



**UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE TECNOLOGIA E GEOCIÊNCIAS
DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL
CURSO DE GRADUAÇÃO EM ENGENHARIA CIVIL**

ALEXANDRE DE SOUZA JUNIOR

**DESENVOLVIMENTO DE SIMULADOR DE FLUXO
PARA RESERVATÓRIO DE PETRÓLEO**

**RECIFE
2017**

ALEXANDRE DE SOUZA JUNIOR

**DESENVOLVIMENTO DE SIMULADOR DE FLUXO
PARA RESERVATÓRIO DE PETRÓLEO**

Monografia apresentada à Universidade Federal de Pernambuco como parte dos requisitos para obtenção de grau de Bacharel em Engenharia Civil.

Área de concentração: Engenharia de Petróleo

Orientador: Prof. Dr. Bernardo Horowitz

RECIFE

2017

Catálogo na fonte
Bibliotecária Margareth Malta, CRB-4 / 1198

S729e Souza Junior, Alexandre de
Desenvolvimento de Simulador de Fluxo para Reservatório de Petróleo.
/ Alexandre de Souza Junior - 2017.
57 folhas, Ils. e Tabs.

Orientador: Prof. Dr. Bernardo Horowitz.

TCC (Graduação) – Universidade Federal de Pernambuco. CTG.
Departamento de Graduação em Engenharia Civil e Ambiental. 2017.
Inclui Referências e Apêndices.

1. Engenharia Civil. 2 Petróleo. 3. Recuperação. 4. Injeção.
5. Simulação. I. Horowitz, Bernardo (Orientador). II. Título.

UFPE

624 CDD (22. ed.) BCTG/2017-422



SERVIÇO PÚBLICO FEDERAL
UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE TECNOLOGIA E GEOCIÊNCIAS
DEPARTAMENTO DE ENGENHARIA CIVIL E AMBIENTAL
COORDENAÇÃO DO CURSO DE GRADUAÇÃO EM ENGENHARIA CIVIL

ATA DA DEFESA DO TRABALHO DE CONCLUSÃO DE CURSO PARA CONCESSÃO DO GRAU DE ENGENHEIRO CIVIL

CANDIDATO: Alexandre de Souza Junior

BANCA EXAMINADORA:

Orientador: Prof^o Bernardo Horowitz

Examinador 1: Prof^o Leonardo Correia de Oliveira

Examinador 2: Prof^a Liliane de Allan Fonseca

TÍTULO DO TRABALHO DE CONCLUSÃO DE CURSO:

Desenvolvimento de Simulador de Fluxo para Reservatório de Petróleo.

LOCAL:

DATA: ____/____/____ **HORÁRIO DE INÍCIO:** _____.

Em sessão pública, após exposição de cerca de 30 minutos, o(s) candidato(s) foi (foram) arguido(s) oralmente pelos membros da banca com NOTA: _____(deixar 'Exame Final', quando for o caso).

1) () aprovado(s) (nota $\geq 7,0$), pois foi demonstrado suficiência de conhecimento e capacidade de sistematização no tema da monografia e o texto do trabalho aceito.

As revisões observadas pela banca examinadora deverão ser corrigidas e verificadas pelo orientador no prazo máximo de 30 dias (o verso da folha da ata poderá ser utilizado para pontuar revisões).

O trabalho com nota no seguinte intervalo, **$3,0 \leq \text{nota} < 7,0$** , será reapresentado, gerando-se uma nota ata; sendo o trabalho aprovado na reapresentação, o aluno será considerado **aprovado com exame final**.

2) () reprovado(s). (nota $< 3,0$)

Na forma regulamentar foi lavrada a presente ata que é assinada pelos membros da banca e pelo(s) candidato(s).

Recife, de de 20....

Orientador:

Avaliador 1:

Avaliador 2:

Candidato:

Aos meus pais, irmãs, parentes, amigos e professores.

AGRADECIMENTOS

Agradeço a Deus por ter me dado saúde e sabedoria para superar as dificuldades e conseguir a realização deste trabalho.

À UFPE por ter me dado a oportunidade de realizar este curso, através de seu corpo docente composto de profissionais qualificados, que, com a devida atenção e dedicação, trabalham incansavelmente para que nós, alunos, possamos contar com um ensino de qualidade.

Agradeço ao professor Bernardo Horowitz pela orientação e ajuda que me foram dadas, e também ao professor Jonathan Teixeira pela disposição e pelos ensinamentos referentes ao simulador MRST, essencial para o desenvolvimento deste trabalho.

Aos meus pais, pelo amor, carinho, paciência e seus ensinamentos, por confiarem em mim e estarem do meu lado em todos os momentos da vida, e por não medirem esforços para que eu pudesse levar meus estudos adiante.

RESUMO

O registro da participação do petróleo na vida do homem remonta a tempos bíblicos e, desde então, com os grandes avanços tecnológicos, o petróleo foi se impondo como fonte de energia. Com o advento da petroquímica, além da grande utilização dos seus derivados, centenas de novos compostos são produzidos, desde plásticos até produtos farmacêuticos e cosméticos. Com isso, o petróleo, além de produzir combustível, passou a ser peça fundamental às facilidades e comodidades da vida moderna. A engenharia de reservatórios, por sua vez, se preocupa basicamente com a retirada dos fluidos do interior das rochas, de modo que eles possam ser conduzidos até a superfície, desenvolvendo métodos de recuperação para se obter uma produção maior do que aquela que se obteria, caso apenas a energia natural do reservatório fosse utilizada. Atualmente, a injeção de água consiste em um dos métodos convencionais de recuperação, cujas tecnologias são bem conhecidas e cujo grau de confiança na aplicação é bastante elevado. Para o gerenciamento da técnica de injeção de água, faz-se necessária a simulação do fluxo dinâmico dos fluidos na rocha-reservatório a fim de decidir quais as vazões de injeção de água nos poços injetores serão adotadas para maximizar o retorno, impedindo ao máximo a irrupção precoce de água nos poços produtores e, conseqüentemente, avaliando a viabilidade econômica do projeto. É necessária a solução de um sistema de equações diferenciais parciais não-lineares, sendo utilizados para isto simuladores de fluxo, desenvolvidos através da ferramenta MRST, composta de um conjunto extenso de rotinas em MATLAB, que permite livre acesso ao código. Em comparação com um código simples também em MATLAB para um fluxo bifásico óleo-água, utilizando Diferenças Finitas e o método IMPES (*Implicit Pressure Explicit Saturation*), foram obtidos resultados para as pressões e saturações dos fluidos, sendo observada a semelhança por ambos os métodos para o modelo apresentado.

Palavras-chaves: Petróleo. Recuperação. Injeção. Simulação. MATLAB. MRST.

ABSTRACT

The record of oil's share in humanity dates back to biblical times, and since then, with the great technological advances, oil has been imposed as a source of energy, and with the advent of petrochemicals, in addition to the great use of its derivatives, hundreds of new compounds are produced, from plastics to pharmaceuticals and cosmetics. As a result, oil, in addition to producing fuel, has become fundamental to the facilities and amenities of modern life. Reservoir engineering, in turn, is concerned primarily with a withdrawal of the fluids from the interior of the rocks, as they are carried out to a surface, and developed recovery methods to obtain a production greater than that obtained if only a natural energy from the reservoir were used. Currently, a consistent water injection in one of the conventional recovery methods, important issues are well known and the degree of confidence in the application is quite high. For the management of the water injection technique, a simulation of the dynamic flow of fluids in the reservoir rock in order to decide which are injection of water rates in the injector wells adopted to maximize the return and to prevent the water cut in the producing wells and also evaluate an economic viability of the project. A solution of a system of nonlinear partial differential equations is required, being used for the flow simulator, developed with the MRST tool, composed of an extensive set of routines in MATLAB language, which allows access to the code. Compared to a simple code in MATLAB of a two-phase oil flow, using Finite Difference and IMPES Method (Implicit Pressure Explicit Saturation), oil pressure and water saturation results were obtained and a similarity was observed by all the methods for the presented model.

Keywords: Oil. Recovery. Injection. Simulation. MATLAB. MRST.

LISTA DE ILUSTRAÇÕES

Figura 1 – Esquema típico de um reservatório de petróleo	12
Figura 2 – Esquema de injeção de base ou periférica	13
Figura 3 – Etapas principais do processo de simulação de um reservatório	14
Figura 4 – Diagrama de fases de uma mistura de hidrocarbonetos	17
Figura 5 – Elemento de meio poroso	19
Figura 6 – Sistema de “ <i>grid-blocks</i> ” igualmente espaçados	23
Figura 7 – Sistema de “ <i>grid-blocks</i> ” desigualmente espaçados	24
Figura 8 – Exemplo de instabilidade numérica com a formulação explícita	25
Figura 9 – Exemplo de estabilidade numérica com a formulação implícita	26
Figura 10 – Esquema do <i>grid</i> composto por 10 células	28
Figura 11 – Pontos das curvas de permeabilidade relativa, função da saturação ..	30
Figura 12 – Pressão P_o em $t = 5s$ (KLEPPE)	31
Figura 13 – Pressão P_o em $t = 50s$ (KLEPPE)	31
Figura 14 – Pressão P_o em $t = 135s$ (KLEPPE)	31
Figura 15 – Pressão P_o em $t = 240s$ (KLEPPE)	31
Figura 16 – Pressão P_o em $t = 300s$ (KLEPPE)	31
Figura 17 – Pressão P_o em $t = 390s$ (KLEPPE)	31
Figura 18 – Saturação S_w em $t = 5s$ (KLEPPE)	32
Figura 19 – Saturação S_w em $t = 65s$ (KLEPPE)	32
Figura 20 – Saturação S_w em $t = 160s$ (KLEPPE)	32
Figura 21 – Saturação S_w em $t = 240s$ (KLEPPE)	32
Figura 22 – Saturação S_w em $t = 320s$ (KLEPPE)	32
Figura 23 – Saturação S_w em $t = 397s$ (KLEPPE)	32
Figura 24 – Curvas de permeabilidade relativa ajustadas aos pontos da Tabela 2	33
Figura 25 – Pressão P_o em $t = 5s$ (MRST)	33
Figura 26 – Pressão P_o em $t = 60s$ (MRST)	33
Figura 27 – Pressão P_o em $t = 120s$ (MRST)	34
Figura 28 – Pressão P_o em $t = 250s$ (MRST)	34
Figura 29 – Pressão P_o em $t = 300s$ (MRST)	34
Figura 30 – Pressão P_o em $t = 400s$ (MRST)	34
Figura 31 – Saturação S_w em $t = 2,5s$ (MRST)	34
Figura 32 – Saturação S_w em $t = 75s$ (MRST)	34

Figura 33 – Saturação S_w em $t = 135s$ (MRST)	35
Figura 34 – Saturação S_w em $t = 215s$ (MRST)	35
Figura 35 – Saturação S_w em $t = 305s$ (MRST)	35
Figura 36 – Saturação S_w em $t = 400s$ (MRST)	35
Figura 37 – Pressões P_o obtidas por ambas as simulações no instante $t = 250s$	37
Figura 38 – Saturações S_w obtidas por ambas as simulações no instante $t = 250s$	37
Figura 39 – Pressões P_o obtidas por ambas as simulações no instante $t = 400s$	38
Figura 40 – Saturações S_w obtidas por ambas as simulações no instante $t = 400s$	39

LISTA DE TABELAS

Tabela 1 – Dados do Sistema, Condições Iniciais e de Contorno	29
Tabela 2 – Pontos das curvas de permeabilidade relativa e pressão capilar, funções da saturação	30
Tabela 3 – Pontos das curvas de fator volume de formação e viscosidade, funções da pressão	30
Tabela 4 – Pressões P_o e Saturações S_w obtidas por ambas as simulações em $t = 250s$...	36
Tabela 5 – Pressões P_o e Saturações S_w obtidas por ambas as simulações em $t = 400s$...	38

SUMÁRIO

1	INTRODUÇÃO	12
1.1	Justificativa e motivação	14
1.2	Objetivos gerais e específicos	15
2	REFERENCIAL TEÓRICO	16
2.1	Introdução à Engenharia de Petróleo	16
2.2	Física do fluxo multifásico em meios porosos	18
2.3	Equações Diferenciais de Difusão	20
3	MATERIAIS E MÉTODOS	22
3.1	Método das Diferenças Finitas	22
3.1.1	Aproximação da primeira derivada	22
3.1.2	Aproximação da segunda derivada	23
3.1.3	Discretização do Espaço	23
3.1.4	Discretização do Tempo	25
3.1.4.1	<i>Métodos Explícitos</i>	25
3.1.4.2	<i>Métodos Implícitos</i>	26
3.1.5	Condições de Contorno	26
3.1.5.1	<i>Condição de Contorno de Dirichlet</i>	26
3.1.5.2	<i>Condição de Contorno de Neumann</i>	27
3.2	Linguagem MATLAB	27
4	RESULTADOS	28
4.1	Codificação de problema de fluxo bifásico óleo/água com formulação IMPES	28
4.2	Simulação do problema anterior com utilização da <i>toolbox</i> MRST	33
4.3	Comparação dos resultados obtidos pelos métodos anteriores	36
5	CONCLUSÕES	40
	REFERÊNCIAS BIBLIOGRÁFICAS	41
	APÊNDICE I – Codificação de fluxo bifásico óleo-água pelo método IMPES	42
	APÊNDICE II – Codificação de fluxo bifásico óleo-água pelo MRST	51

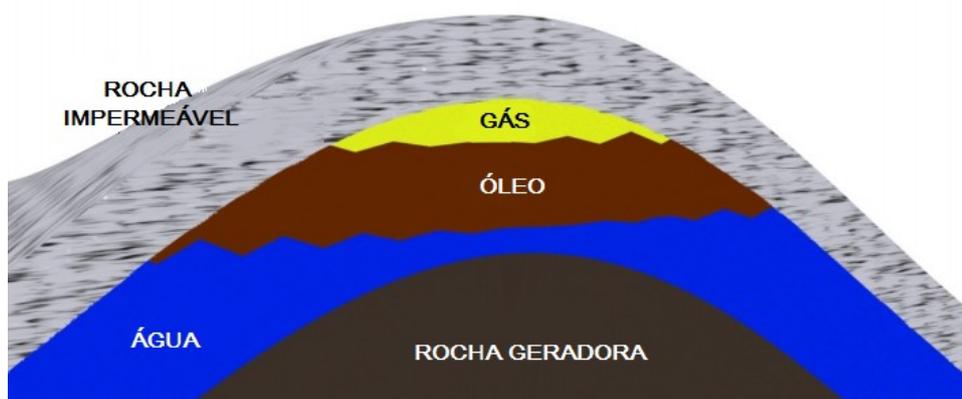
1 INTRODUÇÃO

O registro da participação do petróleo na vida do homem remonta a tempos bíblicos, sendo retirado de exsudações naturais encontradas em todos os continentes. Na antiga Babilônia, por exemplo, os tijolos eram assentados com asfalto e o betume era usado na calafetação de embarcações. Mais tarde, já na era moderna, descobriu-se que a destilação do petróleo resultava em produtos que substituíam, com grande margem de lucro, o querosene obtido do carvão e o óleo de baleia, que eram largamente utilizados para iluminação, marcando o início da era do petróleo. Posteriormente, com a invenção dos motores a gasolina e a diesel, estes derivados até então desprezados adicionaram lucros expressivos à atividade (THOMAS, 2004).

A partir desse período, acontecem os grandes avanços tecnológicos no aprimoramento de dispositivos de aquisição, processamento e interpretação de dados sísmicos, como também nos processos de recuperação de petróleo das jazidas já conhecidas. Ao longo do tempo, o petróleo foi se impondo como fonte de energia e, atualmente, com o advento da petroquímica, além da grande utilização dos seus derivados, centenas de novos compostos são produzidos, alguns deles diariamente consumidos, desde plásticos até produtos farmacêuticos e cosméticos. Com isso, o petróleo, além de produzir combustível, passou a ser peça fundamental às facilidades e comodidades da vida moderna.

A engenharia de reservatórios, por sua vez, se preocupa basicamente com a retirada dos fluidos do interior das rochas, de modo que eles possam ser conduzidos até a superfície. A Figura 1 ilustra um reservatório de petróleo subterrâneo típico. Após a formação do óleo na rocha fonte ou geradora, ele migra para a rocha-reservatório, onde se acumula desde que haja a rocha selante para se interpor em seu caminho.

Figura 1 – Esquema típico de um reservatório de petróleo

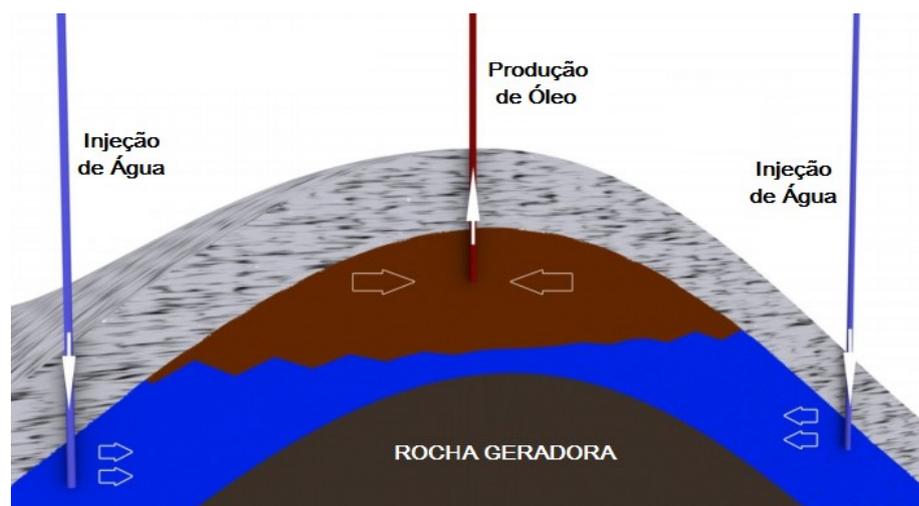


Fonte: ASADOLLAHI, 2012.

Os reservatórios, cujos mecanismos são pouco eficientes e que, por consequência, retêm grandes quantidades de hidrocarbonetos após a exaustão da sua energia natural (recuperação primária), são fortes candidatos ao emprego de uma série de processos que visam à obtenção de uma recuperação adicional (recuperação secundária). Diante disso, métodos de recuperação foram desenvolvidos para se obter uma produção maior do que aquela que se obteria, caso apenas a energia natural do reservatório fosse utilizada. Atualmente, a injeção de água consiste em um dos métodos convencionais de recuperação, cujas tecnologias são bem conhecidas, cujo grau de confiança na aplicação é bastante elevado (THOMAS, 2004). A recuperação via injeção de água busca um comportamento puramente mecânico, sem qualquer interação de natureza química ou termodinâmica entre os fluidos ou entre os fluidos e a rocha.

O fluido injetado, neste caso a água, deve deslocar o óleo para fora dos poros da rocha-reservatório e ao mesmo tempo ir ocupando o espaço deixado à medida que este vai sendo expulso. A Figura 2 ilustra um esquema de injeção de água do tipo periférico. Para o gerenciamento da técnica de injeção de água, faz-se necessária a simulação do fluxo dinâmico dos fluidos na rocha-reservatório a fim de decidir quais as vazões de injeção nos poços injetores serão adotadas para maximizar o retorno e uniformizar o varrido de óleo, impedindo ao máximo a irrupção precoce de água nos poços produtores e reduzindo custos de tratamento e de separação óleo-água.

Figura 2 – Esquema de injeção de base ou periférica



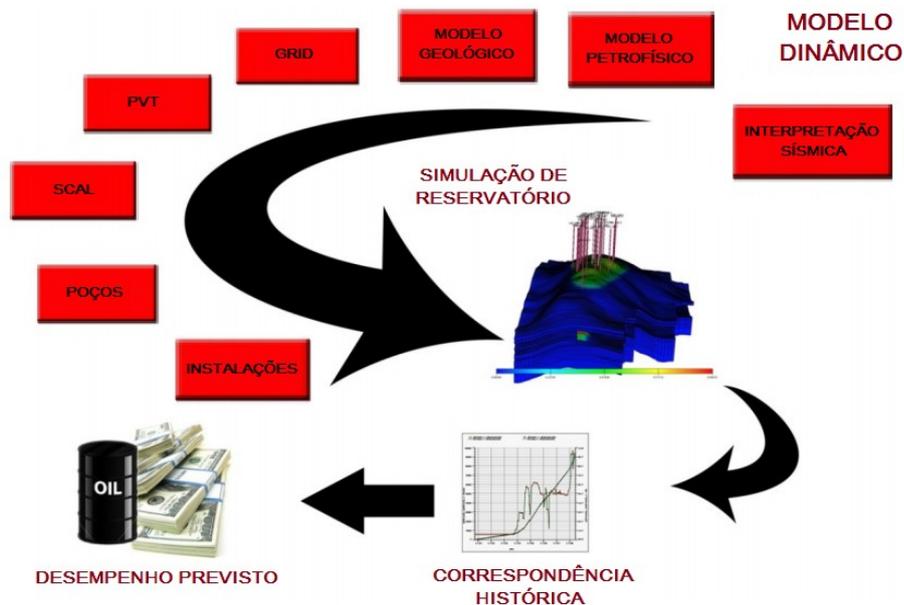
Fonte: ASADOLLAHI, 2012.

Para realizar a simulação do fluxo dos fluidos no reservatório, é necessária a solução de um sistema de equações de fluxo de fluidos em meios porosos que consistem em equações diferenciais parciais não-lineares, sendo utilizados para isto simuladores de fluxo.

Atualmente, existem simuladores eficazes disponíveis no mercado. No entanto, não há como ter acesso ao seu código-fonte, sendo, portanto, impossível o estudo de implementações de novas técnicas de otimização eficazes do gerenciamento que requerem alterações na codificação do simulador.

A ferramenta MRST (*MATLAB Reservoir Simulation Toolbox*), desenvolvido pelo grupo SINTEF, da Noruega, contém um conjunto extenso de rotinas para montagem de simuladores de fluxo, utilizando várias formulações de solução das equações diferenciais. Trata-se de código aberto em linguagem MATLAB, que permite livre acesso ao código e, com isso, o desenvolvimento de novas tecnologias de simulação e otimização.

Figura 3 - Etapas principais do processo de simulação de um reservatório



Fonte: ASADOLLAHI, 2012.

1.1 Justificativa e motivação

Haja vista a importância do petróleo no dia a dia da sociedade, é fundamental, como toda atividade industrial, a viabilidade econômica do projeto, representada por uma boa produtividade, aliada às boas condições de injetividade, resultando em vazões de produção de óleo economicamente atrativas e com o menor volume de fluido injetado possível.

Os projetos de recuperação de óleo, geralmente exclusivos para cada reservatório, devem especificar aspectos como quantidades e distribuição dos poços de injeção e de produção, pressões e vazões de injeção, estimativas das vazões de produção e volumes de

fluidos a serem injetados e produzidos. Tais dados, além de serem necessários para o dimensionamento dos equipamentos, são indispensáveis para a exequibilidade do projeto.

Além do exposto acima, o estudo da simulação de fluxo de fluidos em meios porosos pode ser aplicado em problemas similares na Engenharia Civil, como o estudo de implementação de barragens subterrâneas ou também na análise de infiltração de águas pluviais no solo, tratando-se, nestes casos, de fluxo monofásico (água).

1.2 Objetivos gerais e específicos

O presente trabalho tem como objetivo desenvolver um simulador de fluxo em reservatórios de petróleo, a partir da codificação de um problema de fluxo bifásico óleo/água, em código aberto em linguagem MATLAB. Para isto, serão analisados e apresentados os aspectos básicos da engenharia de petróleo, podendo-se listar como objetivos específicos:

- Descrever a morfologia e o comportamento mecânico da rocha-reservatório;
- Avaliar propriedades intrínsecas a tais rochas, como permeabilidade e porosidade;
- Entender e explicar o uso de equações básicas de fluxo em meio poroso;
- Codificação de um problema de fluxo bifásico óleo/água em formulação IMPES (*Implicit Pressure Explicit Saturation Method*);
- Simulação do problema anterior com utilização da *toolbox* MRST;
- Comparação dos resultados obtidos por ambos os métodos acima.

2 REFERENCIAL TEÓRICO

2.1 Introdução à Engenharia de Petróleo

A engenharia de reservatórios de petróleo se preocupa basicamente com a extração de hidrocarbonetos do interior das rochas, de modo que eles sejam conduzidos até a superfície, de maneira eficiente. São estudadas a caracterização das jazidas, as propriedades das rochas e dos fluidos nelas contidos, a maneira como estes fluidos interagem dentro da rocha e as leis físicas que regem o movimento dos fluidos no seu interior, com o objetivo de maximizar a produção de hidrocarbonetos com o menor custo possível.

É fundamental o conhecimento de propriedades básicas da rocha e dos fluidos nelas contidos, pois são elas que determinam as quantidades dos fluidos existentes no meio poroso, a sua distribuição, a capacidade desses fluidos se moverem e a quantidade que pode ser extraída.

Uma rocha-reservatório, de maneira geral, apresenta como volume poroso total ocupado a soma do volume dos materiais sólidos V_s e do volume dos espaços vazios existentes entre eles (volume poroso V_p). Diante disso, a *porosidade* de uma rocha pode ser definida como:

$$\varphi = \frac{V_p}{V_p + V_s} \quad (01)$$

A porosidade pode ainda ser classificada de acordo com a comunicação entre os poros de uma rocha, sendo *porosidade absoluta* a razão entre o volume de todos os poros, interconectados ou não, e o volume total da rocha, enquanto a *porosidade efetiva* refere-se à razão entre o volume dos poros interconectados e o volume total da rocha.

A *compressibilidade*, por definição, é o quociente entre a variação fracional de volume e a variação de pressão, sendo, para rochas-reservatório, a variação de volume corresponde à variação do volume poroso (*compressibilidade efetiva da formação*).

Em fluxos multifásicos, é necessário estabelecer que percentual do volume poroso é ocupado por cada fluido, isto é, a *saturação* das fases óleo, água e gás, e o percentual do volume poroso V_p ocupado por cada uma daquelas.

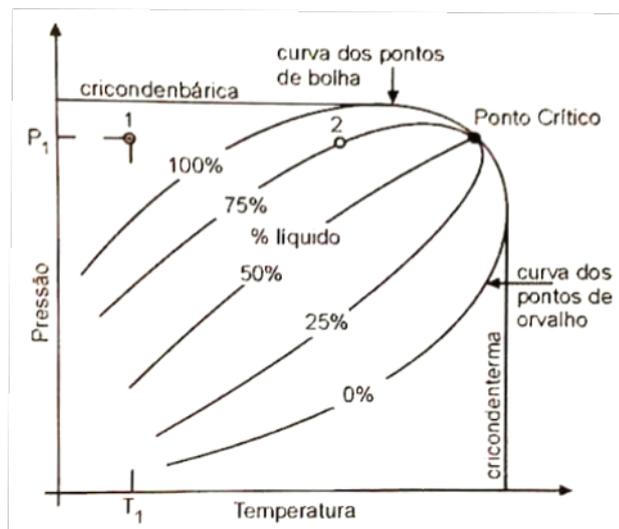
Os fluidos percorrem o que se poderia chamar de “canais porosos”, a medida da capacidade de uma rocha permitir o fluxo de fluidos é chamada permeabilidade. Quando há somente um fluido saturando a rocha, tem-se a *permeabilidade absoluta* do meio. Já a

permeabilidade efetiva aos fluidos depende das saturações de cada um dos fluidos no meio poroso, ou seja, para cada valor de saturação de um fluido corresponde um valor de permeabilidade efetiva daquele fluido.

Outra propriedade de grande importância é a *mobilidade* de um fluido, que corresponde à relação entre a sua permeabilidade efetiva e a sua viscosidade. No caso da razão água-óleo λ_w/λ_o , nota-se que quanto maior for essa razão de mobilidades, menor será a eficiência de deslocamento de óleo, uma vez que, devido a sua maior mobilidade, o fluido injetado tenderá a “furar” o banco de óleo, criando caminhos preferenciais entre os poços injetores e os produtores.

O comportamento de uma determinada mistura de hidrocarbonetos depende das condições de pressão e temperatura a que estiver submetida, podendo ser representado através de um diagrama de fases.

Figura 4 – Diagrama de fases de uma mistura de hidrocarbonetos



Fonte: THOMAS, 2004.

A curva dos *pontos de bolha* (ebulição) e a curva dos *pontos de orvalho* (condensação) dividem o diagrama em uma região de misturas líquidas, uma região de misturas gasosas e uma região em que parte da mistura está no estado líquido e a outra parte no estado gasoso. Na figura, se uma mistura está submetida a uma temperatura T_1 e pressão P_1 (ponto 1), ela se encontra em toda sua totalidade no estado líquido. Já o ponto 2 representa uma mistura na qual 75% da massa se encontra no estado líquido e 25% no estado gasoso.

Devido a essa mudança de comportamento, existem algumas relações dentro da engenharia de petróleo que são utilizadas a fim de avaliar o volume ocupado pelo fluido sujeito a condições de temperatura e pressão quaisquer e o volume ocupado nas condições

padrão de superfície (1 atm e 20°C). Portanto, o *Fator Volume de Formação de gás* (B_g) é definido como a razão entre o volume que ele ocupa numa condição de pressão e temperatura qualquer e o volume que ele ocupa nas condições padrão. Analogamente, tem-se o *Fator Volume de Formação de óleo* (B_o).

Uma mistura líquida em condições de reservatório corresponde a um volume de óleo mais gás dissolvido. O parâmetro que exprime a quantidade de gás presente no líquido é a *razão de solubilidade* de uma mistura de hidrocarbonetos, que, a certa condição de pressão e temperatura, é a relação entre o volume de gás que está dissolvido, expresso em condições de superfície, e o volume de óleo que será obtido na mistura.

Os reservatórios apresentam mecanismos de produção devido a certa quantidade de energia natural ou primária, resultante de todas as situações e circunstâncias geológicas pelas quais a jazida passou até se formar completamente. Contudo, esses mecanismos são pouco eficientes, retendo grandes quantidades de hidrocarbonetos após a exaustão da sua energia natural. Diante disso, são necessários métodos adicionais de recuperação, sendo um dos métodos convencionais mais utilizados, dadas às tecnologias bem conhecidas e grau de confiança na aplicação, o de injeção de água.

Como já descrito anteriormente, os projetos de recuperação de óleo devem especificar aspectos como quantidades e distribuição dos poços de injeção e de produção, pressões e vazões de injeção, estimativas das vazões de produção e volumes de fluidos a serem injetados e produzidos. Esses dados, além de serem necessários para o dimensionamento dos equipamentos, são fundamentais para a viabilidade econômica do projeto.

2.2 Física do fluxo de fluidos em meios porosos

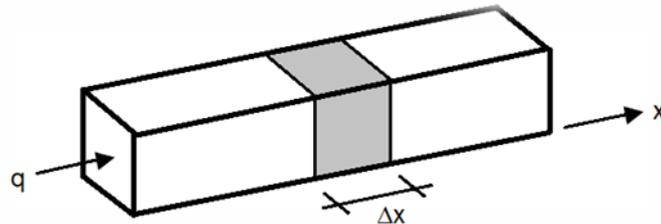
O estudo do fluxo dos fluidos nos meios porosos apresenta como ponto central a equação da difusividade hidráulica ou simplesmente equação da difusão, obtida a partir da associação de três outras equações: a lei de Darcy (transporte de massa), a equação de continuidade (conservação de massa) e uma equação de estado, que para o caso de líquidos, corresponde à equação de compressibilidade.

A lei de Darcy demonstra uma relação direta entre a vazão que atravessa um material poroso e a diferença de carga associada a essa vazão, que pode ser expressa em função da velocidade aparente e gradientes de pressão, para a direção x da seguinte forma:

$$u_x = -\frac{k}{\mu} \frac{\partial P}{\partial x} \quad (02)$$

Considerando um prisma retangular unidimensional composto de material poroso, a conservação de massa pode ser formulada através de um elemento de comprimento Δx , por onde passa um fluido de massa específica ρ e velocidade u .

Figura 5 – Elemento de meio poroso



Fonte: Elaborada pela autor.

A equação da continuidade descreve a taxa de variação de massa dentro do elemento em x de meio poroso devido ao fluxo de entrada e saída do fluido através de suas faces. O balanço de massa pode ser representado pela equação:

$$\{u\rho A\}_x - \{u\rho A\}_{x+\Delta x} = \frac{\partial}{\partial t} \{\phi A \Delta x \rho\} \quad (03)$$

Dividindo ambos os membros por Δx , tomando seu limite tendendo a zero, e admitindo área de seção transversal A constante:

$$-\frac{\partial}{\partial x} (\rho u) = \frac{\partial}{\partial t} \{\phi \rho\} \quad (04)$$

Por ser a porosidade função da pressão, utiliza-se a definição de compressibilidade da rocha, dada por:

$$c_r = -\left(\frac{1}{\phi}\right) \left(\frac{\partial \phi}{\partial P}\right)_T \quad (05)$$

Para uma dada temperatura constante, a Equação (05) pode ser escrita da forma:

$$\frac{\partial \varphi}{\partial P} = \varphi c_r \quad (06)$$

Já a compressibilidade dos fluidos pode ser definida em função do Fator Volume de Formação (B):

$$c_f = B \frac{d(1/B)}{dP} \quad (07)$$

Onde: $B = f(P) = \frac{\text{volume nas condições de reservatório}}{\text{volume nas condições-padrão}}$

Para este trabalho, mais adiante, será visto que foi admitida a incompressibilidade dos fluidos e da rocha, ou seja, que os valores de c_r e c_f são constantes.

2.3 Equações Diferenciais de Difusão

A dedução da equação da difusão é a associação da equação da continuidade com a lei que rege o transporte de fluido no meio poroso, ou seja, a lei de Darcy. Substituindo a Equação (02) na (04), e incluindo os termos de produção/injeção e o Fator Volume de Formação do fluido considerado, obtém-se a equação de derivadas parciais que descreve o fluxo monofásico em um meio poroso unidimensional.

$$\frac{\partial}{\partial x} \left(\frac{k}{\mu B} \frac{\partial P}{\partial x} \right) - q' = \frac{\partial}{\partial t} \left(\frac{\varphi}{B} \right) \quad (08)$$

O lado esquerdo da equação descreve o fluxo do fluido no reservatório e sua produção/injeção, enquanto o lado direito representa as compressibilidades da rocha e do fluido. A fim de tornar a pressão a variável principal do lado direito da equação, por ser mais conveniente escrevê-la em termo dessa grandeza, visto que é ela que se costuma medir nos fluidos nos reservatórios, pode-se aplicar regra da cadeia, no qual resulta em:

$$\frac{\partial}{\partial t} \left(\frac{\varphi}{B} \right) = \frac{1}{B} \frac{\partial \varphi}{\partial t} + \varphi \frac{\partial(1/B)}{\partial t} \quad (09)$$

Por ser a porosidade e o Fator Volume de Formação funções da pressão, pode-se aplicar derivação implícita, e, posteriormente, as definições de compressibilidade da rocha e do fluido vistas anteriormente, no que resulta nas Equações (10) e (11), respectivamente:

$$\frac{\partial}{\partial t} \left(\frac{\varphi}{B} \right) = \frac{1}{B} \frac{d\varphi}{dP} \frac{\partial P}{\partial t} + \varphi \frac{d(1/B)}{dP} \frac{\partial P}{\partial t} \quad (10)$$

$$\frac{\partial}{\partial t} \left(\frac{\varphi}{B} \right) = \frac{\varphi c_r}{B} \frac{\partial P}{\partial t} + \frac{\varphi c_f}{B} \frac{\partial P}{\partial t} \quad (11)$$

Isolando os termos em comum, e substituindo a Equação (11) na (08), tem-se:

$$\frac{\partial}{\partial x} \left(\frac{k}{\mu B} \frac{\partial P}{\partial x} \right) - q' = \frac{\varphi}{B} [c_r + c_f] \frac{\partial P}{\partial t} \quad (12)$$

$$\frac{\partial}{\partial x} \left(\frac{k}{\mu B} \frac{\partial P}{\partial x} \right) - q' = \frac{\varphi c_T}{B} \frac{\partial P}{\partial t} \quad (13)$$

Vale lembrar que, para as equações diferenciais deduzidas acima, é admitido que somente um fluido satura o meio poroso, considerando-se ainda a compressibilidade e a viscosidade dos fluidos constante e o meio poroso homogêneo e isotrópico. Logo, para fluxos multifásicos, deve-se levar em consideração a saturação dos fluidos em questão, bem como o Fator Volume de Formação.

Considerando um fluxo bifásico de água e óleo, substituindo a equação de Darcy, aplicando o respectivo Fator Volume de Formação e incluindo os termos de produção/injeção, têm-se as equações de fluxo para as duas fases:

$$\frac{\partial}{\partial x} \left(\frac{k k_{r1}}{\mu_1 B_1} \frac{\partial P_1}{\partial x} \right) - q'_1 = \frac{\partial}{\partial t} \left(\frac{\varphi S_1}{B_1} \right) \quad (14)$$

Onde:

$$l = o \text{ (óleo) ou } w \text{ (água)}$$

$$P_w = P_o - P_{cow}$$

$$S_o + S_w = 1$$

3 MATERIAIS E MÉTODOS

3.1 Método das Diferenças Finitas

A simulação numérica é uma ferramenta de grande utilidade para o engenheiro de reservatórios, pois os diversos problemas complexos referentes a essa área de engenharia, quando formulado matematicamente, conduzem a equações ou sistemas de equações diferenciais parciais não-lineares, que não se prestam aos métodos analíticos de solução.

É neste caso que o Método das Diferenças Finitas se aplica, possibilitando obter soluções aproximadas para essas equações. Denomina-se *discretização* de uma função o processo no qual tanto o domínio quanto a imagem da função passam de contínuos para discretos, realizada através da expansão de uma série de Taylor, dado um certo número de subintervalos de largura h (Por simplificação, irá ser demonstrado todo o método somente para uma direção).

Por ser a pressão em um reservatório uma função do tempo e do espaço $p(x,t)$, o processo de aproximação neste caso deve ser feito através das derivadas parciais, cujo procedimento é análogo às derivadas totais, bastando trabalhar separadamente com cada uma das variáveis.

Diante disso, procede-se à discretização em cada uma das variáveis (espaço e tempo), usando os métodos de aproximação por diferenças finitas, a fim de implementar uma formulação explícita ou implícita da equação de difusão.

Quanto às condições de contorno, há basicamente dois tipos: condição de Dirichlet, onde é especificado o valor da função (pressão), e a de Neumann, onde é especificado o valor da derivada (fluxo).

3.1.1 Aproximação da primeira derivada

Para o cálculo da primeira derivada através de diferenças finitas, são consideradas as formas progressiva, regressiva e centrada.

Na forma progressiva, tem-se que a expansão da função por uma série de Taylor é dada pela Equação (15).

$$f(x+h) = f(x) + \frac{h}{1!}f'(x) + \frac{h^2}{2!}f''(x) + \frac{h^3}{3!}f'''(x) + \dots \quad (15)$$

Desprezando-se os termos após a primeira derivada, isto é, “truncando-se” a equação, tem-se que a formulação da aproximação da derivada primeira por diferença finita na forma progressiva resulta na Equação (16).

$$f'(x) = \frac{f(x+h) - f(x)}{h} + O(h) \quad (16)$$

Onde $O(h)$ é definido como sendo o erro de truncamento, isto é, a diferença entre o valor exato da derivada primeira e a sua aproximação. Como se observa, $O(h)$ será da ordem de grandeza do intervalo h , e, portanto, neste caso que o erro de truncamento é de primeira ordem.

De maneira análoga, para a forma regressiva:

$$f'(x) = \frac{f(x) - f(x-h)}{h} + O(h) \quad (17)$$

Se a aproximação for feita a partir dos pontos $x-h$ e $x+h$, e posteriormente uma equação for subtraída da outra, obtém-se a forma centrada da aproximação da derivada primeira de uma função, cujo erro de truncamento agora é de segunda ordem, ou seja, de maior precisão que as duas outras formas anteriores.

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + O(h^2) \quad (18)$$

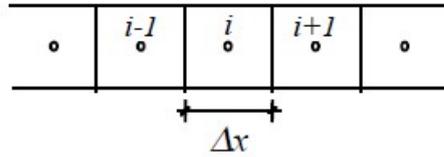
3.1.2 Aproximação da segunda derivada

Somando-se as expansões progressiva e regressiva de $f(x)$ e isolando o termo da segunda derivada, a Equação (19) fornece uma aproximação centrada para a segunda derivada em termos de diferenças finitas, com um erro de truncamento de segunda ordem.

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + O(h^2) \quad (19)$$

3.1.3 Discretização do Espaço

Figura 6 – Sistema de “grid-blocks” igualmente espaçados



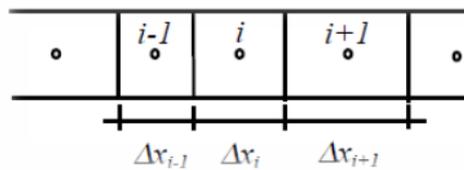
Fonte: KLEPPE, 2017.

Para subintervalos (“grid-blocks”) de largura constante Δx , a segunda derivada, por exemplo, da pressão, em relação à x pode ser expressa pela aproximação com erro de segunda ordem dada pela Equação (20).

$$\frac{\partial^2 P(x_i, t)}{\partial x^2} = \frac{P(x_{i+1}, t) - 2P(x_i, t) + P(x_{i-1}, t))}{(\Delta x)^2} + \mathcal{O}(\Delta x^2) \quad (20)$$

Um sistema unidimensional de *grid-blocks* desigualmente espaçados encontra-se representado abaixo, usado mais frequentemente nas simulações por ser mais realista, visto que é possível melhor refinamento na geometria e melhor acurária nas regiões de rápidas mudanças de pressão e saturação, assim como nas proximidades dos poços de produção/injeção (KLEPPE, 2017).

Figura 7 – Sistema de “grid-blocks” desigualmente espaçados



Fonte: KLEPPE, 2017.

Utilizando de expansões da série de Taylor, obtém-se a forma aproximada do termo de fluxo monofásico representado pela forma da Equação (21).

$$\frac{\partial}{\partial x} \left[f(x) \frac{\partial p}{\partial x} \right]_i = \frac{2f(x)_{i+1/2} \frac{(P_{i+1} - P_i)}{(\Delta x_{i+1} - \Delta x_i)} - 2f(x)_{i-1/2} \frac{(P_i - P_{i-1})}{(\Delta x_i - \Delta x_{i-1})}}{\Delta x_i} + \mathcal{O}(\Delta x) \quad (21)$$

Para $f(x) = \frac{k}{\mu B}$, tem-se que:

$$\frac{\partial}{\partial x} \left(\frac{k}{\mu B} \frac{\partial p}{\partial x} \right)_i = \frac{2 \left(\frac{k}{\mu B} \right)_{i+1/2} \frac{(P_{i+1} - P_i)}{(\Delta x_{i+1} - \Delta x_i)} - 2 \left(\frac{k}{\mu B} \right)_{i-1/2} \frac{(P_i - P_{i-1})}{(\Delta x_i - \Delta x_{i-1})}}{\Delta x_i} + \mathcal{O}(\Delta x) \quad (22)$$

Pode-se definir o termo que multiplica a diferença de pressões como *Transmissibilidade*, e, portanto, o termo do lado esquerdo da equação de fluxo aproximada

por diferenças finitas resulta na Equação (23). Fisicamente, a Transmissibilidade compreende a razão entre o gradiente de vazão do fluido com a variação de pressão entre células subsequentes (CARDOSO, 2009).

$$\frac{\partial}{\partial x} \left(\frac{k}{\mu B} \frac{\partial P}{\partial x} \right)_i \approx T_{x_{i+1/2}} (P_{i+1} - P_i) + T_{x_{i-1/2}} (P_i - P_{i-1}) \quad (23)$$

3.1.4 Discretização do Tempo

Na discretização do tempo, pode-se usar a aproximação centrada, cujo erro de truncamento é de segunda ordem, expressa pela Equação (24).

$$\frac{\partial P(x_i, t)}{\partial t} = \frac{P(x_i, t + \Delta t) - P(x_i, t - \Delta t)}{2\Delta t} + O(\Delta t^2) \quad (24)$$

3.1.4.1 Métodos Explícitos

São métodos que usam uma fórmula explícita para calcular o valor da variável dependente no próximo valor da variável independente. Em uma fórmula explícita, o lado direito da equação tem apenas grandezas conhecidas, ou seja, o valor subsequente de uma variável dependente y_{n+1} é calculado a partir de uma expressão na forma:

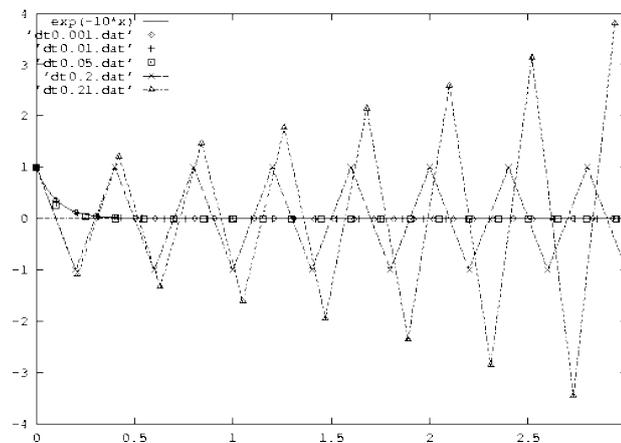
$$y_{n+1} = y_n + hf(y_n, t_n) \quad (25)$$

Usando como exemplo a equação de fluxo monofásico e de *grid-blocks* de comprimentos iguais, a forma por diferenças finitas pode ser descrita pela Equação (26).

$$\frac{P_{i+1}^t - 2P_i^t + P_{i-1}^t}{(\Delta x)^2} = \left(\frac{\phi \mu c}{k} \right) \frac{P_i^{t+\Delta t} - P_i^t}{\Delta t} \quad (26)$$

Métodos explícitos são fáceis de serem implementados, contudo, há limitações quanto ao tamanho do passo a fim de garantir estabilidade numérica, isto é, a partir de um certo valor do passo h , os valores obtidos para y_{n+1} podem oscilar consideravelmente, gerando uma expressiva instabilidade numérica. Logo, são raramente utilizados (ZELTKEVIC, 1998).

Figura 8 – Exemplo de instabilidade numérica com a formulação explícita



Fonte: ZELTKEVIC, 1998.

3.1.4.2 Métodos Implícitos

Nos métodos implícitos, a equação usada para calcular o y_{n+1} tem a forma:

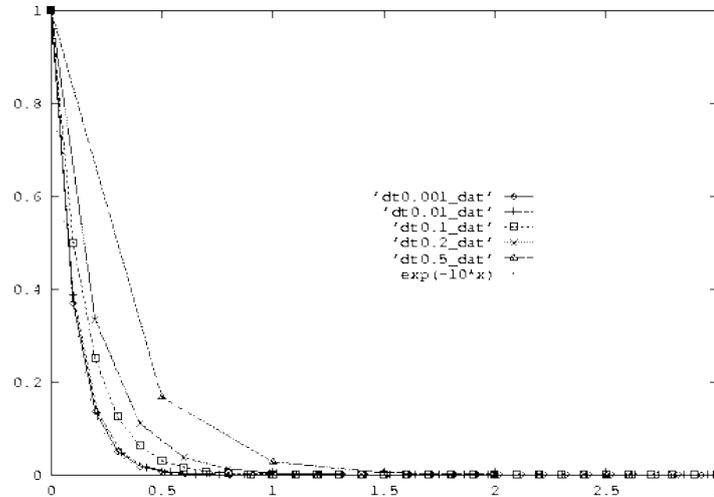
$$y_{n+1} = y_n + hf(y_{n+1}, t_{n+1}) \quad (27)$$

Neste caso, a incógnita y_{n+1} aparece em ambos os lados da equação. Dessa forma, têm-se um sistema de N equações e N incógnitas, que devem ser resolvidas simultaneamente, por eliminação de Gauss, por exemplo. Para o mesmo exemplo do item 3.1.4.1, tem-se que:

$$\frac{P_{i+1}^{t+\Delta t} - 2P_i^{t+\Delta t} + P_{i-1}^{t+\Delta t}}{(\Delta x)^2} = \left(\frac{\varphi \mu c}{k}\right) \frac{P_i^{t+\Delta t} - P_i^t}{\Delta t} \quad (28)$$

Métodos implícitos são mais precisos que os métodos explícitos, porém requerem maior esforço computacional em cada passo. Além disso, métodos implícitos podem ser usados para substituir os explícitos em situações em que os requisitos de estabilidade destes últimos impõem condições rigorosas quanto ao tamanho do passo h (ZELTKEVIC, 1998).

Figura 9 – Exemplo de estabilidade numérica com a formulação implícita



Fonte: ZELTKEVIC, 1998.

3.1.5 Condições de Contorno

3.1.5.1 Condição de Contorno de Dirichlet:

Nesta condição, é especificado o valor da pressão, normalmente aplicada nas faces de extremidade do sistema em questão. Para um sistema linear simples, por exemplo, pode-se especificar tais pressões da seguinte forma:

$$P(x = 0, t > 0) = P_{i=1/2}^{t>0} = P_L \quad (29)$$

$$P(x = L, t > 0) = P_{i=N+1/2}^{t>0} = P_R \quad (30)$$

3.1.5.2 Condição de Contorno de Neumann:

Alternativamente, é especificada a taxa de fluxo, seja para dentro ou para fora da face de extremidade do sistema em questão. Como a taxa de fluxo pode ser expressa pela lei de Darcy, logo, para o mesmo sistema do item anterior, pode-se expressar as taxas de fluxos nas faces de extremidade esquerda e direita respectivamente por:

$$Q_L = -\frac{kA}{\mu} \left(\frac{\partial P}{\partial x} \right)_{x=0} \quad (31)$$

$$Q_R = -\frac{kA}{\mu} \left(\frac{\partial P}{\partial x} \right)_{x=L} \quad (32)$$

3.2 Linguagem MATLAB

O MATLAB é um programa de computador de uso específico, otimizado para executar cálculos científicos e de engenharia. A linguagem MATLAB oferece uma ampla biblioteca de funções predefinidas para que a programação técnica se torne mais fácil e eficiente do que em outras linguagens, como FORTRAN ou C.

Dentre as vantagens de se utilizar o MATLAB como ferramenta de programação técnica pode-se listar as seguintes (CHAPMAN, 2003):

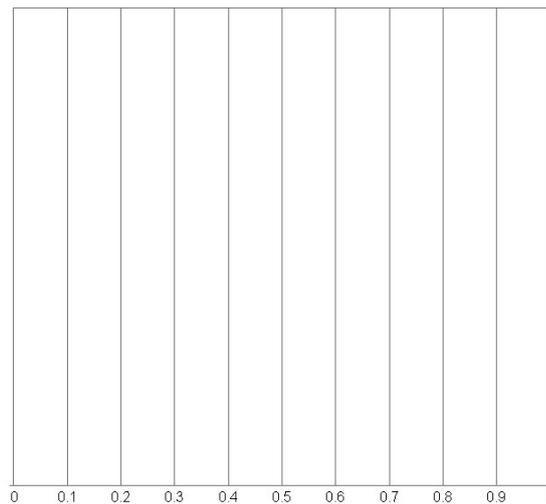
- Facilidade de uso: por ser uma linguagem interpretada;
- Independência de plataforma: com suporte para muitos sistemas computacionais diferentes;
- Funções predefinidas: que proporcionam soluções testadas e empacotadas para muitas tarefas técnicas básicas. Somam-se a essas funções as caixas de ferramenta-padrão (“*toolboxes*”) para resolução de problemas complexos em áreas específicas, como é o caso da *toolbox* para simulação de reservatórios de petróleo MRST;
- Diagramas independentes de dispositivos: com comandos próprios para imagens e desenhos integrais.

4 RESULTADOS

4.1 Codificação de um problema de fluxo bifásico óleo-água com formulação IMPES

Para o desenvolvimento do simulador para fluxo bifásico óleo-água, tomou-se como exemplo o Exercício 2 das notas de aula do professor Jon Kleppe, que, originalmente, utilizou da linguagem FORTRAN para sua implementação, adaptando-o para a linguagem MATLAB, mantendo-se o método IMPES (*Implicit Pressure Explicit Saturation*) como o método utilizado para a solução do problema, no qual é utilizada a formulação implícita para a obtenção das pressões de óleo e a explícita para a obtenção das saturações de água ao longo do tempo.

Figura 10 – Esquema do *grid* composto de 10 células



Fonte: Elaborada pelo autor (MATLAB).

Para isso, foi feita a discretização de um bloco retangular constituído de material poroso de comprimento igual a 1m em 10 (dez) células de comprimento $\Delta x = 10\text{cm}$, constituindo assim um conjunto de *grid-blocks*, cujas propriedades, como porosidade e permeabilidade, são os parâmetros de entrada, inseridos por planilhas de Excel importadas ao MATLAB. Os parâmetros de saída, isto é, as pressões e saturações são obtidas a partir da iteração das equações de fluxo vistas anteriormente para um determinado intervalo de tempo Δt , que, atingido um tempo máximo de 400s (critério de parada adotado), são posteriormente exportadas para outras planilhas em Excel. A partir desses últimos dados, utilizou-se a ferramenta de plotagem de células do MRST para melhor visualização e comparação dos resultados.

Tabela 1 – Dados do Sistema, Condições Iniciais e de Contorno

A (cm²)	Φ	k (darcy)	S_{wi}	C_r	T_{máx} (s)	P_{init} (atm)	P_R (atm)	Q_{wi} (cm³/s)
1000	0,25	0,1	0,2	0,000045	400	200	200	-25

Fonte: KLEPPE, 2017.

Onde:

A é área da seção transversal do bloco retangular (cm²);

Φ é a porosidade do meio;

k é a permeabilidade do meio (darcy);

S_{wi} é a saturação inicial de água aplicada às 10 células;

C_r é a compressibilidade da rocha, que, pela ordem de grandeza, adotou-se como a rocha sendo incompressível;

T_{máx} é o critério de parada aplicado às iterações ao longo do tempo (s);

P_{init} é a condição inicial das pressões de óleo aplicada às células do *grid* (atm);

P_R é a condição de contorno de Dirichlet aplicada à face da extremidade direita do *grid* (atm);

Q_{wi} é a condição de contorno de Neumann aplicada à face da extremidade esquerda do *grid* (cm³/s), cujo valor é negativo para indicar que se trata de injeção de água.

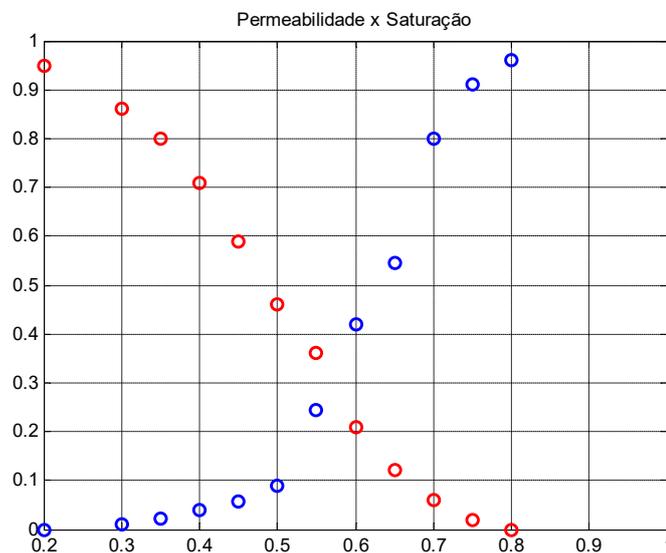
Neste exemplo, parâmetros que são funções da saturação, como permeabilidades relativas e pressão capilar, e os que são funções da pressão, como viscosidade e os fatores volume de formação da água e do óleo, são obtidos a partir da interpolação dos pontos das suas respectivas curvas mostradas na Figura 11, cujos valores destes parâmetros encontram-se nas Tabelas 2 e 3. Vale ressaltar que a porosidade, embora também seja função da pressão, é admitida constante, visto que se esteja adotando fluidos e rochas incompressíveis neste trabalho.

Tabela 2 – Pontos das curvas de permeabilidade relativa e pressão capilar, funções da saturação

S_w	K_{ro}	K_{rw}	P_{cow} (atm)
0,2	0,95	0	1
0,3	0,86	0,01	0,52
0,35	0,8	0,021	0,41
0,4	0,71	0,039	0,34
0,45	0,59	0,058	0,28
0,5	0,46	0,09	0,23
0,55	0,36	0,245	0,18
0,6	0,21	0,42	0,14
0,65	0,12	0,545	0,1
0,7	0,06	0,8	0,06
0,75	0,02	0,91	0
0,8	0	0,96	0

Fonte: KLEPPE, 2017.

Figura 11 - Pontos das curvas de permeabilidade relativa, função da saturação



Fonte: Elaborada pelo autor (MATLAB).

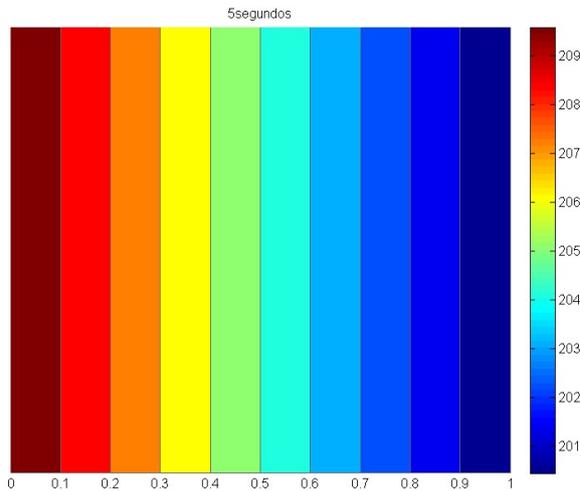
Tabela 3 – Pontos das curvas de fator volume de formação e viscosidade, funções da pressão.

P (atm)	B_o	B_w	μ_{o0}	μ_{w0}
100	1,522	1,0044	0,45	0,95
200	1,5	1	0,5	1
300	1,478	0,9956	0,55	1,05

Fonte: KLEPPE, 2017.

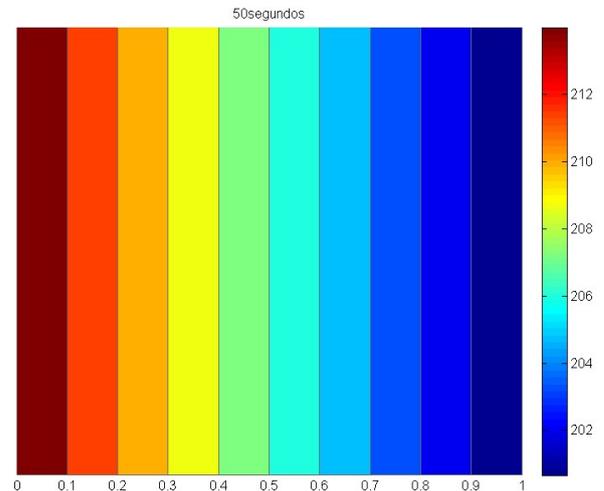
Diante dos dados inseridos, obteve-se para as pressões de óleo P_o (atm), em diferentes instantes de tempo, os seguintes resultados. Pode-se observar nas Figuras 12 a 17 que a pressão, na célula 1, varia de 200 (condição inicial) a 224 atm.

Figura 12 – Pressão P_o em $t = 5s$ (KLEPPE).



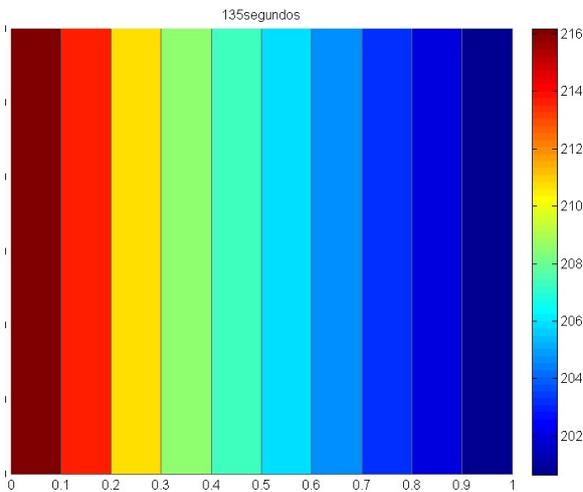
Fonte: Elaborada pelo autor (MATLAB).

Figura 13 - Pressão P_o em $t = 50s$ (KLEPPE).



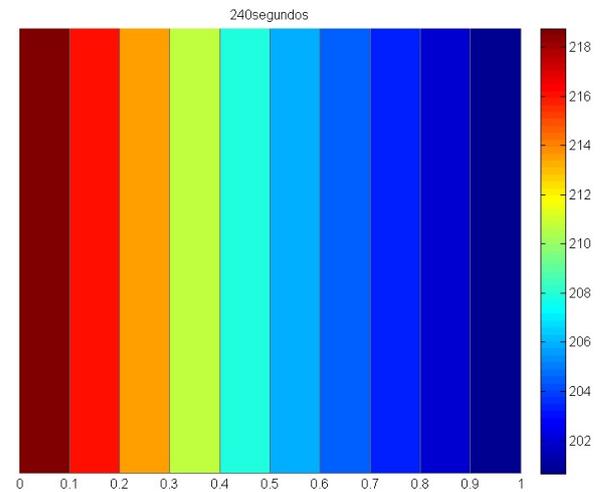
Fonte: Elaborada pelo autor (MATLAB).

Figura 14 - Pressão P_o em $t = 135s$ (KLEPPE)

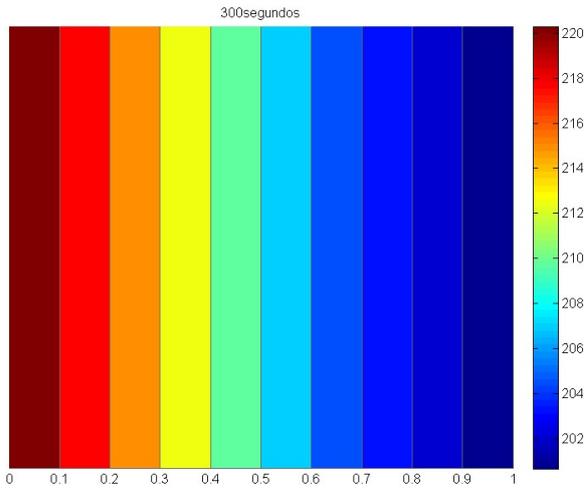


Fonte: Elaborada pelo autor (MATLAB).

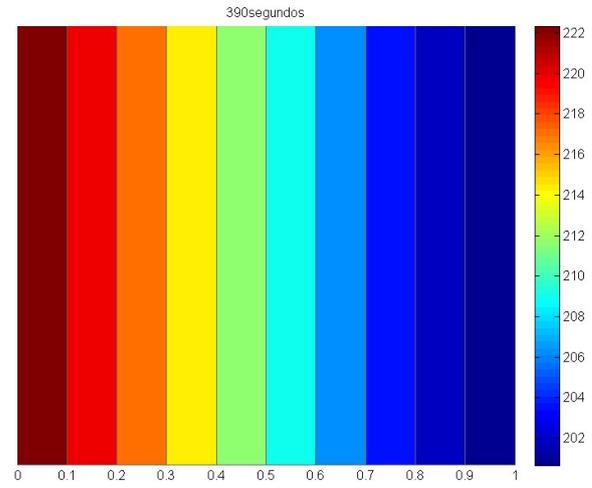
Figura 15 - Pressão P_o em $t = 240s$ (KLEPPE)



Fonte: Elaborada pelo autor (MATLAB).

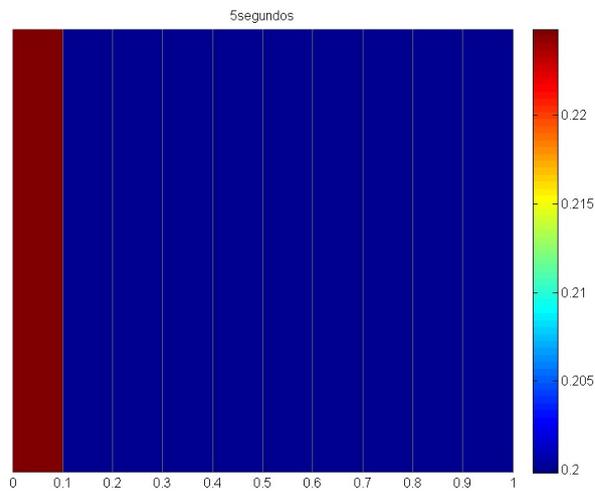
Figura 16 - Pressão P_o em $t = 300s$ (KLEPPE)

Fonte: Elaborada pelo autor (MATLAB).

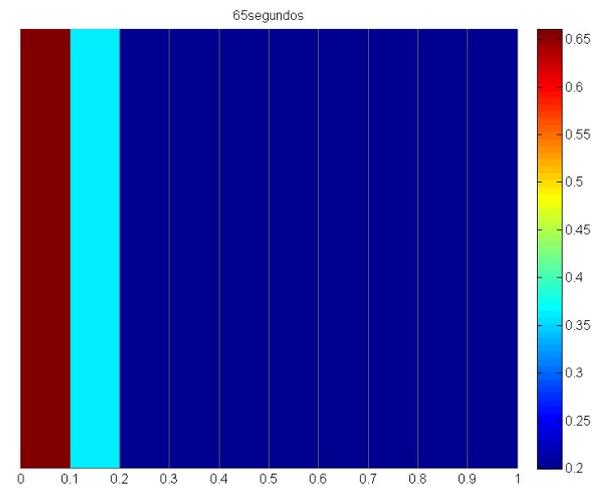
Figura 17 - Pressão P_o em $t = 390s$ (KLEPPE)

Fonte: Elaborada pelo autor (MATLAB).

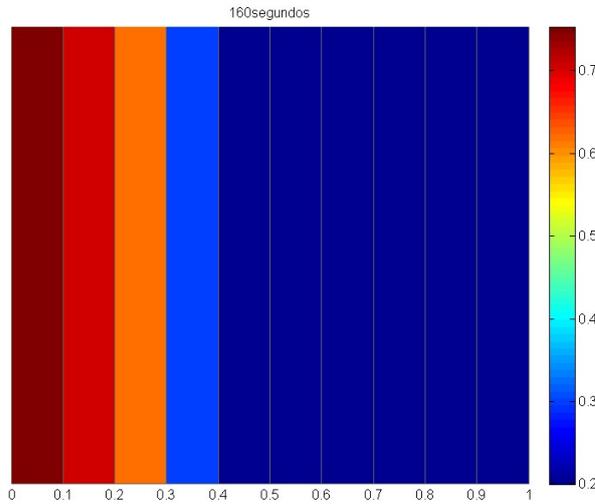
Nas Figuras 18 a 23, pode-se visualizar o crescimento da saturação de água (da esquerda para a direita) com o tempo, conforme se dá a sua injeção na extremidade esquerda.

Figura 18 – Saturação S_w em $t = 5s$ (KLEPPE)

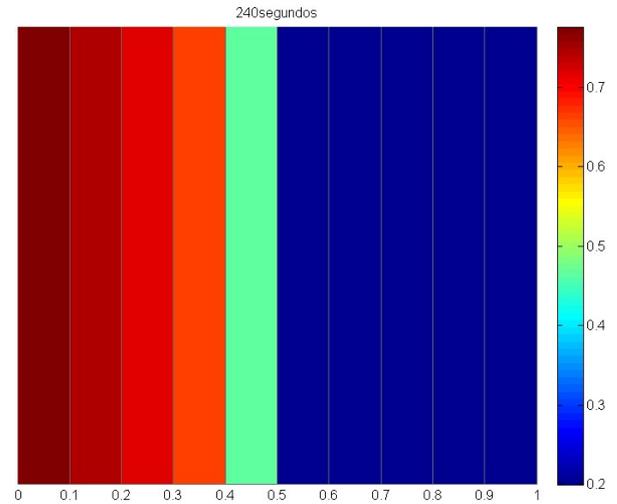
Fonte: Elaborada pelo autor (MATLAB).

Figura 19 – Saturação S_w em $t = 65s$ (KLEPPE)

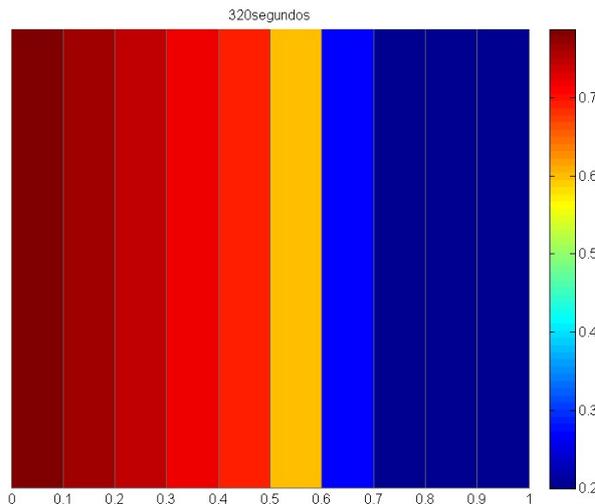
Fonte: Elaborada pelo autor (MATLAB).

Figura 20 – Saturação S_w em $t = 160s$ (KLEPPE)

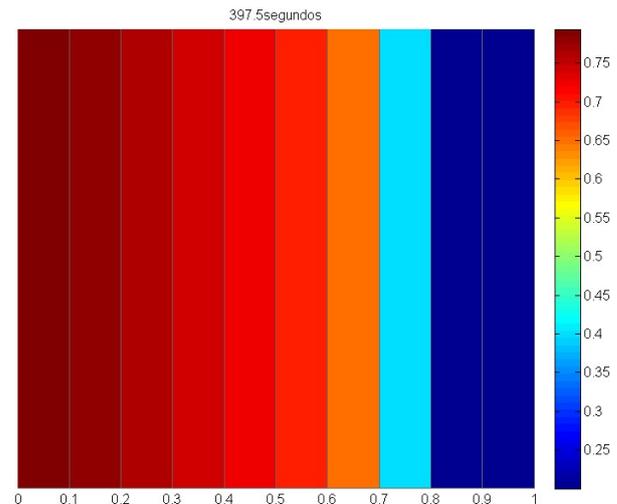
Fonte: Elaborada pelo autor (MATLAB).

Figura 21 - Saturação S_w em $t = 240s$ (KLEPPE)

Fonte: Elaborada pelo autor (MATLAB).

Figura 22 - Saturação S_w em $t = 320s$ (KLEPPE)

Fonte: Elaborada pelo autor (MATLAB).

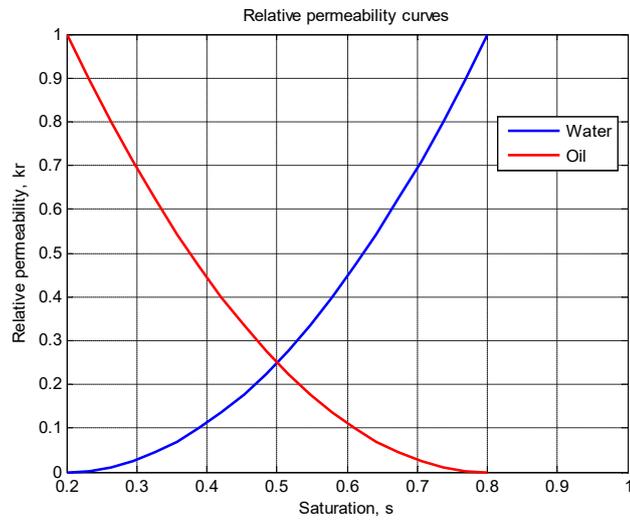
Figura 23 - Saturação S_w em $t = 397s$ (KLEPPE).

Fonte: Elaborada pelo autor (MATLAB).

4.2 Simulação do problema anterior com utilização da *toolbox* MRST

Após a instalação da *toolbox* do MRST no MATLAB, utilizou-se o conjunto de seus módulos e funções intrínsecas para a formulação do exercício exposto no item anterior, mantendo-se os parâmetros de entrada, isto é, adotando-se ainda a incompressibilidade dos fluidos e da rocha, as condições iniciais e de contorno. Contudo, nesta simulação, a interpolação dos pontos das Tabelas 2 e 3 foi substituída pelas suas respectivas curvas devidamente ajustadas (ver figura 24), sendo necessário, nestes casos, especificar os expoentes e coeficientes de tais funções, por ter sido adotado aproximação do tipo Corey.

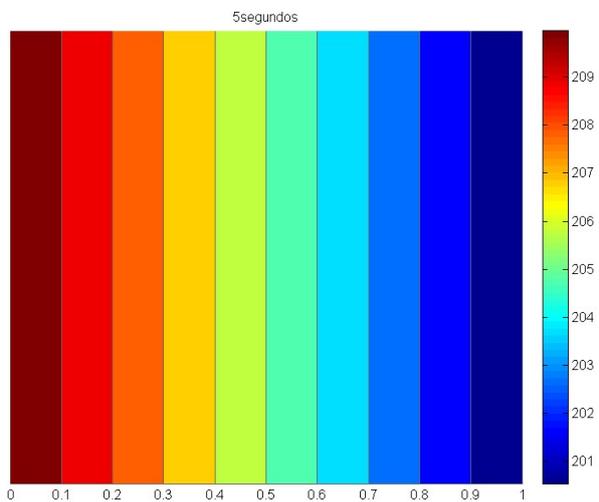
Figura 24 – Curvas de permeabilidade relativa ajustadas aos pontos da Tabela 2



Fonte: Elaborada pelo autor (MATLAB).

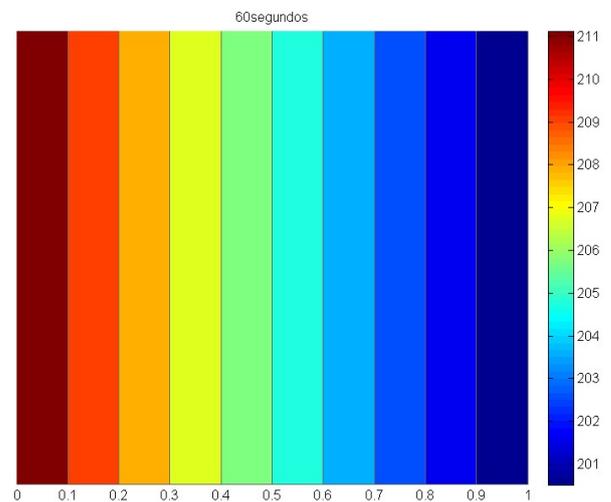
Seguem abaixo nas Figuras 25 a 30, os resultados obtidos para a pressão de óleo P_o , em atm, e nas Figuras 31 a 36, as saturações de água S_w , com a simulação através da utilização da *toolbox* MRST.

Figura 25 – Pressão P_o em $t = 5$ s (MRST)



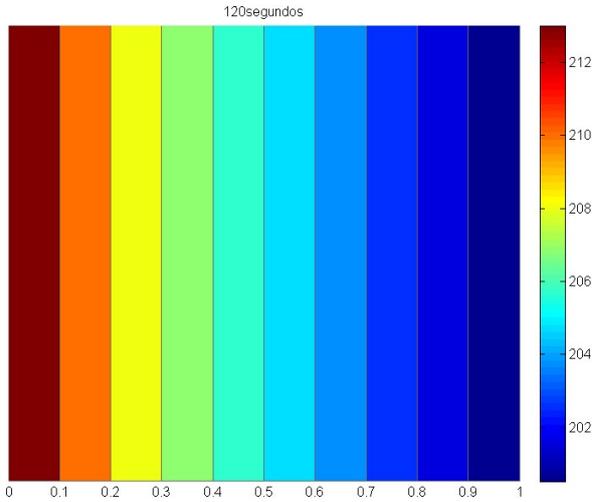
Fonte: Elaborada pelo autor (MATLAB).

Figura 26 - Pressão P_o em $t = 60$ s (MRST)



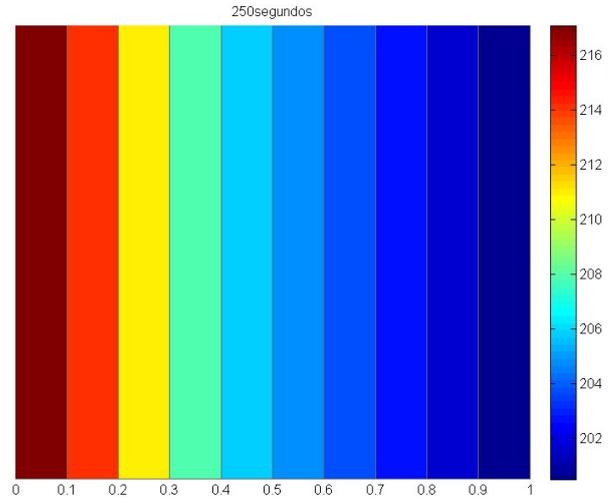
Fonte: Elaborada pelo autor (MATLAB).

Figura 27 - Pressão P_o em $t = 120s$ (MRST)



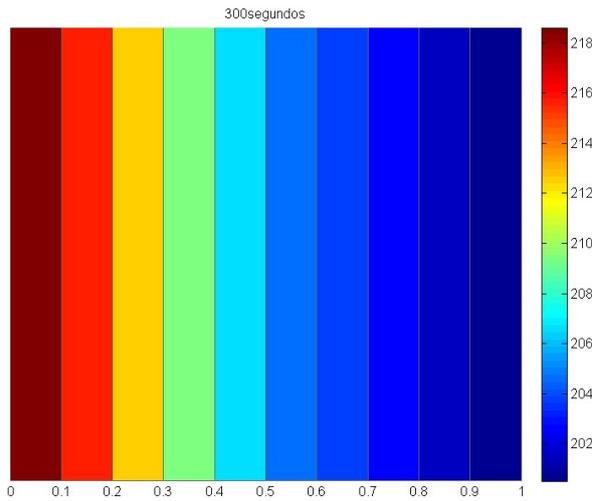
Fonte: Elaborada pelo autor (MATLAB).

Figura 28 – Pressão P_o em $t = 250s$ (MRST)



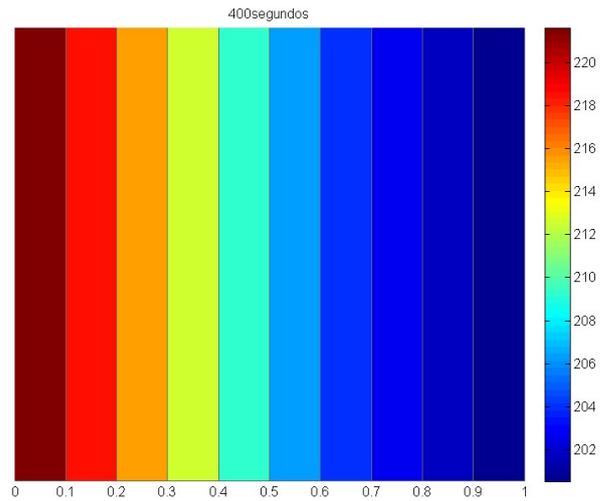
Fonte: Elaborada pelo autor (MATLAB).

Figura 29 - Pressão P_o em $t = 300s$ (MRST)



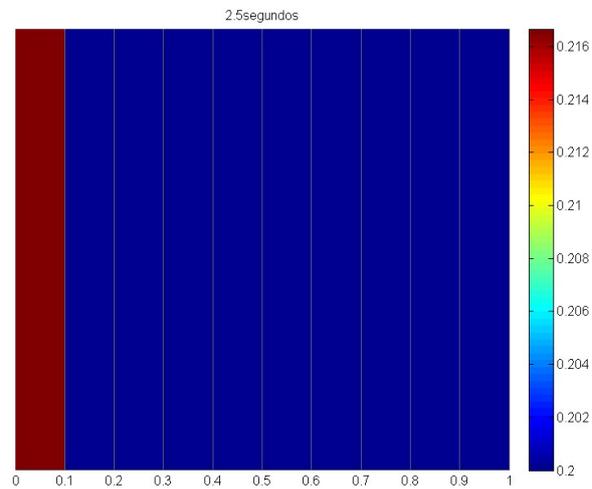
Fonte: Elaborada pelo autor (MATLAB).

Figura 30 - Pressão P_o em $t = 400s$ (MRST)



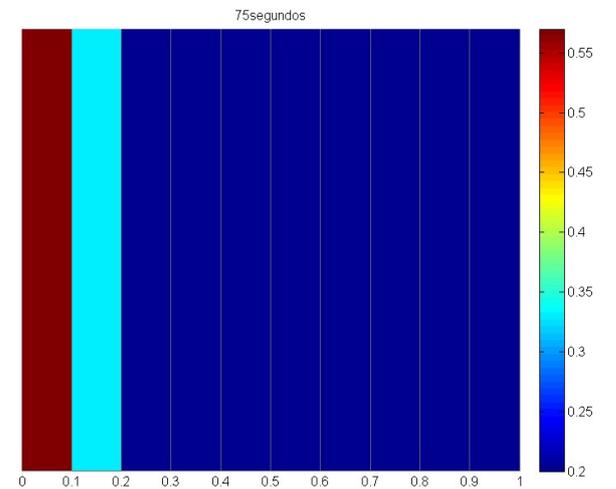
Fonte: Elaborada pelo autor (MATLAB).

Figura 31 - Saturação S_w em $t = 2,5s$ (MRST)



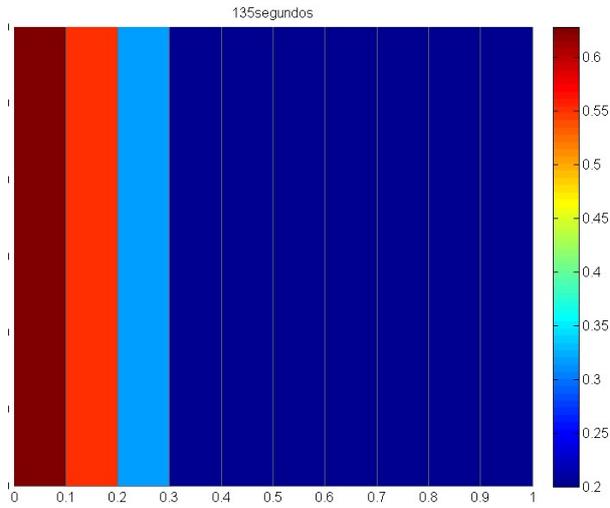
Fonte: Elaborada pelo autor (MATLAB).

Figura 32 - Saturação S_w em $t = 75s$ (MRST)



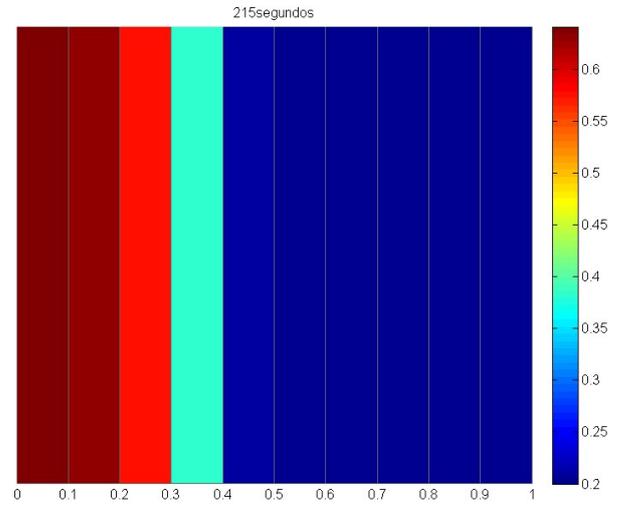
Fonte: Elaborada pelo autor (MATLAB).

Figura 33- Saturação S_w em $t = 135s$ (MRST)



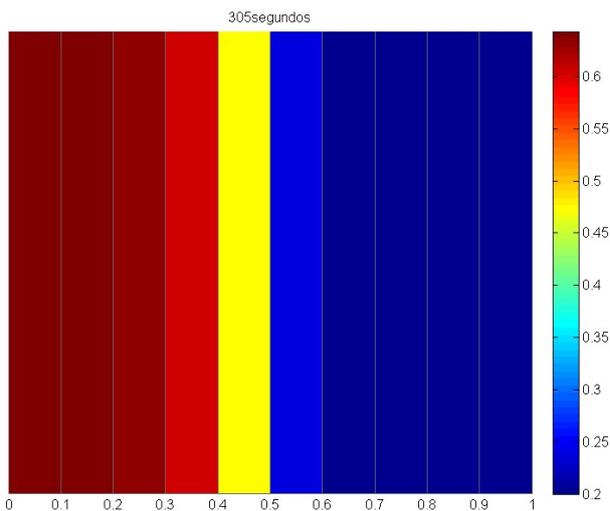
Fonte: Elaborada pelo autor (MATLAB).

Figura 34 - Saturação S_w em $t = 215s$ (MRST)



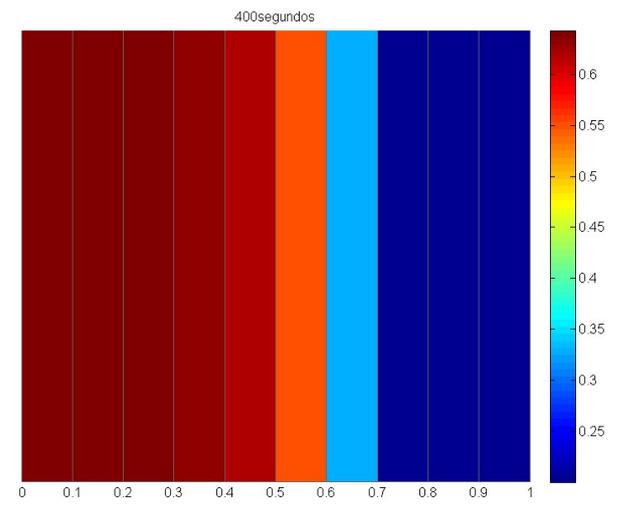
Fonte: Elaborada pelo autor (MATLAB).

Figura 35 - Saturação S_w em $t = 305s$ (MRST)



Fonte: Elaborada pelo autor (MATLAB).

Figura 36 - Saturação S_w em $t = 400s$ (MRST)



Fonte: Elaborada pelo autor (MATLAB).

4.3 Comparação dos resultados obtidos pelos métodos anteriores

A seguir, tem-se a comparação dos resultados obtidos pelo simulador do prof. Jon Kepple e pelo simulador desenvolvimento com o uso do MRST, sendo considerados, para tal finalidade, dois intervalos de tempo quaisquer: 250s e 400s. Como se pode observar, as pressões de óleo P_o obtidas pelos dois métodos convergem de modo mais eficiente que as saturações da água S_w , embora apresentem a mesma tendência.

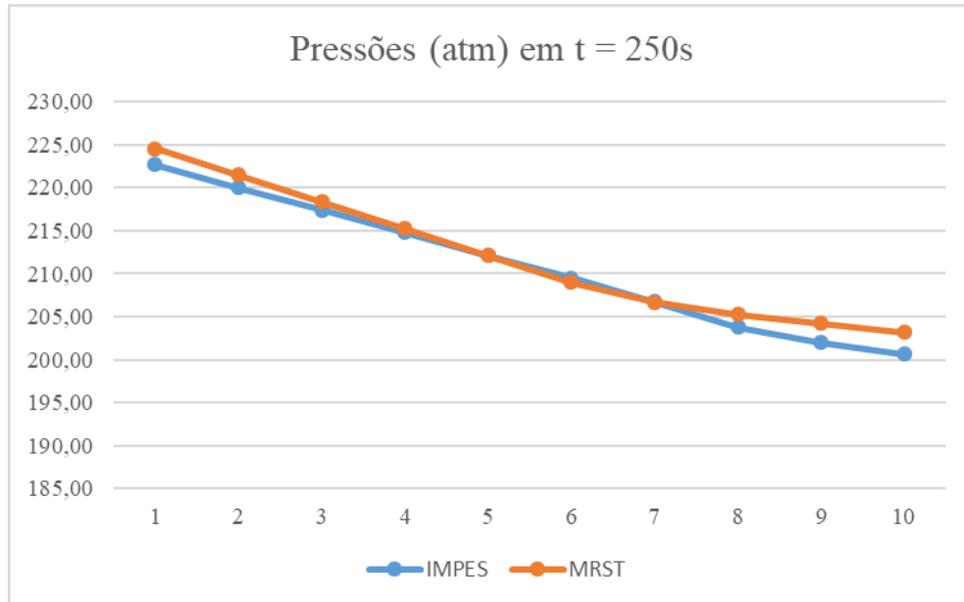
Tais diferenças nos resultados das saturações podem ser justificadas pelas aproximações das curvas de permeabilidade introduzidas no desenvolvimento do simulador no MRST, a partir dos dados fornecidos na codificação original em MATLAB.

Tabela 4 – Pressões P_o e Saturações S_w obtidas por ambas as simulações em $t = 250s$

t = 250s			
P_o (atm)		S_w	
KEPPLE	MRST	KEPPLE	MRST
219,15	219,96	0,7784	0,6423
216,51	216,85	0,7488	0,6361
213,89	213,71	0,7201	0,6112
211,23	210,69	0,6799	0,5099
208,39	208,59	0,5447	0,2666
205,97	207,40	0,2245	0,2005
204,62	206,34	0,2000	0,2000
203,30	205,29	0,1999	0,2000
201,98	204,23	0,2000	0,2000
200,66	203,18	0,2000	0,2000

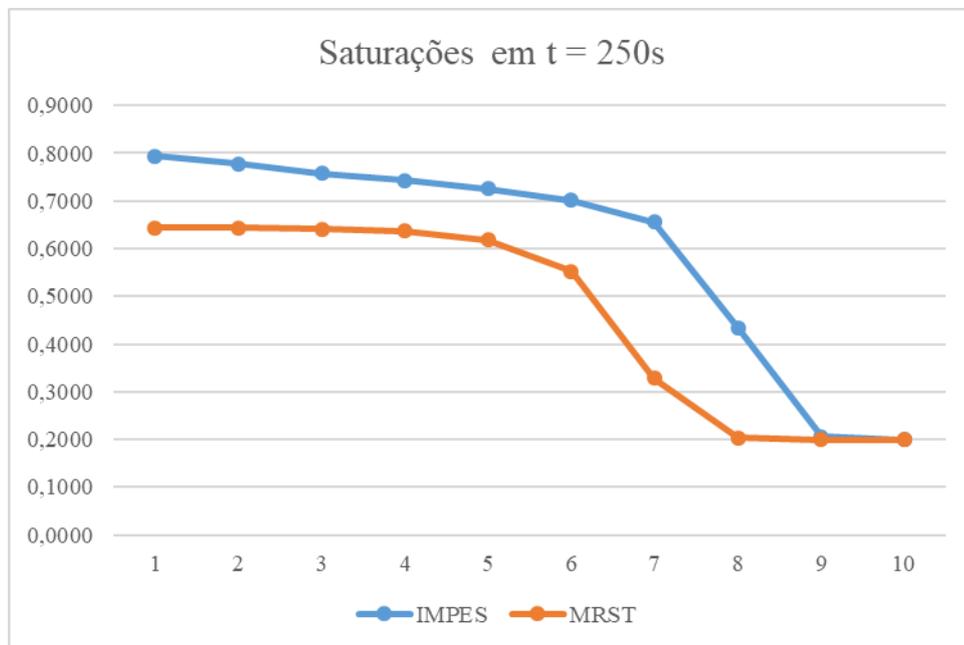
Fonte: Elaborada pelo autor.

Figura 37 – Pressões P_o obtidas por ambas as simulações no instante $t = 250s$



Fonte: Elaborada pelo autor.

Figura 38 – Saturações S_w obtidas por ambas as simulações no instante $t = 250s$

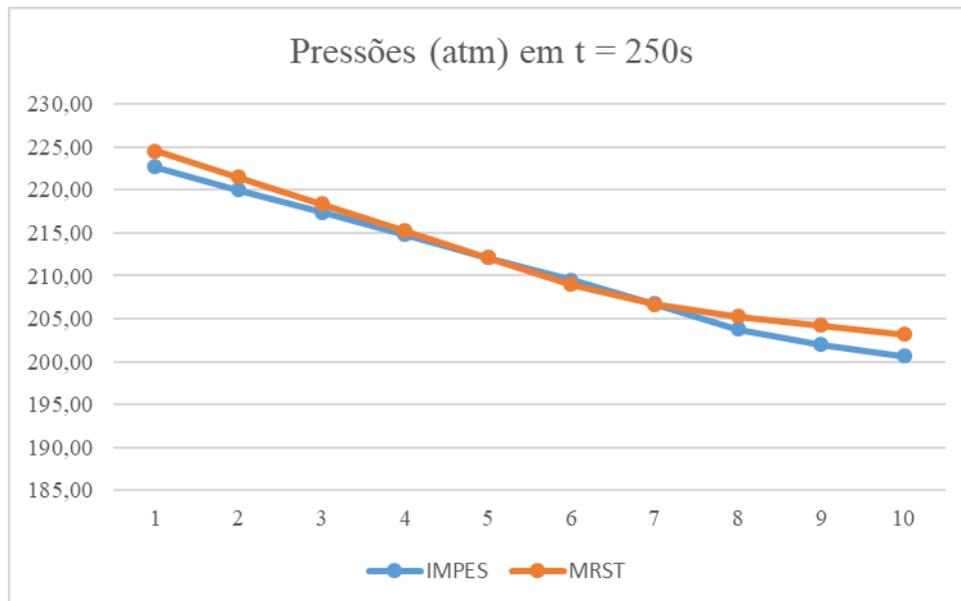


Fonte: Elaborada pelo autor.

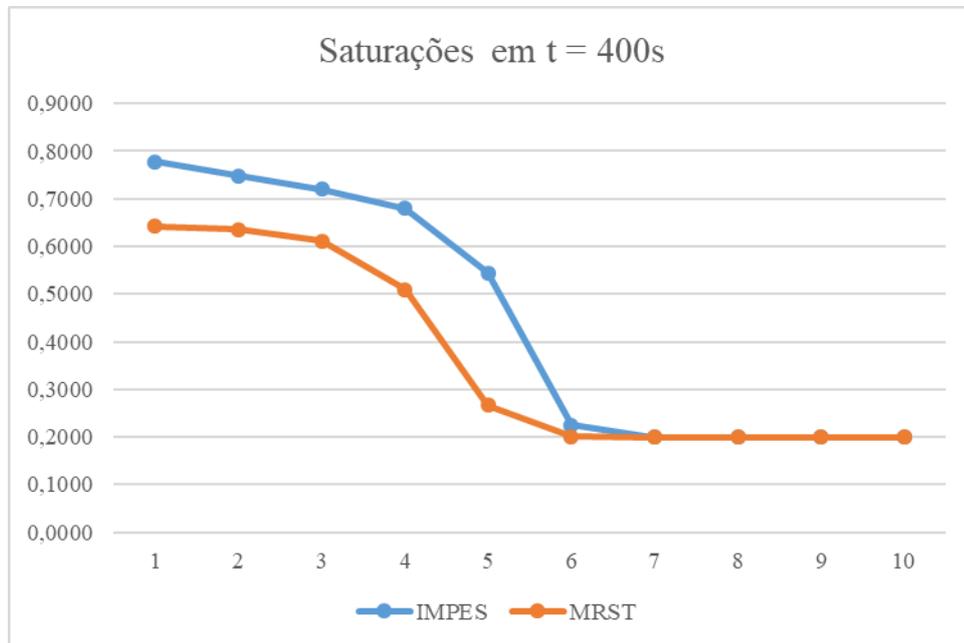
Tabela 5 – Pressões P_o e Saturações S_w obtidas por ambas as simulações em $t = 400s$

t = 400s			
P_o (atm)		S_w	
KEPPLE	MRST	KEPPLE	MRST
222,68	224,56	0,7939	0,6433
220,04	221,47	0,7778	0,6430
217,40	218,37	0,7582	0,6416
214,77	215,26	0,7434	0,6363
212,13	212,13	0,7250	0,6187
209,47	209,02	0,7008	0,5521
206,78	206,65	0,6558	0,3286
203,76	205,29	0,4339	0,2031
201,98	204,23	0,2055	0,2000
200,66	203,18	0,2000	0,2000

Fonte: Elaborada pelo autor.

Figura 39 – Pressões P_o obtidas por ambas as simulações no instante $t = 250s$ 

Fonte: Elaborada pelo autor.

Figura 40 – Saturações S_w obtidas por ambas as simulações no instante $t = 400s$ 

Fonte: Elaborada pelo autor.

Vale salientar ainda que, para os códigos elaborados, e apresentados nos Apêndices I e II, o MRST apresentou maior eficiência quanto ao tempo computacional para a obtenção de resultados frente à codificação do item 4.1.

5 CONCLUSÕES

Como visto anteriormente, a injeção de água consiste num dos métodos mais tradicionais de recuperação de óleo, de comportamento puramente mecânico, sem qualquer interação de natureza química ou termodinâmica entre os fluidos ou entre os fluidos e a rocha, e para o seu gerenciamento, faz-se necessária a simulação do fluxo dinâmico dos fluidos na rocha-reservatório a fim de se obter as vazões de injeção de água nos poços injetores e otimizar a recuperação de óleo, impedindo ao máximo a irrupção precoce de água nos poços produtores.

Para isso, são utilizados atualmente simuladores de fluxo que, por sua vez, utilizam de um sistema de equações diferenciais parciais não-lineares para a obtenção de tais informações, tendo em vista sempre uma boa produtividade, aliada às boas condições de injetividade, resultando em vazões de produção de óleo economicamente atrativas e com o menor volume de fluido injetado possível.

Foi feita, neste trabalho, a simulação de um mesmo problema de fluxo unidirecional bifásico óleo/água através de um bloco retangular de material poroso, cuja codificação utilizou-se de duas ferramentas distintas: a primeira, em código aberto em linguagem MATLAB, com o uso do método por Diferenças Finitas aliado ao Método IMPES (*Implicit Pressure Explicit Saturation Method*), e pela formulação com o uso das rotinas que compõem a *toolbox* MRST (*MATLAB Reservoir Simulation Toolbox*). Em ambos, foram descritos a morfologia e o comportamento mecânico da rocha-reservatório, além de propriedades intrínsecas a tais rochas, como permeabilidade e porosidade.

Diante do exposto e, comparando-se os resultados apresentados no item anterior, pode-se concluir que, visto a comparação dos resultados obtidos para as pressões e saturações pelo Método IMPES e pelo MRST, ambas as formas são válidas e satisfatórias para a análise do processo de injeção de água para a produção de óleo em questão, embora aproximações realizadas ao longo da codificação possam vir a interferir na convergência de alguns desses resultados.

REFERÊNCIAS

- THOMAS, J. E. **Fundamentos de Engenharia de Petróleo**. Editora Interciência, 2004.
- ROSA, A. J.; CARVALHO, R. S.; XAVIER, J.A.D. **Engenharia de Reservatórios de Petróleo**. Editora Interciência, 2006.
- ERTEKIN, T.; ABOU-KASSEM, J.H.; KING, G. R. **Basic Applied Reservoir Simulation**. SPE, 2001.
- Zeltkevic, M. **Numerical Solution of Ordinary Differential Equation**. MIT, 1998.
- ZIENKIEWICZ, O.C.; MORGAN, K. **Finite Elements and Approximation**. Wiley – Interscience, 1993.
- KLEPPE, Jon. **TPG4160 Reservoir Simulation**. NTNU, 2017.
- Disponível em: < <http://www.ipt.ntnu.no/~kleppe/TPG4160/> > Acesso em 12 de dezembro de 2017.
- HOFFMAN, J. D. **Numerical Methods for Engineers and Scientists**. Marcel Dekker Inc., 2001.
- ASADOLLAHI, M. **Waterflooding Optimization for Improved Reservoir Management**. NTNU, 2012.
- MRST - MATLAB Reservoir Simulation Toolbox, SINTEF.
- Disponível em: < <http://www.sintef.no/projectweb/mrst/> > Acesso em 12 de dezembro de 2017.
- CARDOSO, M.A. **Development and Application of Reduced-Order Modeling Procedures for Reservoir Simulation**. Stanford University, 2009.
- CHAPMAN, S.J. **MATLAB para Engenheiros**. Editora Tomson, 2003.

APÊNDICE I – Codificação de fluxo bifásico óleo-água
(Exercício 2 das notas de aula do Professor Jon Kleppe, NTNU - Adaptado)

```

%-----Simulação de Fluxo Bifásico Óleo/Água-----%
%-----Em um sistema horizontal unidimensional-----%
%---Usando Método IMPES e Eliminação de Gauss---%
%-----%
%---Número máximo de grid-blocks definido: 100---%
%-----%

%LEITURA DE DADOS
%-----%

%%DADOS DO ARQUIVO "syst.xls"

%USO e USW = 'Upstream Weighting Factors' para óleo e água
%1.0 para Upstream
%0.0 para Downstream
%Valores entre 0 e 1 (Padrão = 1.0)

syst = importdata('syst.xls');

USO = syst.data(1,1);
USW = syst.data(1,2);
area = syst.data(1,3);
N = syst.data(1,4);
dx = syst.data(:,5);
phi = syst.data(:,6);
perm = syst.data(:,7);
Swi = syst.data(1,8);
Cr = syst.data(1,9);
dt = syst.data(1,10);
tmax = syst.data(1,11);
Pinit = syst.data(1,12);
Pl = zeros(1000,1); Pr = zeros(1000,1);
Pr(1) = syst.data(1,13);
Pl(1) = syst.data(1,14);
Qwi = zeros(1000,1);
Qwi(1) = syst.data(1,15);

%Gerando as posições x(i)
x = zeros(N,1);
x(1) = dx(1)/2;
for i = 2:N
    x(i) = x(i-1) + (dx(i-1)+dx(i))/2;
end

%Condição de Contorno no lado esquerdo a ser usada
%IBC = 2, para Taxa de Injeção Constante
%IBC = 1, para Pressão Constante no lado esquerdo

if Pl ~= 0
    Qwi = 0;
    IBC = 1;
else
    IBC = 2;
end

%%DADOS DO ARQUIVO "sat.xls"
%Tabelas de Permeabilidade Relativa e Pressão Capilar

```

```

sat = importdata('sat.xls');

Nsat = sat.data(1,1);
Pcmult = sat.data(1,2);
Swt = sat.data(:,3);
Krot = sat.data(:,4);
Krw = sat.data(:,5);
Pct = sat.data(:,6);

for i = 1:Nsat
    Pct(i) = Pct(i)*Pcmult;
end

%%DADOS DO ARQUIVO "pvt.xls"

pvt = importdata('pvt.xls');

Npvt = pvt.data(1,1);
Pt = pvt.data(:,2);
Bot = pvt.data(:,3);
Bwt = pvt.data(:,4);
Muot = pvt.data(:,5);
Muwt = pvt.data(:,6);

%Convertendo Bo e Bw para 1/Bo e 1/Bw
for i = 1:Npvt
    Bot(i) = 1/Bot(i);
    Bwt(i) = 1/Bwt(i);
end

%INICIALIZAÇÃO
%-----%

t=zeros(1000);
Po = zeros(N,1); Pw = zeros(N,1); Sw = zeros(N,1);
matrizPO = zeros(N,1000);
matrizSW = zeros(N,1000);

for i = 1:N
    Po(i) = Pinit;
    Sw(i) = Swi;
end

if IBC == 2
    Pl(1) = Po(1);
end

Qop = zeros(1000,1);
Qwp = zeros(1000,1);
Wc = zeros(1000,1);

%Inicializando vetores
Kro = zeros(N,1); Krw = zeros(N,1);
Pcow = zeros(N,1); DPcow = zeros(N,1);
Bo = zeros(N,1); DBo = zeros(N,1);
Bw = zeros(N,1); DBw = zeros(N,1);
Muo = zeros(N,1); Muw = zeros(N,1);
Lamo = zeros(N,1); Lamw = zeros(N,1);

```

```

Txom = zeros(N,1); Txwm = zeros(N,1);
Txop = zeros(N,1); Txwp = zeros(N,1);
Cpoo = zeros(N,1); Csw = zeros(N,1);
Cpow = zeros(N,1); Csww = zeros(N,1);

A = zeros(N,1); B = zeros(N,1); C = zeros(N,1); D = zeros(N,1);

%SAÍDA DE DADOS
%-----%

phead = {'TIME', 'PL', 'P( 1)', 'P( 2)', 'P( 3)', ...
'P( 4)', 'P( 5)', 'P( 6)', 'P( 7)', 'P( 8)', ...
'P( 9)', 'P(10)', 'P(11)', 'P(12)', 'P(13)', ...
'P(14)', 'P(15)', 'P(16)', 'P(17)', 'P(18)', ...
'P(19)', 'P(20)', 'P(21)', 'P(22)', 'P(23)', ...
'P(24)', 'P(25)', 'P(26)', 'P(27)', 'P(28)', ...
'P(29)', 'P(30)', 'P(31)', 'P(32)', 'P(33)', ...
'P(34)', 'P(35)', 'P(36)', 'P(37)', 'P(38)', ...
'P(39)', 'P(40)', 'P(41)', 'P(42)', 'P(44)', ...
'P(44)', 'P(45)', 'P(46)', 'P(47)', 'P(48)', ...
'P(49)', 'P(50)', 'P(51)', 'P(52)', 'P(55)', ...
'P(54)', 'P(55)', 'P(56)', 'P(57)', 'P(58)', ...
'P(59)', 'P(60)', 'P(61)', 'P(62)', 'P(66)', ...
'P(64)', 'P(65)', 'P(66)', 'P(67)', 'P(68)', ...
'P(69)', 'P(70)', 'P(71)', 'P(72)', 'P(77)', ...
'P(74)', 'P(75)', 'P(76)', 'P(77)', 'P(78)', ...
'P(79)', 'P(80)', 'P(81)', 'P(82)', 'P(83)', ...
'P(84)', 'P(85)', 'P(86)', 'P(87)', 'P(88)', ...
'P(89)', 'P(90)', 'P(91)', 'P(92)', 'P(99)', ...
'P(94)', 'P(95)', 'P(96)', 'P(97)', 'P(98)', ...
'P(99)', 'P(100)', 'PR'};

shead = {'TIME', 'SW( 1)', 'SW( 2)', 'SW( 3)', ...
'SW( 4)', 'SW( 5)', 'SW( 6)', 'SW( 7)', 'SW( 8)', ...
'SW( 9)', 'SW(10)', 'SW(11)', 'SW(12)', 'SW(13)', ...
'SW(14)', 'SW(15)', 'SW(16)', 'SW(17)', 'SW(18)', ...
'SW(19)', 'SW(20)', 'SW(21)', 'SW(22)', 'SW(23)', ...
'SW(24)', 'SW(25)', 'SW(26)', 'SW(27)', 'SW(28)', ...
'SW(29)', 'SW(30)', 'SW(31)', 'SW(32)', 'SW(33)', ...
'SW(34)', 'SW(35)', 'SW(36)', 'SW(37)', 'SW(38)', ...
'SW(39)', 'SW(40)', 'SW(41)', 'SW(42)', 'SW(44)', ...
'SW(44)', 'SW(45)', 'SW(46)', 'SW(47)', 'SW(48)', ...
'SW(49)', 'SW(50)', 'SW(51)', 'SW(52)', 'SW(55)', ...
'SW(54)', 'SW(55)', 'SW(56)', 'SW(57)', 'SW(58)', ...
'SW(59)', 'SW(60)', 'SW(61)', 'SW(62)', 'SW(66)', ...
'SW(64)', 'SW(65)', 'SW(66)', 'SW(67)', 'SW(68)', ...
'SW(69)', 'SW(70)', 'SW(71)', 'SW(72)', 'SW(77)', ...
'SW(74)', 'SW(75)', 'SW(76)', 'SW(77)', 'SW(78)', ...
'SW(79)', 'SW(80)', 'SW(81)', 'SW(82)', 'SW(83)', ...
'SW(84)', 'SW(85)', 'SW(86)', 'SW(87)', 'SW(88)', ...
'SW(89)', 'SW(90)', 'SW(91)', 'SW(92)', 'SW(99)', ...
'SW(94)', 'SW(95)', 'SW(96)', 'SW(97)', 'SW(98)', ...
'SW(99)', 'SW(100)'};

whead = {'TIME', 'Pl', 'Qwi', 'Pr', 'Qop', 'Qwp', 'Wc'};

phead(N+3) = {'Pr'};

x = x'; %Transpondo x para formato da tabela

```

```

titulopo = {'TIME, RATES AND OIL PRESSURES'};
xlswrite('po.xls', titulopo, 1, 'A1');
xlswrite('po.xls', phead(1:N+3), 1, 'A2');
xlswrite('po.xls', 'X= ', 1, 'A3');
xlswrite('po.xls', x(1:N), 1, 'B3');

titulosw = {'TIME AND WATER SATURATIONS'};
xlswrite('sw.xls', titulosw, 1, 'A1');
xlswrite('sw.xls', shead(1:N+1), 1, 'A2');
xlswrite('sw.xls', 'X= ', 1, 'A3');
xlswrite('sw.xls', x(1:N), 1, 'B3');

titulowells = {'PRODUCTION/INJECTION RESULTS'};
xlswrite('wells.xls', titulowells, 1, 'A1');
xlswrite('wells.xls', whead, 1, 'A2');

for j = 1:1000 %Time Loop

    if t(j) > tmax %critério de parada
        index = j-1;
        break
    end

    %Interpolação nas tabelas dos dados de entrada
    %para as propriedades funções da saturação (Kro, Krw, Pcow)
    %e para as propriedades funções da pressão (Bo, Bw, Muo, Muw)

    for i = 1:N
        %Permeabilidade Relativa do Óleo
        Kro(i) = INTERP(Sw(i), 0, Nsat, Swt, Krot);
        %Permeabilidade Relativa da Água
        Krw(i) = INTERP(Sw(i), 0, Nsat, Swt, Krwt);
        %Pressão Capilar e sua Derivada
        [Pcow(i), DPcow(i)] = INTERP(Sw(i), 1, Nsat, Swt, Pct);
        %(1/Bo) e sua Derivada
        [Bo(i), DBo(i)] = INTERP(Po(i), 1, Npvt, Pt, Bot);
        %(1/Bw) e sua Derivada
        Pw(i) = Po(i) - Pcow(i);
        [Bw(i), DBw(i)] = INTERP(Pw(i), 1, Npvt, Pt, Bwt);
        %Viscosidade do Óleo
        Muo(i) = INTERP(Po(i), 0, Npvt, Pt, Muot);
        %Viscosidade da Água
        Muw(i) = INTERP(Pw(i), 0, Npvt, Pt, Muwt);

        %Mobilidades

        Lamo(i) = Kro(i) * Bo(i) / Muo(i);
        Lamw(i) = Krw(i) * Bw(i) / Muw(i);

    end

    %Loop para Grid-Blocks

    for i = 1:N
        if i~=1
            Lamom = Lamo(i-1) * USO + Lamo(i) * (1-USO);
            Lamwm = Lamw(i-1) * USW + Lamw(i) * (1-USW);
            if Po(i-1) < Po(i)
                Lamom = Lamo(i) * USO + Lamo(i-1) * (1-USO);
            end
        end
    end

```

```

end
if Pw(i-1) < Pw(i)
    Lamwm = Lamw(i)*USW + Lamw(i-1)*(1-USW);
end
else
    Lamom = Lamo(i);
    Lamwm = Lamw(i);
end
if i ~= N
    Lamop = Lamo(i)*USO + Lamo(i+1)*(1-USO);
    Lamwp = Lamw(i)*USW + Lamw(i+1)*(1-USW);
    if Po(i+1) > Po(i)
        Lamop = Lamo(i+1)*USO + Lamo(i)*(1-USO);
    end
    if Pw(i+1) > Pw(i)
        Lamwp = Lamw(i+1)*USW + Lamw(i)*(1-USW);
    end
else
    Lamop = Lamo(i);
    Lamwp = Lamw(i);
end

%Transmissibilidades

if i ~= 1
    Txom(i) = 2*Lamom/(dx(i)/perm(i)+dx(i-1)/perm(i-1))/dx(i);
    Txwm(i) = 2*Lamwm/(dx(i)/perm(i)+dx(i-1)/perm(i-1))/dx(i);
else
    Txom(i) = 2*Lamom/dx(i)*perm(i)/dx(i);
    Txwm(i) = 2*Lamwm/dx(i)*perm(i)/dx(i);

%Injeção de água no lado esquerdo requer somatório das
transmissibilidades
Txwm(i) = Txwm(i) + Txom(i);
end

if i ~= N
    Txwp(i) = 2*Lamwp/(dx(i+1)/perm(i+1)+dx(i)/perm(i))/dx(i);
    Txop(i) = 2*Lamop/(dx(i+1)/perm(i+1)+dx(i)/perm(i))/dx(i);
else
    Txop(i) = 2*Lamop/dx(i)*perm(i)/dx(i);
    Txwp(i) = 2*Lamwp/dx(i)*perm(i)/dx(i);
end

%Coeficientes
Cpoo(i) = (1-Sw(i))*phi(i)*(Cr*Bo(i)+DBo(i))/dt;
Csw(i) = -phi(i)*Bo(i)/dt;
Cpow(i) = Sw(i)*phi(i)*(Cr*Bw(i)+DBw(i))/dt;
Csww(i) = phi(i)*Bw(i)/dt-Cpow(i)*DPcow(i);
%Matriz dos Coeficientes

alfa = -Csw(i)/Csww(i);
A(i) = Txom(i)+alfa*Txwm(i);
C(i) = Txop(i)+alfa*Txwp(i);
if i ~= 1
    B(i) = -(Txop(i)+Txom(i)+Cpoo(i))-
(Txwp(i)+Txwm(i)+Cpow(i))*alfa;
end
if i ~= N && i ~= 1

```

```

        D(i) = - (Cpoo(i)+alfa*Cpow(i)) *Po(i)+alfa*(Txwp(i) * (Pcow(i+1) -
Pcow(i)) +Txwm(i) * (Pcow(i-1) -Pcow(i)));
    end
    if i == 1 && IBC == 2
        D(i) = - (Cpoo(i)+alfa*Cpow(i)) *Po(i)+alfa*(Txwp(i) * (Pcow(i+1) -
Pcow(i)) +Qwi(j)/dx(i)/area);
        B(i) = - (Txop(i)+Cpoo(i)) - (Txwp(i)+Cpow(i)) *alfa;
    end
    if i == 1 && IBC == 1
        D(i) = - (Cpoo(i)+alfa*Cpow(i)) *Po(i)+alfa*(Txwp(i) * (Pcow(i+1) -
Pcow(i)) +Txwm(i) * (Pl(j)+Pcow(1)));
        B(i) = - (Txop(i)+Cpoo(i)) - (Txwp(i)+Txwm(i)+Cpow(i)) *alfa;
    end
    if i == N
        D(i) = - (Cpoo(i)+alfa*Cpow(i)) *Po(i) -
(Txop(i)+alfa*Txwp(i)) *Pr(j)+alfa*Txwm(i) * (Pcow(i-1) -Pcow(i));
    end

    t(j+1) = t(j) + dt;

end

%PRESSURE SOLUTION
%-----

Ponew = TRIDIA(N,A,B,C,D);

%SATURATION SOLUTION
%-----

for i = 1:N
    matrizSW(i,j) = Sw(i);
    if i ~= N && i ~= 1
        Sw(i) = Sw(i) + (Txop(i) * (Ponew(i+1) -Ponew(i)) +Txom(i) * (Ponew(i-
1) -Ponew(i)) -Cpoo(i) * (Ponew(i) -Po(i))) /Csw(i);
    end
    if i == N
        Sw(i) = Sw(i) + (Txop(i) * (Pr(j) -Ponew(i)) +Txom(i) * (Ponew(i-1) -
Ponew(i)) -Cpoo(i) * (Ponew(i) -Po(i))) /Csw(i);
    end
    if i == 1
        Sw(i) = Sw(i) + (Txop(i) * (Ponew(i+1) -Ponew(i)) -
Cpoo(i) * (Ponew(i) -Po(i))) /Csw(i);
    end
end

%Atualização das pressões

for i = 1:N
    matrizPO(i,j) = Po(i);
    Po(i) = Ponew(i);
end

if IBC == 2 %Computar Pl se IBC=2
    Pl(j+1) = Po(1) -Pcow(1) -Qwi(j)/dx(1)/area/Txwm(1);
    Qwi(j+1) = Qwi(j);
end

```

```

Pr(j+1) = Pr(j); %mantém valor de Pr

if IBC == 1 %Computar Qwi se IBC=1
    Qwi(j+1) = (Po(1)-Pcow(1)-Pl(j))*dx(1)*area*Txwm(1);
    Pl(j+1) = Pl(j);
end

%Computar Qop, Qwp e Wc

Qop(j+1) = -(Pr(j)-Po(N))*dx(N)*area*Txop(N);
Qwp(j+1) = -(Pr(j)-Po(N)+Pcow(N))*dx(N)*area*Txwp(N);
Wc(j+1) = Qwp(j+1)/(Qwp(j+1)+Qop(j+1));

end

t = t(1:index,1);
Qwi = Qwi(1:index,1);
Qop = Qop(1:index,1);
Qwp = Qwp(1:index,1);
Wc = Wc(1:index,1);
Pl = Pl(1:index,1);
Pr = Pr(1:index,1);

matrizPO = matrizPO(1:N,1:index);
matrizPO = matrizPO'; %Transpondo para formato da planilha
planilhaPO = [t,Pl,matrizPO,Pr]; %concatenando todas as matrizes/vetores
xlswrite('po.xls',planilhaPO,1,'A4');

matrizSW = matrizSW(1:N,1:index);
matrizSW = matrizSW'; %Transpondo para formato da planilha
planilhaSW = [t,matrizSW]; %concatenando todas as matrizes/vetores
xlswrite('sw.xls',planilhaSW,1,'A4');

planilhawells = [t,Pl,Qwi,Pr,Qop,Qwp,Wc];
xlswrite('wells.xls',planilhawells,1,'A3');

function [y,dy] = INTERP(x,ISW,N,xt,yt)
%A rotina busca na tabela e calcula por interpolação linear
%o valor da ordenada y no ponto x.
%Se o argumento x estiver fora do intervalo da tabela,
% são utilizados os pontos da extremidade.
%-----
%Dados de entrada:
%x = argumento;
%Se ISW = 0, a derivada não é calculada;
%N = número de entradas na tabela;
%xt = vetor da variável independente;
%yt = vetor da variável dependente.
%-----
%Dados de saída:
%y = valor interpolado;
%dy = derivada do valor interpolado.
%-----

if x < xt(N) %Se x for maior que o maior valor da tabela
    if x > xt(1) %Se x for menor que o menor valor da tabela
        for i = 2:N %Regra Geral
            if x >= xt(i-1) && x < xt(i)

```

```

        y = yt(i-1) + (x-xt(i-1))*(yt(i)-yt(i-1))/(xt(i)-xt(i-1));
        if ISW ~= 0
            dy = (yt(i)-yt(i-1))/(xt(i)-xt(i-1));
        end
    end
end
else
y = yt(1);
if ISW ~= 0
    dy = (yt(2)-yt(1))/(xt(2)-xt(1));
end
end
else
y = yt(N);
if ISW ~= 0
    dy = (yt(N)-yt(N-1))/(xt(N)-xt(N-1));
end
end
end

```

```

function [P] = TRIDIA(N,A,B,C,D)
%A rotina usa Eliminação de Gauss para solução
%de um sistema de equações do tipo:
%-----
%A(i)*P(i-1) + B(i)*P(i) + C(i)*P(i+1) = D(i)
%-----
%Dados de entrada:
%A(i), B(i), C(i), D(i) = matriz dos coeficientes
%P = Pressão
%N = número de equações
%-----

BB = zeros(N,1);
DD = zeros(N,1);

BB(1) = B(1);
DD(1) = D(1);

for i = 2:N
    X = A(i)/BB(i-1);
    BB(i) = B(i) - X*C(i-1);
    DD(i) = D(i) - X*DD(i-1);
end

P(N) = DD(N)/BB(N);

for k = 2:N
    i = N-k+1;
    P(i) = (DD(i) - C(i)*P(i+1))/BB(i);
end

end

```

APÊNDICE II – Codificação de fluxo bifásico óleo-água pelo MRST

```

%SIMULAÇÃO DE FLUXO BIFÁSICO ÓLEO/ÁGUA (MRST)
%-----%

clc; clear, close all;

%% DADOS DE ENTRADA

nx = 10;           % Número de divisões em X
ny = 1;           % Número de divisões em Y
Lx = 1.0*meter;   % Comprimento total em X
Ly = 1200*meter;  % Comprimento total em Y

% Propriedades da Rocha

porosidade = 0.25; % todos iguais
% porosidade = [...]; % todos diferentes (heterogeneo)
permeabilidade = 0.1*darcy; % todos iguais
% permeabilidade = [...]; % todos diferentes (heterogeneo)

% Condições Iniciais
% Saturação da Água
sw0 = 0.2;
% Pressão em atm
p0 = 200;

% passo de tempo
dtime = 2.5*second;
tmax = 400*second;
nprint = 1; % plotagem a cada 10 iterações

% Condições de Contorno
% Tipos: true = Dirichlet; false = Neuman
% [esquerda, direita]
tipoCC = [false, true];
pCC = [p0 , 200]*atm;
qCC = [25. ,25.]*centi*meter.^3/second;

% Propriedades dos Fluidos
% [água, óleo]
visco = [ 1., 0.5]*centi*poise;
dens = [ 1., 0.8].*1000*kilogram/meter^3;
n_kr = [ 2, 2]; % fator exponencial dos kr's
satr = [ .2, .2]; % saturações residuais
fac_A = [ 1, 1]; % fator multiplicador da kr

%% INICIALIZAÇÃO
% Criando um grid cartesiano
G = cartGrid([nx, ny], [Lx, Ly]);
G = computeGeometry(G);

% plotando grid
% figure()
% plotGrid(G);
% hold on
% plota vetores normais das faces das celulas
% quiver(G.faces.centroids(:,1),G.faces.centroids(:,2), ...
% G.faces.normals(:,1),G.faces.normals(:,2))
% hold off

```

```

% Criando estrutura rock
rock = makeRock(G, permeabilidade, porosidade);

% Criando estrutura do Reservatório
res = initState(G, [], p0*atm, [sw0, 1-sw0]);

% Aplicando as Condições de Contorno
% Lado Esquerdo
if (tipoCC(1))
    bc = pside ([], G, 'left', pCC(1), 'sat', [1-satr(1), satr(2)]);
else
    bc = fluxside([], G, 'left', qCC(1), 'sat', [1-satr(1), satr(2)]);
end
% Lado Direito
if (tipoCC(2))
    bc = pside (bc, G, 'right', pCC(2), 'sat', [sw0, 1-sw0]);
else
    bc = fluxside(bc, G, 'right', qCC(2), 'sat', [sw0, 1-sw0]);
end

% Adicionando módulo do MRST <incomp>
mrstModule add incomp

% Propriedades dos Fluidos
fluid = initCoreyFluid('mu' , visco, ...
                      'rho', dens, ...
                      'n' , n_kr, ...
                      'sr' , satr, ...
                      'kwm', fac_A);

% Plotando a Curva de Permeabilidade Relativa
figure()
s=linspace(satr(1),1-satr(2),20)'; kr=fluid.relperm(s);
plot(s, kr(:,1), 'b', s, kr(:,2), 'r', 'LineWidth',2);
title('Relative permeability curves')
xlabel('Saturation, s'); ylabel('Relative permeability, kr')
legend('Water', 'Oil', 'Location', 'Best')

% Solução para Equação da Pressão
T = computeTrans(G, rock);
sol = incompTPFA(res, G, T, fluid, 'bc', bc);

%% SAÍDA DAS PRESSÕES

clf;
plotCellData(G, convertTo(sol.pressure, atm()));
title('pressao inicial'), colormap('jet'),
c = colorbar; c.Label.String = 'Pressure [atm]';
axis off

t = 0; plotNo = 0; n = 0;

while t < tmax,
    sol = explicitTransport(sol, G, dtime, rock, fluid, 'bc', bc,
'computedt', false);

    % Verificação de inconsistência nas saturações
    s = sol.s(:,1);
    assert(max(s) < 1+eps && min(s) > -eps);
end

```

```

% Atualizando as pressões
sol = incompTPFA(sol, G, T, fluid, 'bc', bc);

t = t + dtime;
n = n + 1;

fprintf('Time step %3d done. Time: %10.4f, time step size:
%10.4f\n',...
    n, t,dtime);

% -----< write results >-----
if or(mod(n, nprint)== 0,t==tmax)
    plotNo = plotNo+1;

    % Plotando as Pressões e Saturações
    figure();
    heading = [num2str(t), 'segundos'];
    plotCellData(G, sol.s(:,1)); colormap('jet');
    c = colorbar; c.Label.String = 'Water Saturation [-]';
    title(heading);
    print('-dtiff', '-r110', ['saturacao@',heading, '.tiff'])

    figure();
    plotCellData(G, convertTo(sol.pressure(:,1), atm()));
    colormap('jet');
    c = colorbar; c.Label.String = 'Pressao [atm]';
    title(heading);
    print('-dtiff', '-r110', ['pressao@',heading, '.tiff'])
end
end
fprintf('\n\nNumber of iteration    : %d\n', n);
disp('End execution.');
```