ter suporte a rodadas em paralelo utilizando técnicas MPI, todas as simulações foram executadas em modo escalar. Essa escolha foi feita porque os núcleos da fatoração ILU e do solver triangular são chamados para cada domínio gerado pela decomposição da malha, o que na prática somente diminui o tamanho da matriz a ser resolvida. Com isso, não faz sentido avaliar as duas ordenações em simulações que estão rodando em múltiplos nós, uma vez que o efeito da troca da ordenação não será observado por nós diferentes. Quando o simulador está executando uma simulação em um único nó, a decomposição de domínio gera somente um domínio e, por isso, os resultados apresentados foram provenientes de simulações rodadas em somente um nó.

A estação que executou as simulações foi uma HP-Z820 com 128GB de memória RAM, 2 processadores Intel Xeon ES-2687W a uma frequência de 3.10GHz totalizando 16 cores em um sistema operacional Windows 7 Enterprise com Service Pack 1.

5.1 Resultados Numéricos

Uma simulação de reservatórios é a solução de um sistema de equações diferenciais parciais (*EDP*) não-lineares que estão sendo resolvidas através da discretização por volumes finitos, nela, o meio poroso é dividido em células, o tempo é dividido em uma sucessão de intervalos, que são chamados de passos de tempo, as derivadas são substituídas por razões entre diferenças e, com isso, as EDPs são transformadas em uma sucessão de problemas algébricos. Durante um passo de tempo, diversas iterações não-lineares são executadas e, para cada iteração não-linear, várias iterações lineares são realizadas. Quando um passo de tempo não converge por qualquer motivo, o simulador está configurado para cortar esse passo de tempo e diminuí-lo. Note que, um corte é um passo de tempo que deu errado, e, portanto, ele realizou uma série de iterações não-lineares e lineares, que não tiveram uma utilidade prática. Com isso, um corte é de extrema importância durante uma simulação de um campo. Para todos os testes executados nesse capítulo, o solver de Krylov utilizado para solução das iterações lineares foi o OrthoMin.

A Tabela 5.1 mostra a quantidade de células ativas de cada campo, sendo que o menor tem 8×10^3 células ativas enquanto o maior tem 765×10^3 células ativas, o percentual de células que estavam como FIM ao final da simulação e o resultado dessas simulações. Alguns desses testes tiveram falha devido ou à convergência ou ao se exceder o tempo limite definido pelo simulador. O caso C4 BO, apesar do problema de convergência com ambos os pré-condicionadores, conseguiu avançar por mais 6 anos de simulação só pela utilização da ordenação natural.

A Tabela 5.2 mostra os resultados numéricos, como a quantidade de passos de

Caso	Células Ativas (x 1000)	% Células FIM		Status	
		Híbrido	Padrão	Híbrido	Padrão
C1 EOS	10	9,09%	7,90%	Success	Success
C1 R BO	111	1,43%	1,43%	Success	Success
C1 R EOS		1,49%	1,43%	Success	Success
C2 BO	8	12,03%	12,04%	Fail (Conv)	Fail (Max Time)
C2 EOS		12,22%	12,08%	Success	Fail (Conv)
C3 EOS	13	5,45%	5,46%	Success	Success
C4 BO	35	3,28%	4,04%	Fail (Conv)	Fail (Conv)
C4 EOS		3,34%	3,28%	Success	Fail (Max Time)
C5 BO	765	8,20%	8,20%	Success	Success
C5 EOS		8,20%	8,20%	Success	Success

Tabela 5.1: Campos utilizados para executar os testes computacionais e o resultado das simulações.

tempo (TS), cortes (Cuts), iterações não-lineares (NLin) e iterações lineares (Lin), para os seis casos que rodaram com sucesso para ambos os pré-condicionadores. Em geral, ao analisar a quantidade de passos de tempo, o pré-condicionador híbrido se sai melhor que o pré-condicionador padrão para todos os casos C1 e C3, enquanto o oposto ocorre para os dois casos C5. Pode-se observar também que as diferenças estão em torno de 10% aproximadamente, exceto pelos valores de passos de tempo e cortes obtidos para o C1 EOS e para todos os valores para o C3 EOS.

Caso	Híbrido				Padrão				Híbrido / Padrão			
	TS	Cuts	NLin	Lin	TS	Cuts	NLin	Lin	TS	Cuts	NLin	Lin
C1 EOS	405	136	1865	84674	464	179	2075	95469	0,87	0,76	0,90	0,89
C1 R BO	672	323	2975	124479	706	347	3145	131783	0,95	0,93	0,95	0,94
C1 R EOS	802	412	3681	118772	802	412	3857	120290	1,00	1,00	0,95	0,99
C3 EOS	1009	568	7328	279099	1225	727	9655	223792	0,82	0,78	0,76	1,25
C5 BO	180	73	1149	6299	165	66	1105	6225	1,09	1,11	1,04	1,01
C5 EOS	171	62	1111	10815	167	61	1093	11261	1,02	1,02	1,02	0,96

Tabela 5.2: Dados numéricos para os casos executados com sucesso.

A partir do último grupo de colunas da Tabela 5.2 pode-se observar que o pré-condicionador híbrido se saiu melhor para os casos C1 e C3 mas se saiu pior para os casos C5. Excetuando-se os valores dos passos de tempo e cortes para o caso C1 EOS e as iterações lineares e não lineares do caso C3 EOS as diferenças entre eles não é grande e fica em torno de 10%.

Para realizar a comparação entre os dois pré-condicionadores deve-se normalizar os resultados numéricos através de duas médias possíveis. A primeira média está relacionada com o número de iterações não-lineares por passos de tempo e a segunda média está relacionada com o número de iterações lineares por iterações não-lineares. Considera-se aqui que o valor do denominador da relação é o número de passos de tempo somado à quantidade de cortes que houve durante a simulação. A Tabela 5.3

usa como base a Tabela 5.2 para o cálculo das médias. Portanto, o primeiro grupo de colunas da Tabela 5.3 exibe as médias obtidas a partir dos dados numéricos do pré-condicionador híbrido da Tabela 5.2. A mesma regra se aplica para o pré-condicionador padrão da Tabela 5.3. Ao observar o último grupo de colunas da Tabela 5.3 pode-se notar que os valores chegam cada vez mais próximos com uma diferença de no máximo 5%. Porém, o resultado obtido para o caso C3 EOS mostra que o pré-condicionador híbrido realizou mais iterações lineares por iteração não-linear.

Caso	Híbrido		Padrão		Híbrido / Padrão	
	NLin/TS	Lin/NLin	NLin/TS	Lin/NLin	NLin/TS	Lin/NLin
C1 EOS	3,45	45,40	3,23	46,01	1,07	0,99
C1 R BO	2,99	41,84	2,99	41,90	1,00	1,00
C1 R EOS	3,03	32,27	3,18	31,19	0,95	1,03
C3 EOS	4,65	38,09	4,95	23,18	0,94	1,64
C5 BO	4,54	5,48	4,78	5,63	0,95	0,97
C5 EOS	4,77	9,73	4,79	10,30	0,99	0,94

Tabela 5.3: Médias utilizadas para se comparar os dois pré-condicionadores.

Simulações FIM foram executadas para os modelos da Tabela 5.1 que concluíram com sucesso com exceção do caso C1 R EOS, pois não existia modelo pronto disponível. Os resultados numéricos dessas simulações podem ser observados na Tabela 5.4, no primeiro grupo de colunas. Os outros dois grupos de colunas nessa tabela servem de comparação entre os valores dos pré-condicionadores híbrido e padrão em relação aos valores obtidos pela simulação FIM. O cálculo dessas relações é feita de forma direta, a partir dos dados mostrados na Tabela 5.2, para todas os dados numéricos exibidos. O primeiro grupo de colunas da Tabela 5.5 mostram relações semelhantes às utilizadas na Tabela 5.3 porém utilizando como base os valores do primeiro grupo de colunas da Tabela 5.4. Por sua vez, o segundo e terceiro grupo de colunas da Tabela 5.5 relacionam os valores da Tabela 5.3 com os valores do primeiro grupo de colunas da Tabela 5.5.

Analisando a tabela Tabela 5.4 pode-se notar que as simulações que utilizaram os pré-condicionadores com a ordenação padrão e híbrida realizaram muito mais passos de tempo, cortes, iterações lineares e não-lineares para todos os casos executados. Observa-se que para os casos C1 EOS, C1 R BO e C3 EOS, o pré-condicionador híbrido em relação às simulações FIM realizou menos passos de tempo, cortes, iterações lineares e não lineares que o pré-condicionador padrão em relação às simulação FIM. A única exceção está na quantidade de iterações lineares do caso C3 EOS. Para os casos C5, as diferenças de ambos pré-condicionadores não está tão grande quando comparadas com os outros casos. Em relação às simulações FIM, o pré-condicionador híbrido realizou em média 33% mais passos de tempo,

enquanto o pré-condicionador padrão realizou em média 26% mais passos de tempo. Essa mesma avaliação quando aplicada aos cortes e iterações não-lineares mostra padrão similar, onde o pré-condicionador híbrido realiza mais cortes e iterações não-lineares que o pré-condicionador padrão em média. Entretanto, para a relação das iterações lineares, observa-se que o pré-condicionador híbrido realizou 23% a mais enquanto o pré-condicionador padrão realizou em torno de 25%.

Caso	FIM				Híbrido / FIM				Padrão / FIM			
	TS	Cuts	NLin	Lin	TS	Cuts	NLin	Lin	TS	Cuts	NLin	Lin
C1 EOS	195	5	427	13478	2,08	27	4,37	6,28	2,38	35,80	4,86	7,08
C1 R BO	186	0	228	8291	3,61	-	13,05	15,01	3,80	-	13,79	15,89
C3 EOS	241	2	929	34732	4,19	284	7,89	8,04	5,08	363,50	10,39	6,44
C5 BO	130	48	921	5532	1,38	1,5	1,25	1,14	1,27	1,38	1,20	1,13
C5 EOS	133	44	919	8152	1,29	1,4	1,21	1,33	1,26	1,39	1,19	1,38

Tabela 5.4: Dados numéricos para os mesmos casos da Tabela 5.2 para as simulações FIM.

Caso	FIM		Híbrido / FIM		Padrão / FIM	
	NLin/TS	Lin/NLin	NLin/TS	Lin/NLin	NLin/TS	Lin/NLin
C1 EOS	2,14	31,56	1,61	1,44	1,51	1,46
C1 R BO	1,23	36,36	2,44	1,15	2,44	1,15
C3 EOS	3,82	37,39	1,22	1,02	1,29	0,62
C5 BO	5,17	6,01	0,88	0,91	0,92	0,94
C5 EOS	5,19	8,87	0,92	1,10	0,92	1,16

Tabela 5.5: Médias das iterações não-lineares por passo de tempo e iterações lineares por iterações não-lineares para as simulações FIM.

Uma outra comparação possível entre as simulações FIM e as simulações com os dois pré-condicionadores é baseada na normalização entre a quantidade de iterações não-lineares sobre passos de tempo e lineares sobre iterações não-lineares. A Tabela 5.5 mostra o resultado dessa comparação. Para os casos C1 EOS, C1 R BO e C3 EOS, pode-se observar que eles realizaram em ambos os pré-condicionadores mais iterações não-lineares sobre passos de tempo quando comparados com os mesmos valores obtidos pela simulação FIM. Entretanto, para os casos C5 BO e C5 EOS, observa-se um padrão diferente, onde os pré-condicionadores híbrido e padrão realizaram em torno de 10% menos iterações não-lineares por passo de tempo que as simulações FIM. Ao analisar a outra relação, vemos que o caso C1 EOS se saiu pior para os pré-condicionadores híbrido e padrão pois realizou mais iterações lineares por não-lineares em até 45% em média. Já o caso C1 R BO e C5 EOS também obtiveram um resultado ruim porém em um patamar menor. Suas simulações realizaram, em média, somente 15% a mais de iterações lineares por não-lineares. O caso C3 EOS, por sua vez, teve um resultado muito melhor com o pré-condicionador padrão que a simulação FIM pois executou quase 40% a menos iterações lineares por não-lineares

enquanto o pré-condicionador híbrido realizou somente 2% a mais. Por último, para o caso C5 BO pode-se observar que o pré-condicionador híbrido obteve resultados melhores quando comparado com os obtidos pela simulação FIM pois realizou 9% a menos de iterações lineares sobre não-lineares. Esse resultado também pode ser observado para o pré-condicionador padrão, entretanto, este realizou somente 6% a menos ter iterações lineares sobre não-lineares.

Como os testes foram executados em binários que foram compilados a partir de códigos diferentes, um outro teste foi realizado a fim de se avaliar explicitamente a qualidade do pré-condicionador eliminando os possíveis ganhos com a utilização de código novo. O teste consiste em utilizar o binário com as estruturas que permitem a utilização da ordenação natural e executar uma simulação aplicando a ordenação induzida. Deve-se observar que a versão do simulador que executou os primeiros testes é diferente da versão do simulador que executará esse teste. Portanto, podem haver mudanças na quantidade de iterações não-lineares, lineares e passos de tempo envolvendo o caso alvo.

O caso C5 EOS foi escolhido como caso alvo para realizar esse teste e seus resultados numéricos são mostrados na Tabela 5.6. A linha referente à média relaciona os valores obtidos pelo pré-condicionador híbrido em relação ao pré-condicionador padrão. A última coluna dessa tabela indica o porcentual de células FIM ao final da simulação. Deve-se comparar os resultados dessa tabela com os observados na Tabela 5.2, onde, para esse caso, o pré-condicionador híbrido executou 2% a mais passos de tempo, cortes e iterações não-lineares enquanto realizou 4% a menos iterações lineares em relação ao pré-condicionador padrão. Ao observar as médias na Tabela 5.6 pode-se observar que o pré-condicionador híbrido realizou 15% a menos passos de tempo que o pré-condicionador padrão. Entretanto, ao se comparar com os resultados obtidos na tabela Tabela 5.3, observa-se que a relação de iterações não-lineares por passos de tempo, que estava em torno de 1% a menos para o pré-condicionador híbrido aumentou para em torno de 6% e, em relação às iterações lineares sobre não-lineares observa-se que esse número melhorou de 6% para 13% menos iterações. Portanto, atribui-se essa diferença de 17% em média para as melhorias em outros núcleos que foram desenvolvidas no simulador para o código binário utilizado.

	TS	Cuts	NLin	Lin	NLin/TS	Lin/NLin	%Imp
Híbrido	150	43	969	5224	5,02	5,39	8,42%
Padrão	177	54	1091	6769	4,72	6,20	8,43%
Média	0,85	0,80	0,89	0,77	1,06	0,87	

Tabela 5.6: Dados numéricos para o caso C5 EOS.

5.2 Resultados de Tempo de Execução

Para apresentar resultados relativos ao tempo de execução, será examinada a relevância da fatoração simbólica (FS) no desempenho de um solver linear baseado em ILU. A Tabela 5.7 compara o total de tempo gasto nesse núcleo para as simulações com o pré-condicionador padrão e simulações FIM. Para as simulações FIM, a fatoração simbólica é realizada somente uma vez por execução da simulação, com isso, sua contribuição para o tempo total do solver linear é negligenciável para todos os casos exceto pelos casos C5. Essa exceção se dá pelo fato que os casos C5 estão rodando por somente 3 anos ao invés dos 50 anos que foram descritos para o caso. Portanto, se a simulação tivesse sido executada para o tempo original o tempo gasto com a fatoração simbólica seria, então, negligenciável.

Caso	Padrão	FIM
C1 EOS	27,53%	0,03%
C1 R BO	16,26%	0,10%
C1 R EOS	26,69%	-
C2 BO	-	0,06%
C2 EOS	40,43%	0,05%
C3 EOS	39,50%	0,01%
C4 BO	11,00%	0,03%
C4 EOS	-	0,02%
C5 BO	50,46%	3,59%
C5 EOS	52,58%	2,03%

Tabela 5.7: Percentual utilizado pela fatoração simbólica no tempo do solver linear em simulações com pré-condicionador padrão e simulações FIM.

Por sua vez, para os casos rodando em AIM, a fatoração simbólica é executada a cada vez que a distribuição das células FIM/IMPES é alterada, tornando-se muito significativa, mais que a metade do tempo do solver linear para as simulações dos casos C5. Note que, para essas simulações, a média do número de iterações lineares por chamada do solver linear é muito pequena (vide Tabela 5.3), o que explica a alta quantidade de tempo gasto na fatoração simbólica, uma vez que o tempo gasto em outros núcleos não conseguem amortizar esse custo.

Com o objetivo de se avaliar o peso da fatoração simbólica no pré-condicionador híbrido foram desenvolvidos dois pré-condicionadores híbridos. O primeiro, aplica a ordenação natural mas mantém a fatoração simbólica a ser executada a cada mudança na distribuição das células FIM/IMPES, comportamento equivalente ao pré-condicionador padrão. O segundo, aplica a ordenação natural mas realiza a fatoração simbólica somente uma vez durante a simulação completa, o que é equivalente às simulações FIM. Com isso, pode-se avaliar, de fato, o quanto a fatoração simbólica afeta o tempo da simulação utilizando o pré-condicionador híbrido. Não seria possível comparar-se os tempos obtidos pela simulação utilizando pré-condicionador

padrão com os tempos obtidos utilizando pré-condicionador híbrido com a fatoração simbólica sendo executada uma vez só pois entre estaria levando em conta também o peso das mudanças relativas à ordenação aplicada. Nas tabelas abaixo a coluna Híbrido com FS mostra os resultados da simulação que foi executada utilizando o pré-condicionador híbrido que repete a fatoração simbólica, e, por sua vez, a coluna Híbrido sem FS mostra os resultados da simulação considerando que ela é executada somente uma vez no início da simulação.

Caso	Híbrido com FS			Híbrido sem FS			Padrão		
	Asset	Lin	% Lin	Asset	Lin	% Lin	Asset	Lin	% Lin
C1 EOS	2071	961	46,4%	1935	825	42,6%	2420	1257	51,9%
C1 R BO	8671	4811	55,5%	7764	3904	50,3%	8233	4237	51,5%
C1 R EOS	26263	10556	40,2%	22471	6764	30,1%	22820	7839	34,4%
C3 EOS	4853	2069	42,6%	4496	1712	38,1%	6177	2747	44,5%
C5 BO	8348	3113	37,3%	7102	1867	26,3%	7927	3123	39,4%
C5 EOS	31968	11121	34,8%	28146	7299	25,9%	39168	19349	49,4%

Tabela 5.8: Tempos totais de simulação e solver linear para os pré-condicionadores híbrido e padrão.

Caso	Padrão / Híbrido com FS		Padrão / Híbrido sem FS	
	Asset	Linear Solver	Asset	Linear Solver
C1 EOS	1,17	1,31	1,25	1,52
C1 R BO	0,95	0,88	1,06	1,09
C1 R EOS	0,87	0,74	1,02	1,16
C3 EOS	1,27	1,33	1,37	1,60
C5 BO	0,95	1,00	1,12	1,67
C5 EOS	1,23	1,74	1,39	2,65

Tabela 5.9: Razão entre os tempos dos pré-condicionadores padrão e híbrido.

Com isso, comparações de tempo entre as versões dos pré-condicionadores híbrido e padrão são exibidas na Tabela 5.8 e na Tabela 5.9 para os casos que acabaram com sucesso até o final. O valor que aparece na coluna Asset para cada caso significa o tempo total da simulação em segundos. Por sua vez, o valor que aparece na coluna Lin significa de quanto o linear solver gastou no tempo total. E por fim, a última coluna de cada grupo mostra a porcentagem que indica a participação do solver linear na simulação total. O último grupo da Tabela 5.8 mostra os mesmos dados obtidos para as simulações executadas com o pré-condicionador padrão. Na Tabela 5.9, são apresentados as razões entre os tempos do pré-condicionador padrão em relação ao pré-condicionador híbrido com e sem o efeito da fatoração simbólica na simulação.

Sem considerar os efeitos do problema da fatoração simbólica, chega-se a conclusão que o pré-condicionador híbrido ganha em três casos com um speed-up de até 1,74x no solver linear para o C5 EOS. Para o C5 BO não houve melhoria nem

piora. Para os outros dois casos, o pré-condicionador padrão foi melhor e obteve um speed-up máximo de até 1,35x no caso C1 R EOS. Porém, quando os efeitos da fatoração simbólica são mitigados, o pré-condicionador híbrido ultrapassa o pré-condicionador padrão em todos os casos com uma margem alta em quatro dos seis casos, chegando a obter um speed-up de até 2,65x para o caso C5 BO.

Caso	FIM			FIM / Híbrido sem FS		FIM / Padrão	
	Asset	Linear Solver	% Linear Solver	Asset	Linear Solver	Asset	Linear Solver
C1 EOS	1135	686	60,4%	0,59	0,83	0,47	0,55
C1 R BO	1840	1078	58,6%	0,24	0,28	0,22	0,25
C3 EOS	2764	2009	72,7%	0,61	1,17	0,45	0,73
C5 BO	11192	3658	32,7%	1,58	1,96	1,41	1,17
C5 EOS	69248	29819	43,1%	2,46	4,09	1,77	1,54

Tabela 5.10: Informação de tempo das simulações FIM.

Ao se comparar os resultados de tempo do pré-condicionador híbrido sem fatoração simbólica e do pré-condicionador padrão com os resultados de tempo para as simulações FIM pode-se observar na Tabela 5.10 que, como esperado, a fatia de tempo para o solver linear é maior para as simulações utilizando FIM do que para as simulações AIM, seja pela utilização do pré-condicionador híbrido ou pela utilização do pré-condicionador padrão. Em relação ao tempo da solução, as simulações FIM estão muito mais rápidas que o AIM para os casos C1 e C3, devido aos problemas de convergência encontrados nas simulações AIM, independente do pré-condicionador híbrido ou padrão. Entretanto, para os casos C5, as simulações utilizando o AIM rodaram muito mais rápido que o FIM apesar do aumento moderado na quantidade de passos de tempo e iterações (ver Tabela 5.4). Pode-se notar também que para os casos C5 as simulações AIM que utilizaram o pré-condicionador híbrido obtiveram um melhor resultado, onde o speed-up máximo em relação ao tempo da solução alcançou 2,46x para o caso C5 EOS.

Obteve-se a partir do resultado do tempo total da simulação obtido pelo teste do final da seção anterior, que tem como objetivo comparar os dois pré-condicionadores em um mesmo binário, um speed-up de 1,17 do pré-condicionador híbrido em relação ao pré-condicionador padrão. Esse resultado está alinhado com o speed-up encontrado na última linha da Tabela 5.9. Atribui-se, novamente, a razão da diferença entre os speed-ups como sendo as diferentes modificações realizadas no código de outros núcleos entre as duas versões utilizadas.

Capítulo 6

Conclusões

Foram feitos o estudo e a implementação de um pré-condicionador para sistemas lineares para o método AIM. Os estudos realizados mostram que, através da análise dos grafos gerados por matrizes similares às matrizes obtidas durante uma simulação, quando se utiliza a ordenação natural a quantidade de não-zeros da inversa aproximada é maior que quando se utiliza a ordenação induzida. A implementação do pré-condicionador, que foi acoplado a um simulador de nova geração e testado com casos reais em simulações completas, é baseado em uma abordagem que mantém a ordenação natural ao se formar os fatores ILU, usando estruturas de dados especiais para tratar matrizes geradas com blocos de tamanhos híbridos.

Foi desenvolvido um código que possibilitou ao simulador utilizado manter sua ordenação nativa (ordenação induzida), a qual resolve conjuntos de equações onde as matrizes que são geradas a partir delas tem blocos de tamanho constante, para problemas AIM no solver iterativo. Além disso, um novo pré-condicionador foi desenvolvido para utilizar a ordenação natural. Para essa nova ordenação, como os blocos da matriz gerada passam a ter tamanhos híbridos, foi desenvolvida uma nova estrutura de dados para guardar as informações necessárias para realizar corretamente a fatoração ILU e o solver triangular. Foram também realizadas alterações nos códigos da fatoração ILU e do solver triangular para que eles recebessem como entrada dois vetores que fazem o mapeamento entre a ordenação induzida para a ordenação natural e a ordenação inversa.

As simulações AIM com o pré-condicionador que utiliza a ordenação induzida realizavam uma fatoração simbólica a cada mudança na distribuição das células FIM e IMPES, ou seja, no início de cada passo de tempo. Entretanto, a fatoração simbólica é realizada uma vez só e comportamento similar pode ser obtido com a utilização de um pré-condicionador que utilizasse a ordenação natural durante as simulações AIM. Isso foi possível pois com a ordenação natural, não há mudança na estrutura da matriz a ser resolvida. Testes que foram executados somente para avaliar a influência da fatoração simbólica no pré-condicionador híbrido mostraram

que, em relação ao pré-condicionador padrão, ele obteve speed-up de até 1,39 para o tempo total de simulação e 2,65 para o solver linear para o caso com maior número de células ativas.

Vários testes foram executados para casos black-oil e composicionais EOS onde foram comparados os desempenhos dos pré-condicionadores híbrido e padrão. Pode-se chegar à conclusão que o pré-condicionador híbrido é mais robusto, uma vez que ele pode completar mais simulações e levou menos passos de tempo e iterações para os casos com maior número de células ativas. A simulação do caso C5 EOS, com o pré-condicionador híbrido, obteve speed-up de 2,46 no tempo total da simulação quando comparado com o tempo total da mesma simulação FIM. Alguns testes, mesmo utilizando o pré-condicionador híbrido, tiveram problemas de convergência. Porém, esses testes e os testes que simularam até o final são executados com sucesso quando rodados por outros simuladores comerciais. Com isso, assume-se que as técnicas e algoritmos relativos à implementação do método AIM do simulador de nova geração escolhido estão, de alguma forma, atrapalhando no desempenho das suas simulações.

Referências Bibliográficas

- [1] ROSA, A., DE SOUZA CARVALHO, R., XAVIER, J. *Engenharia de reservatórios de petróleo*. <https://books.google.com.br/books?id=aSUNAQAACAAJ>, Interciência, 2006. ISBN: 9788571931350.
- [2] BRIDSON, R., TANG, W.-P. “A structural diagnosis of some IC orderings”, *SIAM Journal on Scientific Computing*, v. 22, n. 5, pp. 1527–1532, 2001.
- [3] MATTAX, C. C., DALTON, R. L., OTHERS. “Reservoir simulation (includes associated papers 21606 and 21620)”, *Journal of Petroleum Technology*, v. 42, n. 06, pp. 692–695, 1990.
- [4] DOUGLAS JR, J., PEACEMAN, D., RACHFORD JR, H., et al. “A method for calculating multi-dimensional immiscible displacement”, *Transactions of the AIME*, v. 216, pp. 297–308, 1959.
- [5] BLAIR, P., WEINAUG, C., OTHERS. “Solution of two-phase flow problems using implicit difference equations”, *Society of Petroleum Engineers Journal*, v. 9, n. 04, pp. 417–424, 1969.
- [6] AZIZ, K., AZIZ, K., SETTARI, A. *Petroleum reservoir simulation*. <https://books.google.com.br/books?id=GJ5TAAAAMAAJ>, Applied Science Publishers, 1979. ISBN: 9780853347873.
- [7] SETTARI, A., AZIZ, K. “Treatment of nonlinear terms in the numerical solution of partial differential equations for multiphase flow in porous media”, *International Journal of Multiphase Flow*, v. 1, n. 6, pp. 817–844, 1975.
- [8] MACHADO, M. V. B. “Modelagem e Simulação Numérica de Reservatórios de Petróleo”. September 2013.
- [9] SHELDON, J., CARDWELL JR, W., OTHERS. “One-dimensional, incompressible, noncapillary, two-phase fluid flow in a porous medium”, *Transactions of the AIME*, v. 216, pp. 290–296, 1959.

- [10] STONE, H., GARDER JR, A., OTHERS. “Analysis of gas-cap or dissolved-gas drive reservoirs”, *Society of Petroleum Engineers Journal*, v. 1, n. 02, pp. 92–104, 1961.
- [11] RUSSELL, T., OTHERS. “Stability analysis and switching criteria for adaptive implicit methods based on the CFL condition”. In: *SPE Symposium on Reservoir Simulation*. Society of Petroleum Engineers, 1989.
- [12] THOMAS, G., THURNAU, D., OTHERS. “Reservoir simulation using an adaptive implicit method”, *Society of Petroleum Engineers Journal*, v. 23, n. 05, pp. 759–768, 1983.
- [13] FORSYTH, P., SAMMON, P. “Practical considerations for adaptive implicit methods in reservoir simulation”, *Journal of Computational Physics*, v. 62, n. 2, pp. 265–281, 1986.
- [14] MARCONDES, F., MALISKA, C. R., ZAMBALDI, M. C. “A comparative study of implicit and explicit methods using unstructured voronoi meshes in petroleum reservoir simulation”, *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, v. 31, n. 4, pp. 353–361, 2009.
- [15] YOUNG, L., RUSSELL, T., OTHERS. “Implementation of an adaptive implicit method”. In: *SPE Symposium on Reservoir Simulation*. Society of Petroleum Engineers, 1993.
- [16] FUNG, L. S., COLLINS, D. A., NGHIEM, L. X., et al. “An Adaptive-Implicit Switching Criterion Based on Numerical Stability Analysis (includes associated paper 18727)”, *SPE reservoir engineering*, v. 4, n. 01, pp. 45–51, 1989.
- [17] YANNAKAKIS, M. “Computing the minimum fill-in is NP-complete”, *SIAM Journal on Algebraic Discrete Methods*, v. 2, n. 1, pp. 77–79, 1981.
- [18] SAAD, Y. *Iterative methods for sparse linear systems*. <http://www-users.cs.umn.edu/saad/books.html>, SIAM, 2003.
- [19] MEURANT, G. *Computer Solution of Large Linear Systems*. Studies in Mathematics and its Applications. <https://books.google.com.br/books?id=fSqfb5a3WrwC>, Elsevier Science, 1999. ISBN: 9780080529516.
- [20] DUFF, I. S., MEURANT, G. A. “The effect of ordering on preconditioned conjugate gradients”, *BIT Numerical Mathematics*, v. 29, n. 4, pp. 635–657, 1989.

- [21] IM, E.-J. *Optimizing the Performance of Sparse Matrix-Vector Multiplication*. Tese de Doutorado, EECS Department, University of California, Berkeley, Jun 2000. Disponível em: <<http://www2.eecs.berkeley.edu/Pubs/TechRpts/2000/5556.html>>.