

UNIVERSIDADE FEDERAL DE PERNAMBUCO

CENTRO DE TECNOLOGIA E GEOCIÊNCIAS

DEPARTAMENTO DE ENGENHARIA CIVIL

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA MECÂNICA

JOÃO PAULO RODRIGUES DE ANDRADE

**PROJETO FINAL DE INTRODUÇÃO AO MATLAB.**

RECIFE

2023

JOÃO PAULO RODRIGUES DE ANDRADE

**PROJETO FINAL DE INTRODUÇÃO AO MATLAB.**

Projeto Final da cadeira de Introdução ao Matlab, ministrada pela professora Silvana Maria Bastos, do Programa de Pós-Graduação em Engenharia Civil da Universidade Federal de Pernambuco, Centro de Tecnologia e Geociências, como requisito parcial para a nota final.

RECIFE

2023

Sumário

[1 Introdução 6](#__RefHeading___Toc3817_527847892)

[2 Objetivos 6](#__RefHeading___Toc3819_527847892)

[2.1 Geral 6](#__RefHeading___Toc3821_527847892)

[2.2 Específicos 6](#__RefHeading___Toc3823_527847892)

[3 Metodologia 6](#__RefHeading___Toc3825_527847892)

[3.1 Escoamento monofásico 6](#__RefHeading___Toc3827_527847892)

[3.2 Escoamento bifásico (água-óleo) 7](#__RefHeading___Toc3829_527847892)

[3.2.1 Equação da saturação 7](#__RefHeading___Toc3831_527847892)

[3.2.2 Equação da pressão 8](#__RefHeading___Toc3833_527847892)

[3.2.3 Método *Upwind* de primeira ordem 9](#__RefHeading___Toc3835_527847892)

[3.2.4 Condições iniciais e de contorno 9](#__RefHeading___Toc3837_527847892)

[3.2.5 Método de solução sequencial implícito 10](#__RefHeading___Toc3839_527847892)

[3.3 Etapa de pré-processamento 11](#__RefHeading___Toc3841_527847892)

[3.4 Etapa de pós-processamento 11](#__RefHeading___Toc3843_527847892)

[4 Fluxograma do código 12](#__RefHeading___Toc3845_527847892)

[5 Resultados 14](#__RefHeading___Toc3847_527847892)

[5.1 Problema 1 – Escoamento bifásico 1D em meio heterogêneo 14](#__RefHeading___Toc3849_527847892)

[5.2 Problema 2 – Escoamento 2D em meio homogêneo 19](#__RefHeading___Toc3851_527847892)

[5.3 Problema 3 - Escoamento bidimensional em meio heterogêneo. 24](#__RefHeading___Toc3853_527847892)

[6 Conclusões 29](#__RefHeading___Toc7407_527847892)

[Referências 30](#__RefHeading___Toc3855_527847892)

# 1 **Introdução**

A simulação numérica de fluxo em reservatórios de petróleo tem o objetivo de prever o comportamento do reservatório quando submetido a condições impostas na superfície (como vazão e pressão dos poços produdores e injetores) bem como aquelas que são advindas no próprio reservatório (como as condições de pressão e saturação iniciais, existência de aquíferos ou capas de gás). Após a aquisição dos dados da rocha reservatório (ou simplesmente reservatório como mencionado antes), são obtidos vários cenários de produção por meio da simulação computacional. O objetivo da simulação é prever o melhor cenário, otimizando o lucro, os custos e, não menos importante, avaliar os impactos ambientais do processo de retirada dos hidrocarbonetos.

Este trabalho tem o objetivo de aplicar os conhecimentos obtidos em sala de aula de programação na linguagem Matlab no desenvolvimento de um programa para simular o escoamento bifásico (água-óleo) em reservatórios de petróleo, utilizando o método segregado sequencial implícito para o cálculo dos campos de pressão e saturação.

# 2 Objetivos

## 2.1 Geral

Desenvolver um simulador bifásico água-óleo, utilizando o método sequencial implícito para resolver a pressão e a saturação na linguagem de programação Matlab.

## 2.2 Específicos

* Estudar o escoamento bifásico e o método de solução sequencial implícito;
* Explanar a metodologia aplicada no simulador;
* Explanar o fluxograma do código de simulação;
* Explicitar a função da rotina principal e das subrotinas;
* Apresentação dos resultados em figuras, tabelas ou gráficos como curvas de produção acumulada de óleo, razão água-óleo de produção, valor dos passos de tempo e quantidade de iterações no cálculo da saturação;

# 3 Metodologia

## 3.1 Escoamento monofásico

A equação da conservação da massa para o escoamento monofásico em reservatórios de petróleo é dada por:

, (Equação 1)



onde , , , , e são a massa específica (Kg/m3), porosidade (adimensional), velocidade (m/s), viscosidade (Pa s), termo fonte ou sumidouro (Kg/(m3/s)) e tempo (s). A porosidade é definida como a porção do espaço ocupada pelo fluido, que é dada por:



, (Equação 2)



onde é o volume do poro e é o volume ocupado pela rocha e pelo poro. A velocidade do fluido é dada pela lei de Darcy para o escoamento de fluidos em meios porosos, dada por (desprezando o efeito da gravidade):



, (Equação 3)



onde (m2) é a permeabilidade da rocha, definida como a facilidade com que o fluido a atravessa e a pressão do fluido. Considerando que o fluido é incompressível, dividindo a equação 1 por , obtemos:



(Equação 4)



onde . O método utilizado nesse trabalho para resolver a equação 4 é o método dos volumes finitos, que é obtido ao integrar esta equação no volume , resultando em:



, (Equação 5)



onde podemos aplicar o teorema da divergência de Gauss na segunda integral da equação 5 para obter:

, (Equação 6)



sendo a superfície de contorno de e , onde é o versor que aponta para fora de , a face que pertence a , e é a área da face . Considerando a rocha incompressível, a equação 4 se torna:



. (Equação 7)



## 3.2 Escoamento bifásico (água-óleo)

### 3.2.1 Equação da saturação

Para o escoamento bifásico utiliza-se a equação 4 para uma das fases (água ou óleo), resultando na equação da saturação (CONTRERAS 2012) :



, (Equação 8)



sendo a saturação da fase , definida como a razão entre o volume da fase e o volume do poro . A velocidade da fase, desprezando efeitos de pressão capilar, é dada por:



, (Equação 9)



sendo

, (Equação 10)



onde e são a mobilidade e a permeabilidade relativa da fase . O modelo de permeabilidade relativa utilizado nesse trabalho é o de Brooks e Corey, dada por:



, (Equação 11)



sendo

, (Equação 12)



onde são os valores máximos da permeabilidade relativa de cada fase, são números determinados experimentalmente de acordo com as curvas de permeabilidade relativa de cada fase, é a saturação irredutível de água e , sendo a saturação residual de óleo. Aplicando o teorema da divergência de Gauss na equação 8 chega-se a:



. (Equação 13)



Discretizando a equação 13 no tempo, chegamos a equação da saturação na forma discreta:

, (Equação 14)



onde é o passo de tempo.



### 3.2.2 Equação da pressão

Assumindo que o meio está totalmente saturado pelas fases água e óleo, ou seja

, (Equação 15)



somando as equações da saturação de cada fase (equação 8) e realizando algumas manipulações algébricas, obtemos a equação da pressão:

, (Equação 16)



sendo

, (Equação 17)



e , de modo que podemos reescrever a equação da velocidade de cada fase como:



. (Equação 18)



Integrando a equação 16 no volume e aplicando o teorema da divergência de Gauss, obtemos a equação da pressão na forma discreta:

(Equação 19)



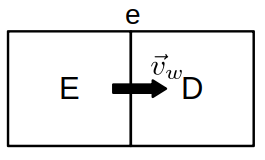
### 3.2.3 Método *Upwind* de primeira ordem

Para aproximar a saturação de água na face, nesse trabalho é utilizado o método *upwind* de primeira ordem descrito na equação 20, onde os volumes à esquerda e à direita da face estão representados na Figura 1.



(Equação 20)



Figura 1: Representação de dois volumes adjacentes.

### 3.2.4 Condições iniciais e de contorno

Para que se possa obter a solução é preciso definir as condições iniciais e de contorno. As condições de contorno são a de *Dirichlet,* onde a pressão é definida:

, (Equação 21)



e a de *Neumman*, onde o fluxo é prescrito:

. (Equação 22)



As condições iniciais são impostas no início da simulação, como a saturação inicial da fase água () dos volumes da malha computacional.

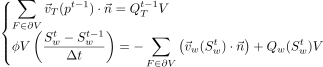


### 3.2.5 Método de solução sequencial implícito

As variáveis escolhidas para obter a solução nesse trabalho são a pressão () e a saturação da fase água (), onde a saturação da fase óleo é obtida pela equação de restrição 15. As equações a serem resolvidas são (CAVALCANTE 2019):



, (Equação 23)



onde o passo de tempo () é calculado pela condição de estabilidade de *Courant-Friedrichs-Lewy* (CFL), dada por (CONTRERAS 2012):



, (Equação 24)



sendo a distância entre os centroides entre dois volumes adjacentes a face onde está sendo avaliada a velocidade . A pressão e a saturação são resolvidas implicitamente de forma sequencial:



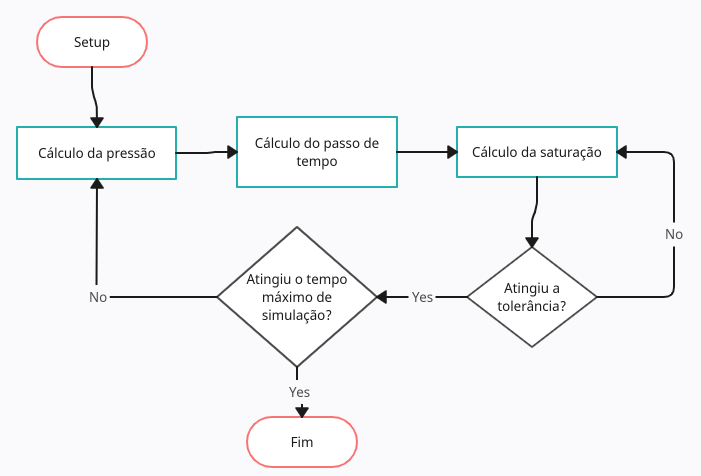
1. primeiro a pressão é calculada implicitamente no instante ;



1. é estabelecido o passo de tempo de acordo com a equação 24;
2. em seguida a saturação é calculada implicitamente no instante .



A Figura 2 mostra o fluxograma do método sequencial implícito.

Figura 2: Fluxograma do método sequencial implícito.

Nesse trabalho está sendo utilizado o método direto para o cálculo da pressão e o método de Newton Raphson para resolver a equação da saturação. A matriz jacobiana, é obtida por meio da biblioteca myAD, que utiliza o método de diferenciação automática (MARTIN 2023). Uma vez que o resíduo é estabelecido, a matriz jacobiana é obtida facilmente por meio de uma função interna da biblioteca. O resíduo é definido como:

, (Equação 25)



e a matriz jacobiana é obtida derivando o resíduo em relação a saturação de cada volume da malha computacional:

, (Equação 26)



de modo que a saturação no passo é definido como:



. (Equação 27)



A condição de parada do método de Newton Raphson acontece quando a norma do erro entre as saturações de dois passos implícitos consecutivos ( e ) é menor que um valor de tolerância definido pelo usuário . Quando a tolerância é atingida temos: . Ainda dentro do *loop* de Newton Raphson, se a quantidade de iterações ou o valor do erro entre dois passos de tempo consecutivos ultrapassar o valor definido pelo usuário, o *loop* é reiniciado com .



## 3.3 Etapa de pré-processamento

O pré-processamento foi realizado na linguagem de programação *python*, onde foram realizados os seguintes passos:

* geração da malha computacional, leitura e obtenção das propriedades da malha (geometria e estrutura de dados) com o uso da biblioteca pymoab;
* definição dos campos permeabilidade e porosidade;
* definição das condições de contorno e iniciais;
* exportação dessas informações por meio da biblioteca *scipy*, que permite escrita e leitura de dados no formato .mat.
* leitura dos arquivos gerados em python no matlab

## 3.4 Etapa de pós-processamento

Após a obtenção dos resultados advindos da simulação realizada no Matlab, os dados são exportados para arquivos .mat. Esses arquivos são pós-processados em python para gerar arquivos do tipo .vtk, que serão utilizados no software Visit. No Matlab, os dados são usados para gerar gráficos de curvas de produção, razão água-óleo, passos de tempo e quantidade de iterações no cálculo implícito da saturação para cada vpi. O vpi, explicitado na equação 28, é uma maneira de mensurar o tempo de simulação.

(Equação 28)



# 4 Fluxograma do código

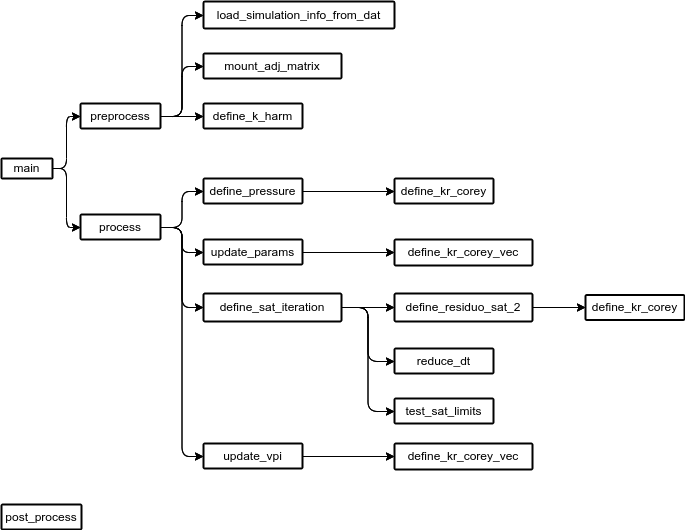
A seguir serão explanados os códigos utilizados e suas finalidades:

* main (rotina principal): executa as rotinas preprocess e process;
* load\_simulation\_info\_from\_dat: executa a leitura dos dados definidos pelo usuário no arquivo simulation\_info.dat;
* define\_k\_harm: calcula a média harmônica da permeabilidade nas faces;
* mount\_adj\_matrix: monta a matriz de adjacências entre faces e volumes;
* preprocess: executa as rotinas mount\_adj\_matrix, load\_simulation\_info\_from dat e define\_k\_harm;
* process: realiza a simulação;
* define\_pressure: realiza o cálculo da pressão;
* define\_sat\_iteration: realiza as iterações de Newton Raphson na variável da saturação;
* define\_residuo\_sat\_2: monta o resíduo em cada passo ;



* reduce\_dt: reduz o passo de tempo conforme os critérios de convergência e as informações fornecidas pelo usuário;
* test\_sat\_limits: verifica se a saturação está dentro dos limites corretos;
* define\_kr\_corey: retorna os valores de permeabilidade relativa de água e óleo;
* define\_kr\_corey\_vec: faz o mesmo que define\_kr\_corey de forma vetorizada;
* update\_params: define o passo de tempo e a direção *upwind*;
* update\_vpi: calcula o vpi, o volume de óleo produzido e a razão água-óleo de produção.
* post\_process: realiza a leitura dos arquivos gerados pela simulação para gerar os gráficos de produção acumulada de óleo, razão água-óleo, passos de tempo e quantidade de iterações no cálculo implícito da saturação.

Na pasta nomeada de ‘dados’ estão os dados que foram pré-processados e também os que serão exportados ao fim da simulação. A Figura 3 mostra como os códigos estão organizados de acordo com o nível de execução de cada função e sua ordem de utilização.

Figura 3: Fluxograma das rotinas.

# 5 Resultados

## 5.1 Problema 1 – Escoamento bifásico 1D em meio heterogêneo

Este problema consiste de um escoamento unidimensional bifásico em prescrição de pressão de valor 100 no volume da esquerda e de 0 no da direita com o objetivo de avaliar a solução dos campos de pressão e saturação num problema simples. A saturação é prescrita no valor de 1 no volume da esquerda. A malha computacional utilizada consiste em 20 volumes, com cada volume de dimensões . O campo de permeabilidade é ilustrado na Figura 4 e os campos de saturação e pressão inicial nas Figuras 5 e 6. A porosidade é homogênea no valor de 0.2. Segue abaixo os valores dos dados que são definidos pelo usuário:



* ;



* ;



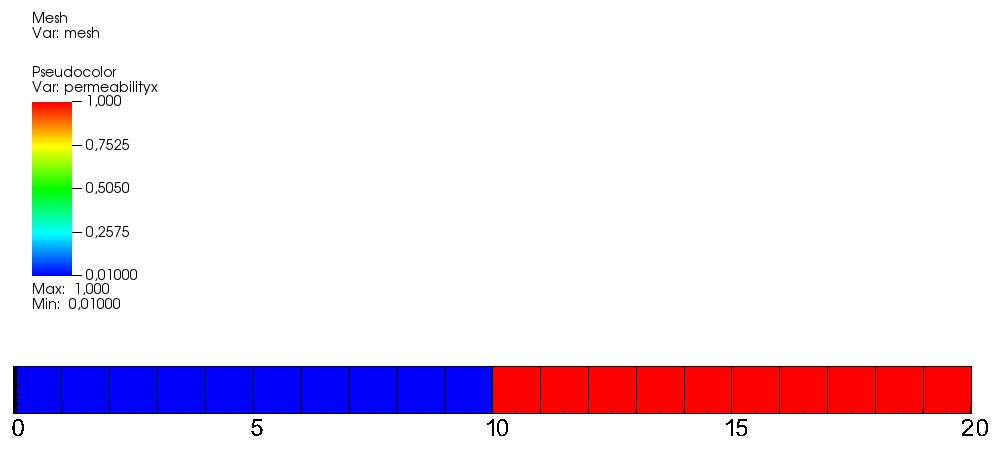
* ;

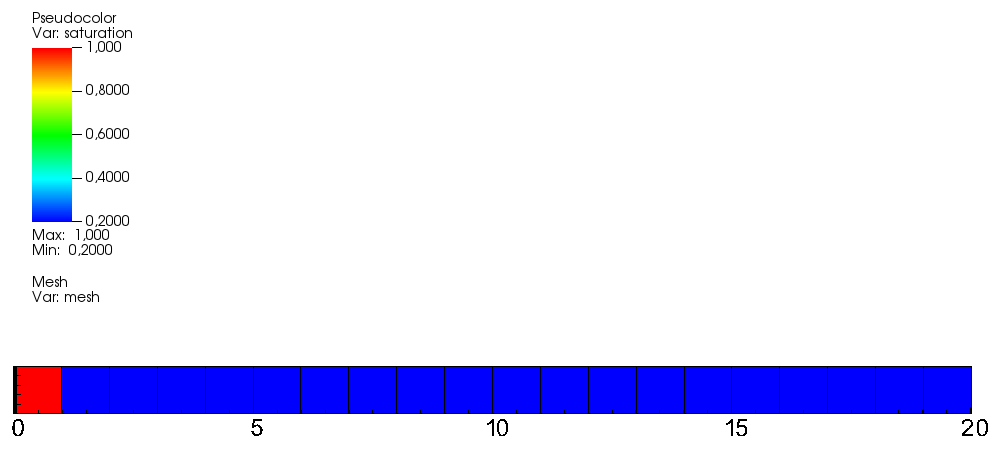


* número máximo de iterações no cálculo da saturação = 10000;
* valor máximo de = 100 (entre duas iterações consecutivas);

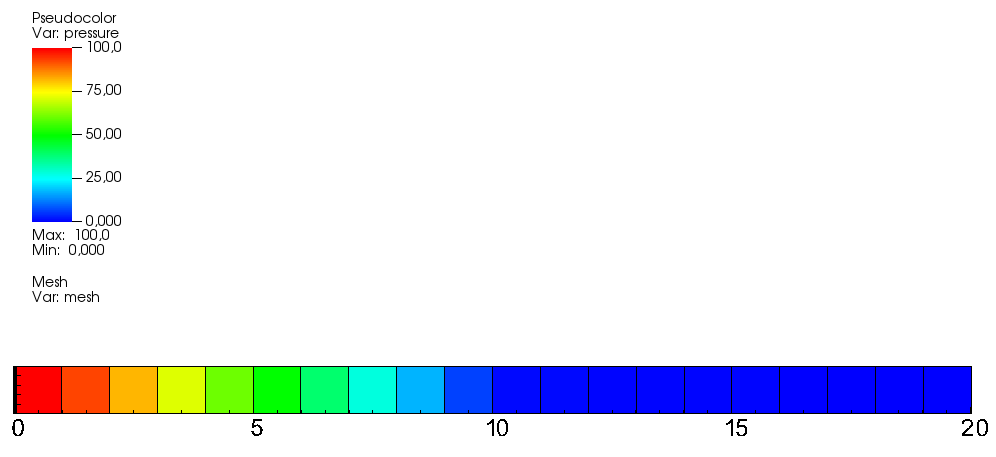


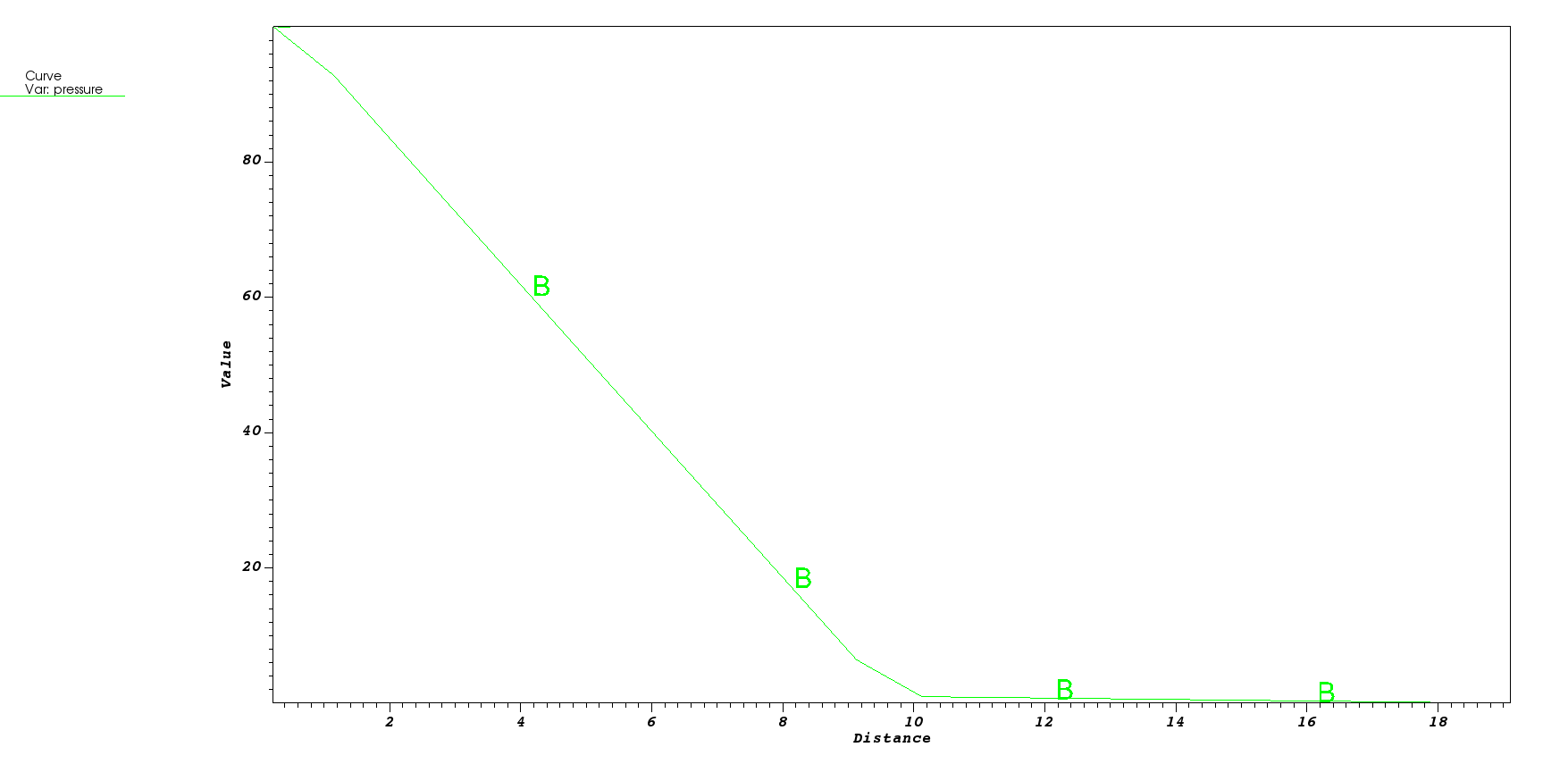
* valor máximo de *loops* para reduzir o passo de tempo: 20;
* CFL = 1;
* vpi máximo: 0.75

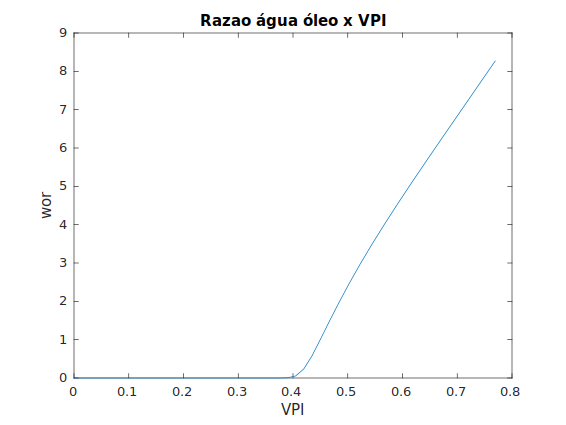
Figura 4: Campo de permeabilidade.

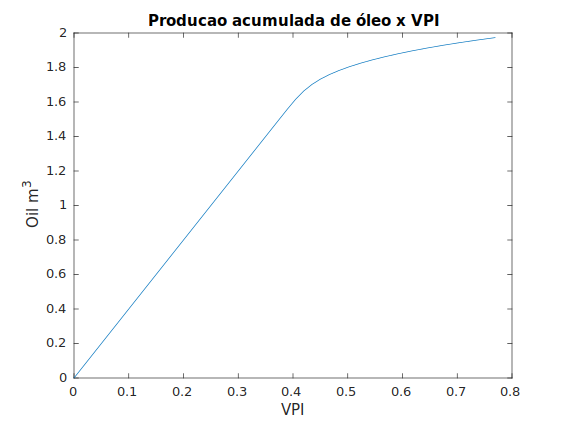
Figura 5: Campo de saturação inicial.

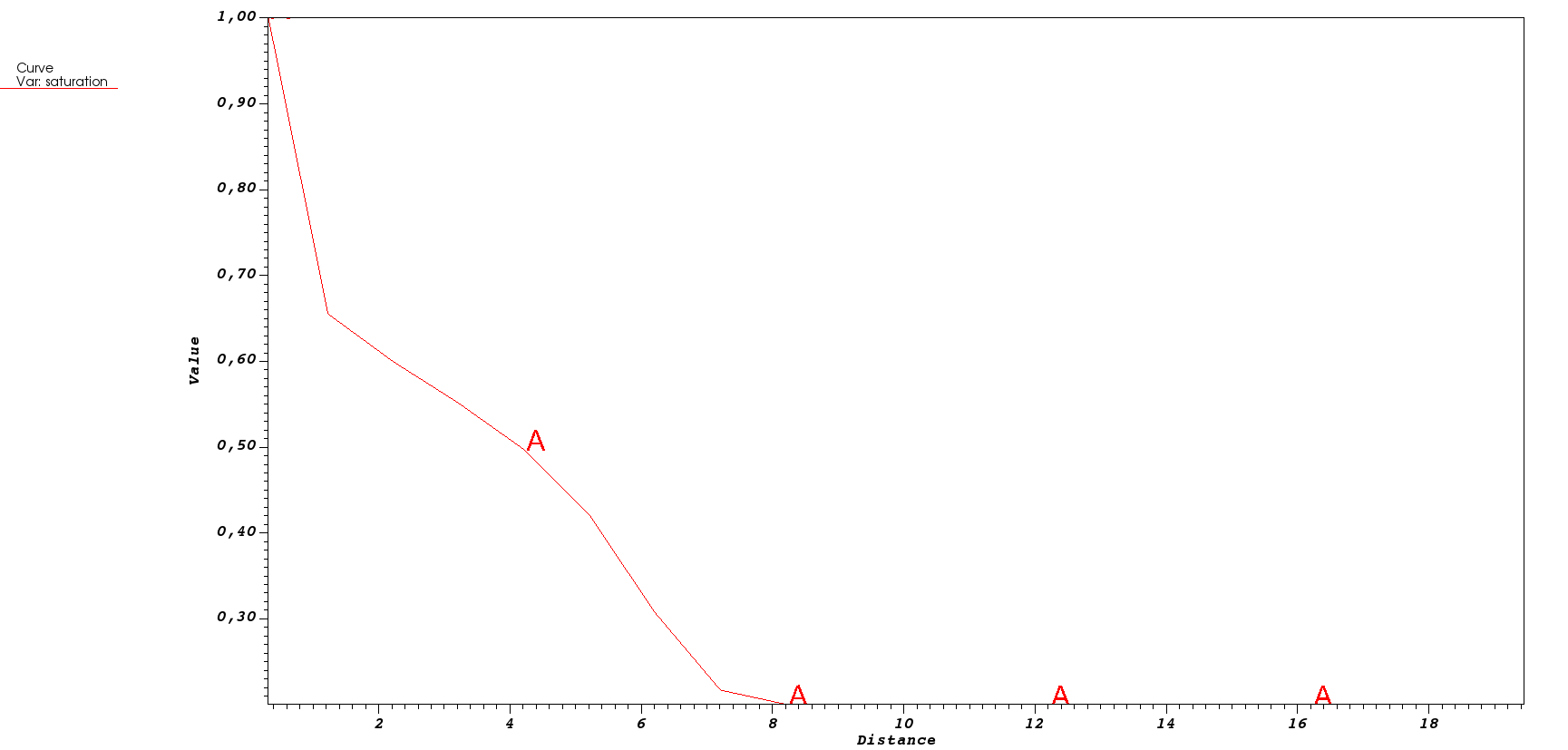
A Figura 7 mostra o gráfico do campo de pressão inicial, ilustrando o gradiente de pressão entre as duas regiões de diferentes permeabilidades. As Figuras 8 e 9 mostram a razão água-óleo e a produção acumulada de óleo ao longo da simulação. As Figuras 10 a 13 ilustram o campo de saturação para diferentes vpi’s.

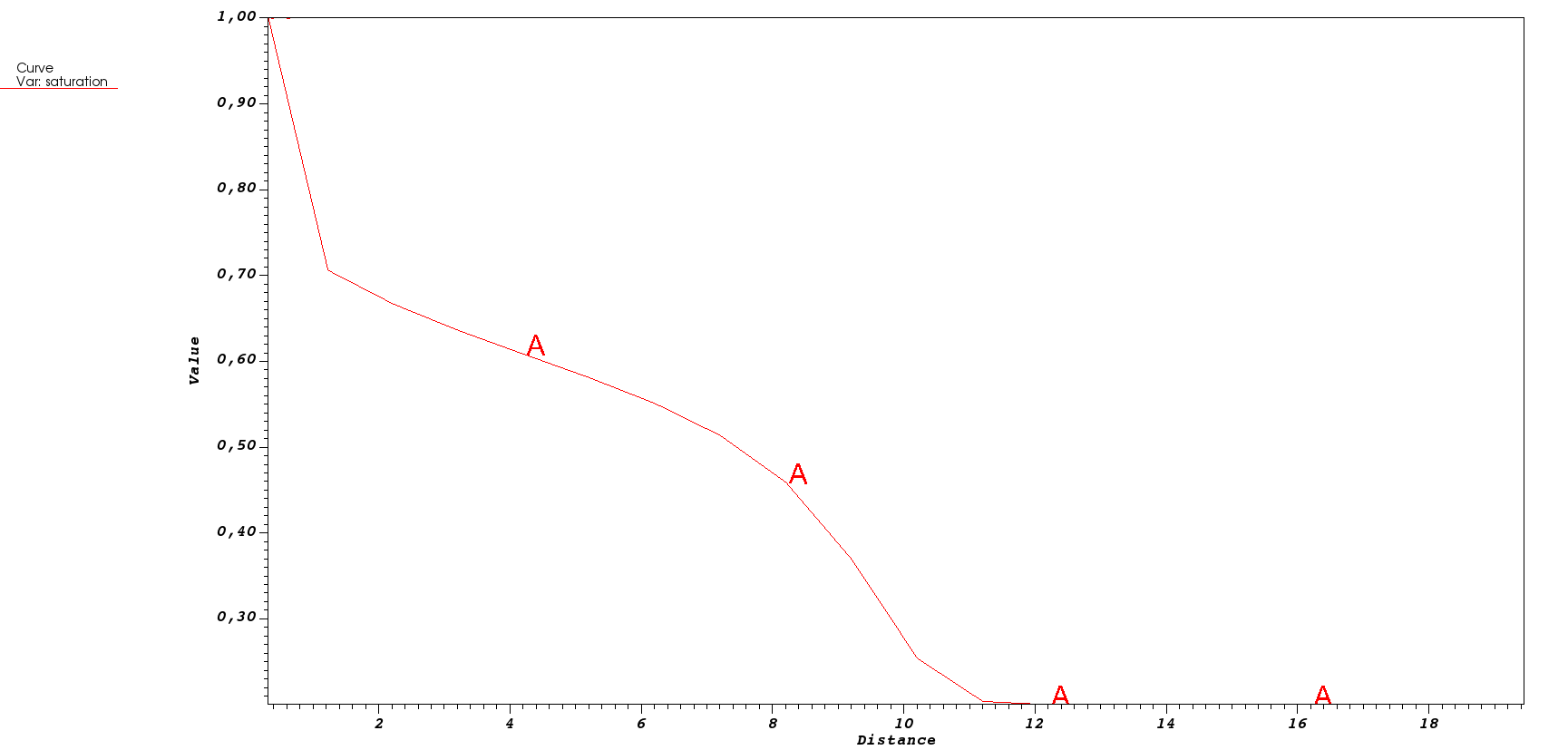
Figura 6: Campo de pressão inicial.

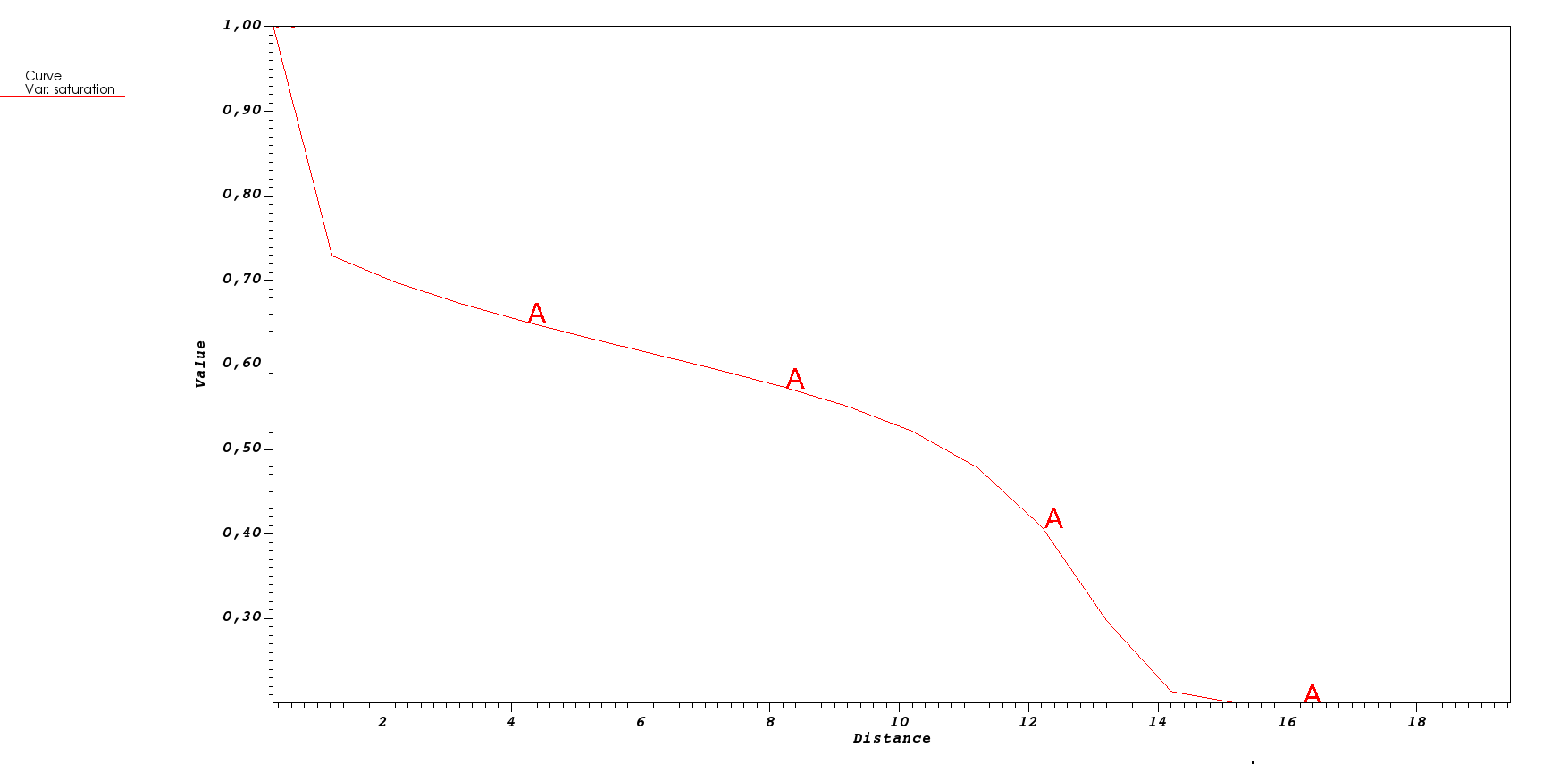
Figura 7: Gráfico do campo de pressão inicial.

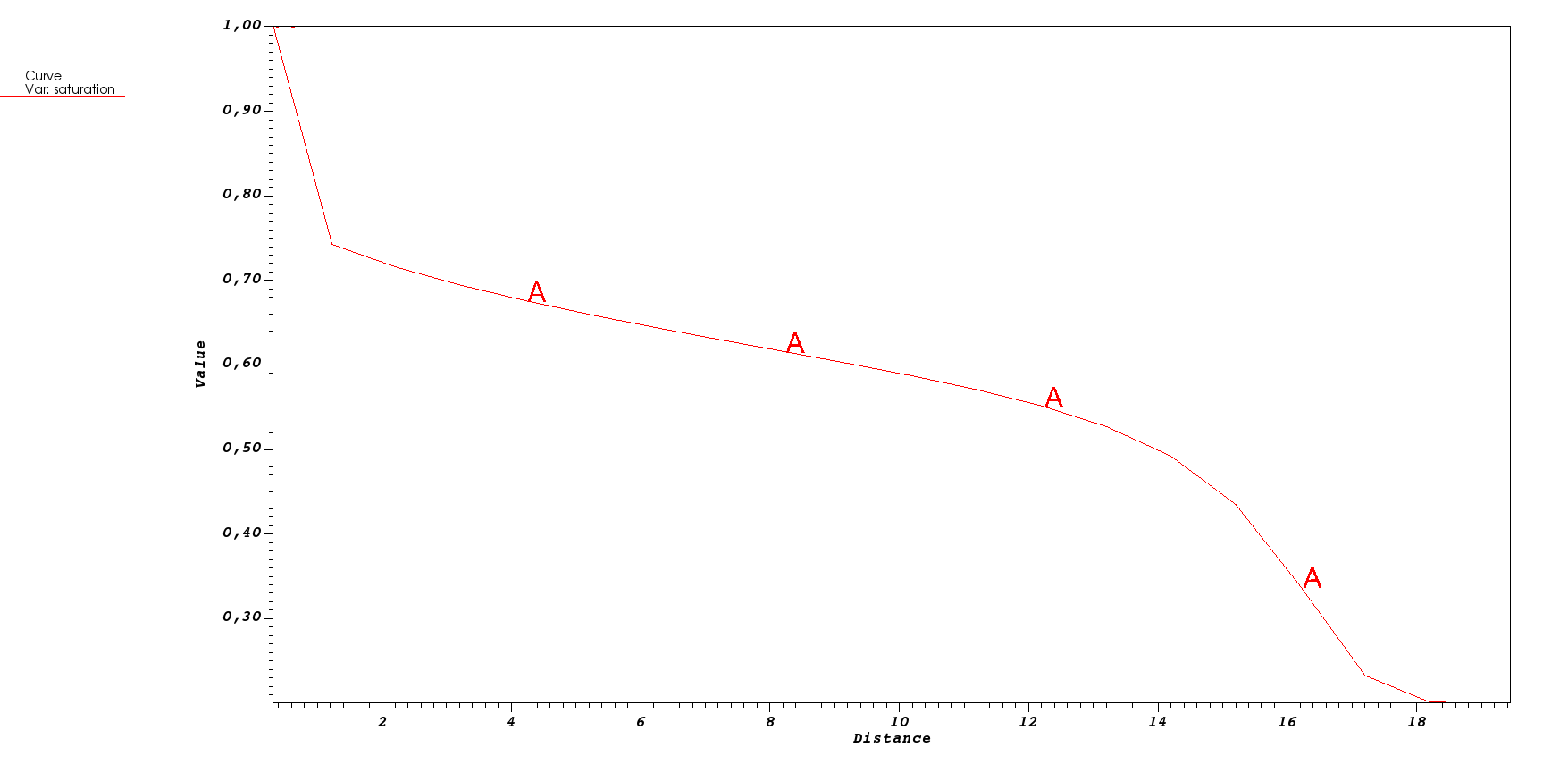
Figura 8: Razão água-óleo

Figura 9: Produção acumulada de óleo

Figura 10: Saturação no vpi = 0.08.

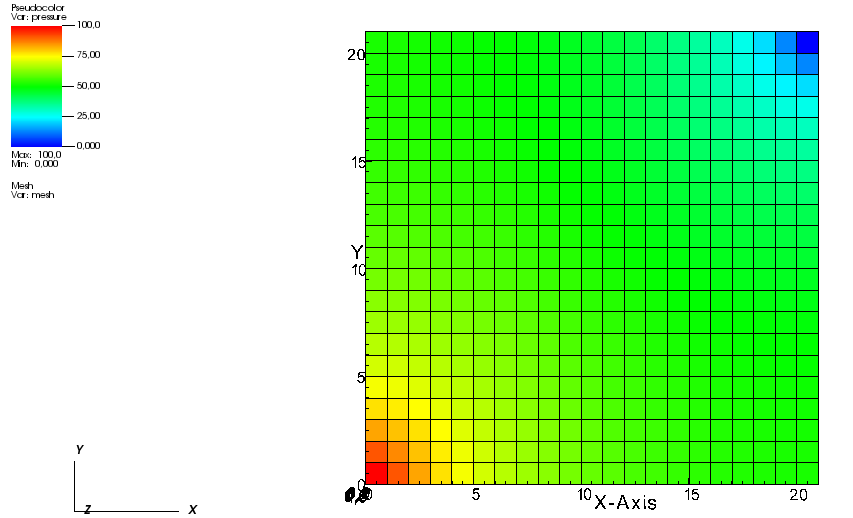
Figura 11: Saturação no vpi = 0.16.

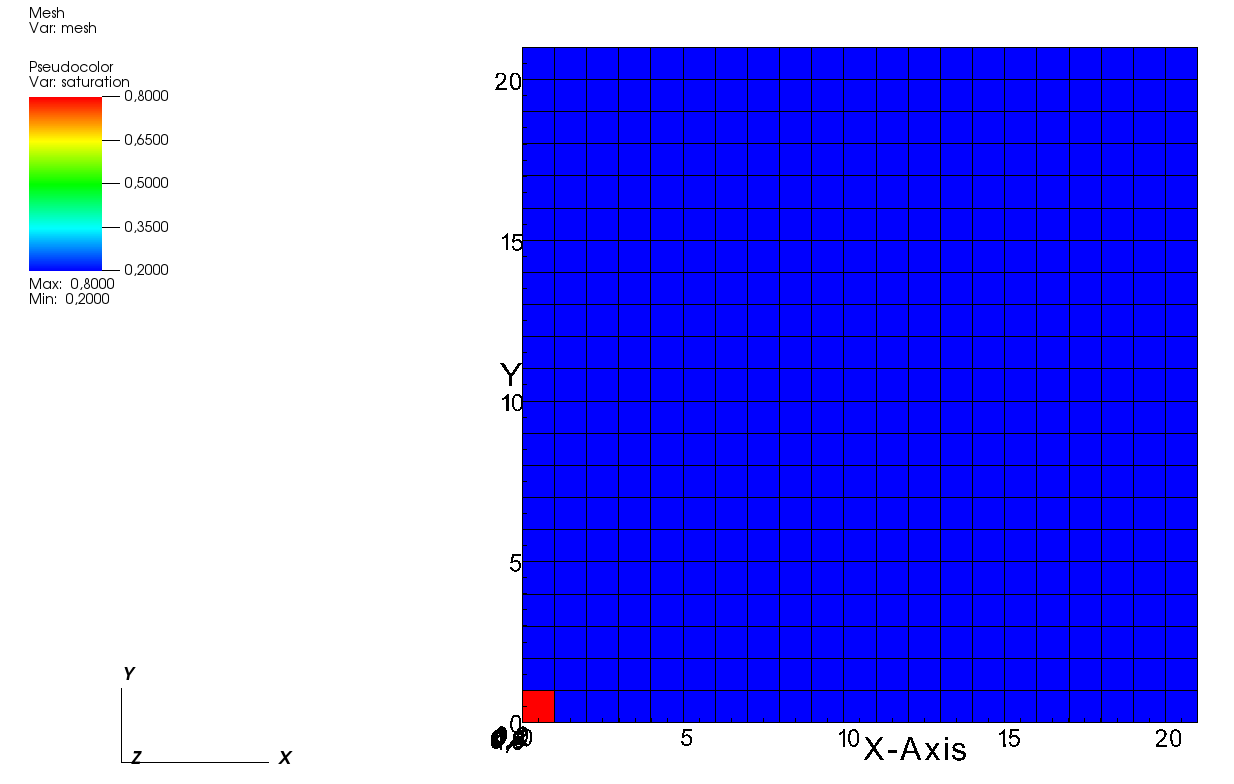
Figura 12: Saturação no vpi = 0.24.

Figura 13: Saturação no vpi = 0.31.

## 5.2 Problema 2 – Escoamento 2D em meio homogêneo

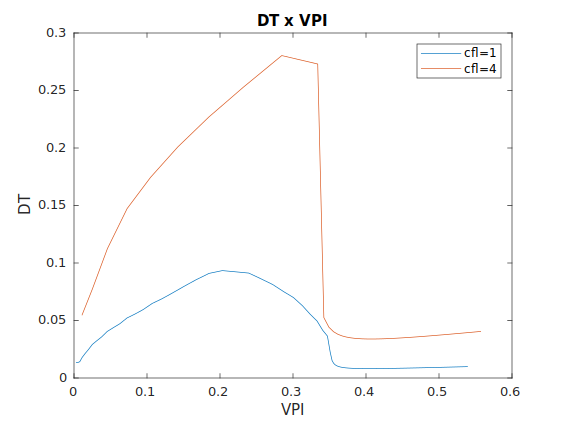
Este problema consiste num escoamento bidimensional de um quarto de *five spot* com campo de permeabilidade homogêneo no valor de 1. As pressões prescritas são no valor de 100 e 0. O campo de pressão e saturação inicial estão ilustrados nas Figuras 14 e 15. Os valores defininidos pelo usuário são os mesmos do problema 1 com a adição do critério de parada de razão máxima de água-óleo no valor de 3. Esse problema foi simulado duas vezes, sendo uma simulação com CFL = 1 e outra com CFL = 4.

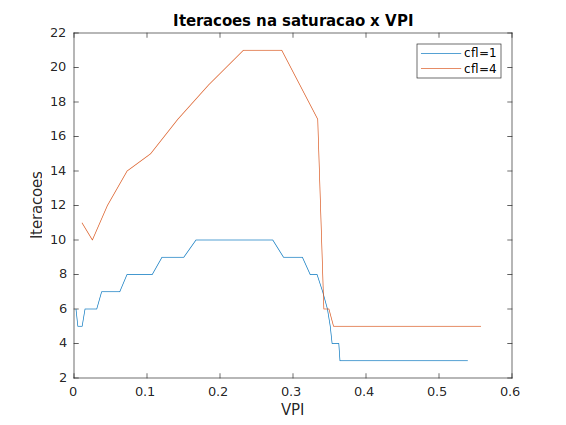
Figura 14: Campo de pressão inicial.

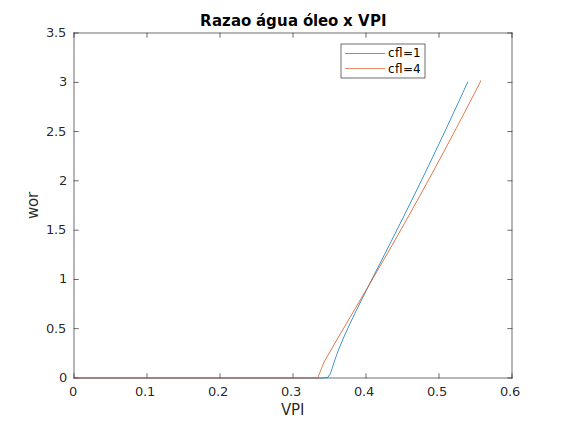
Figura 15: Campo de saturação inicial

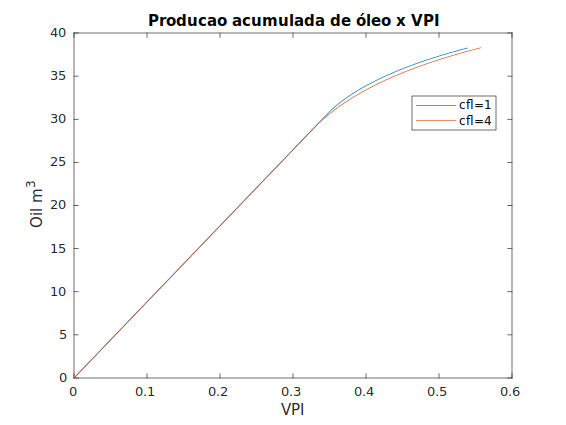
As Figuras 16 a 19 mostram os passos de tempo, a quantidade de iterações no cálculo da saturação, a razão água óleo e a produção acumulada de óleo para os valores de CFL 1 e 4.

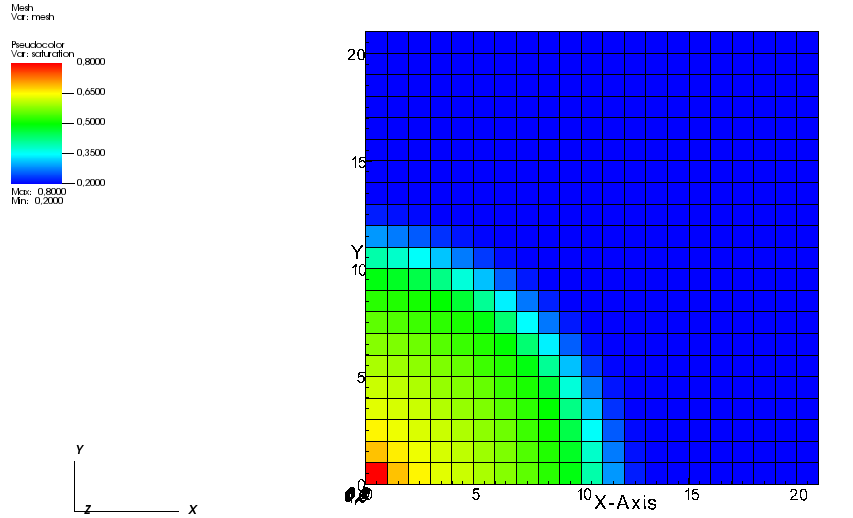
As Figuras 20 a 22 mostram o campo de saturação para diferentes instantes na simulação com CFL = 4.

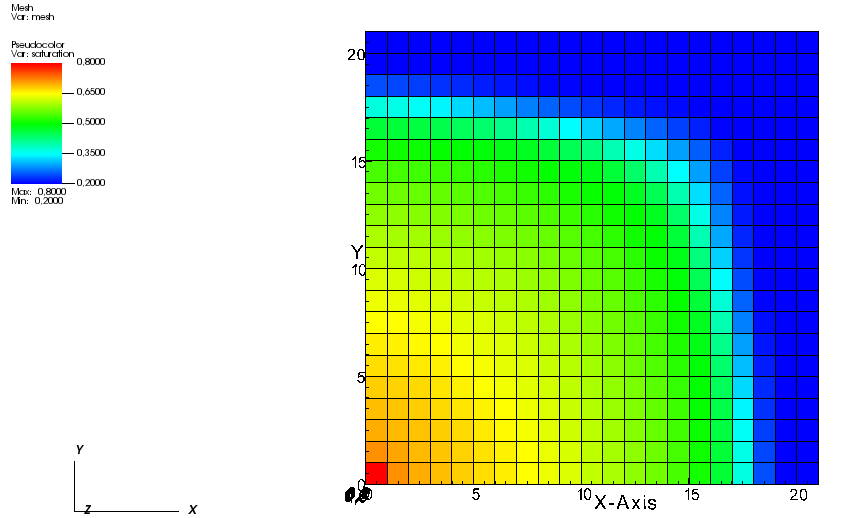
Figura 16: Passos de tempo.

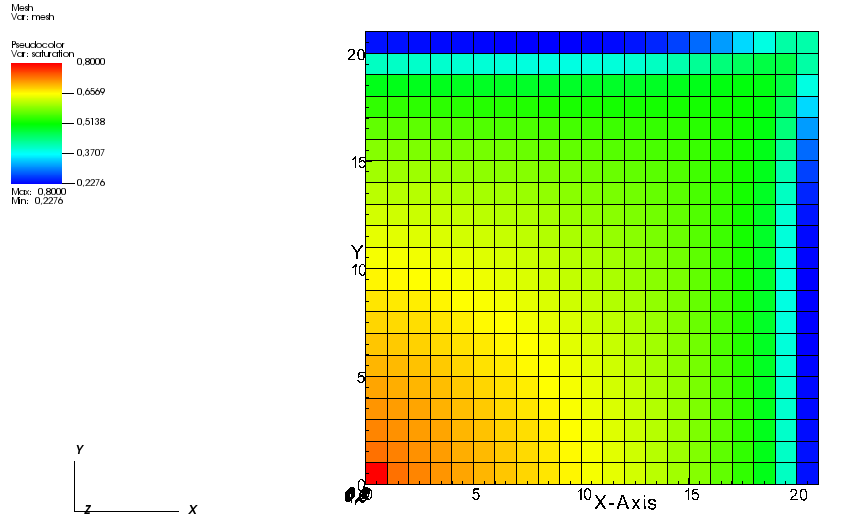
Figura 17: Quantidade de iterações no cálculo da saturação.

Figura 18: Razão água-óleo.

Figura 19: Produção acumulada de óleo.

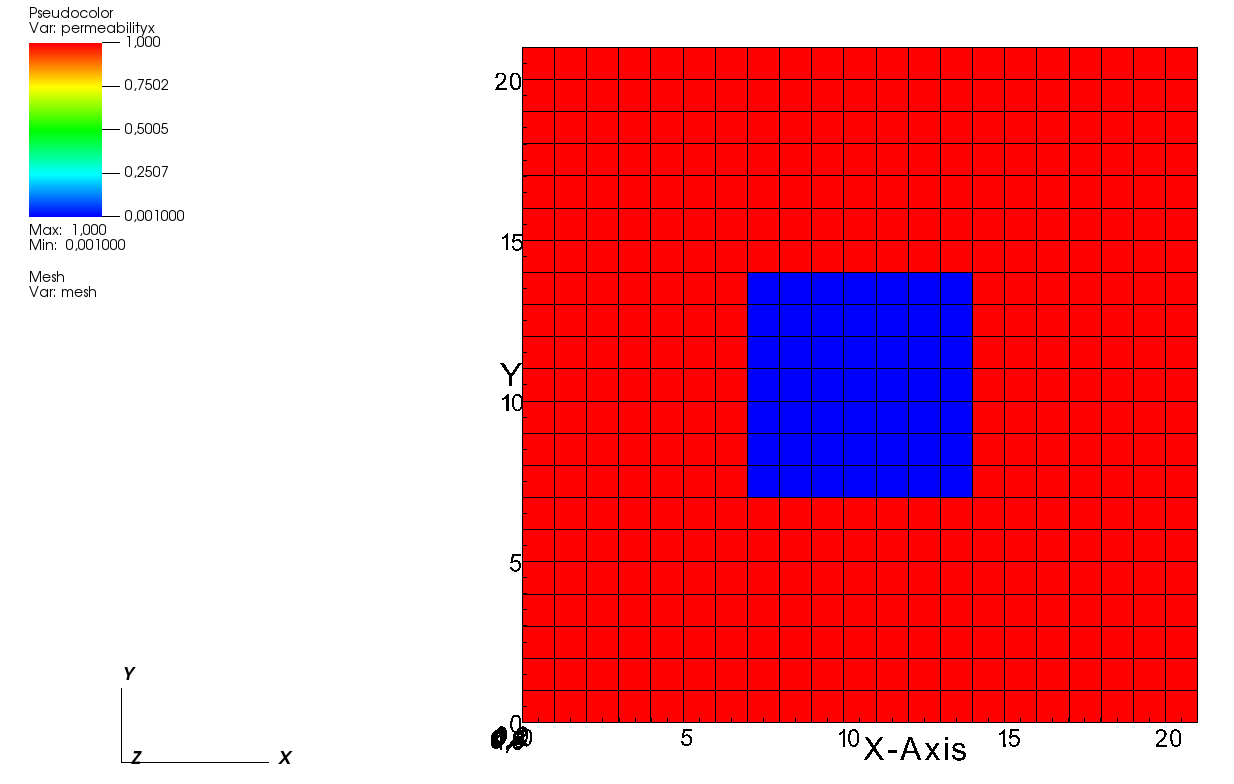
Figura 20: Saturação no vpi = 0.07.

Figura 21: Saturação no vpi = 0.23.

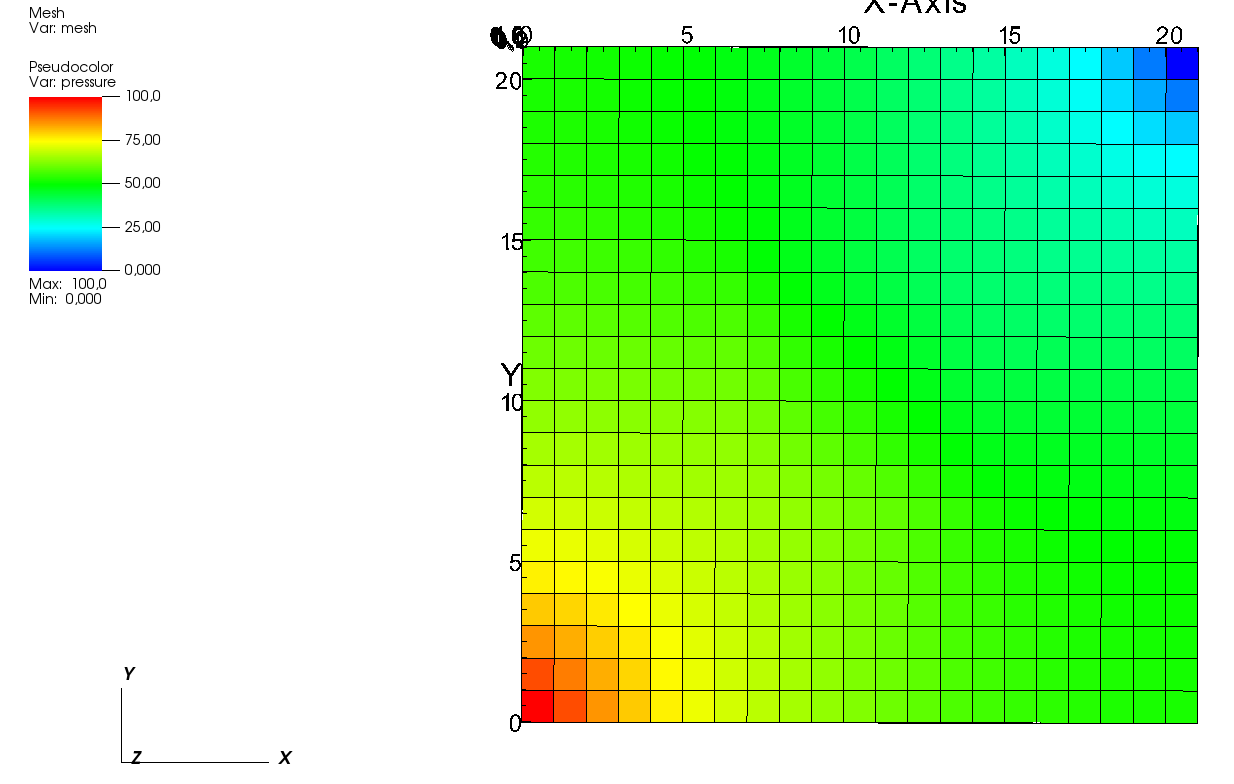
Figura 22: Saturação no vpi = 0.36.

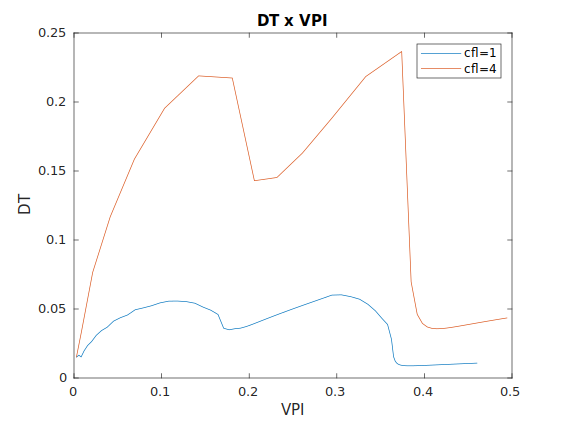
## 5.3 Problema 3 - Escoamento bidimensional em meio heterogêneo.

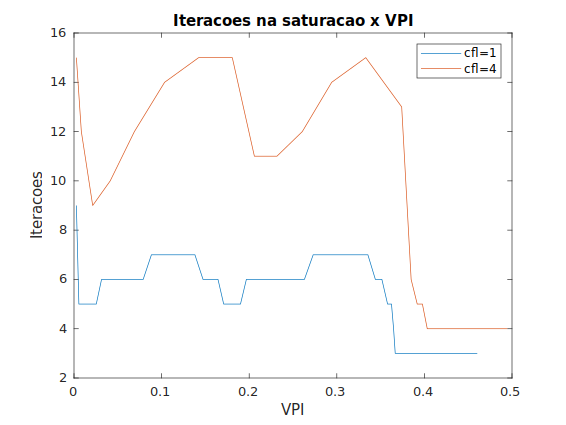
Esse problema consiste no escoamento bidimensional de um quarto de *five spot* num meio heterogêneo. O campo de permeabilidade e de pressão inicial são ilustrados nas Figuras 23 e 24 .As condições de parada, de contorno e iniciais são iguais as do Problema 2.

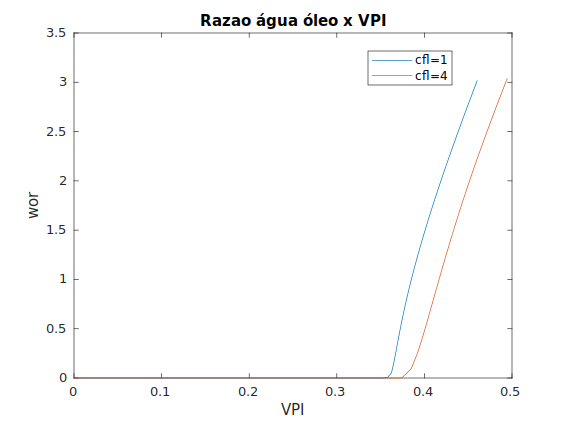
Figura 23: Campo de permeabilidade.

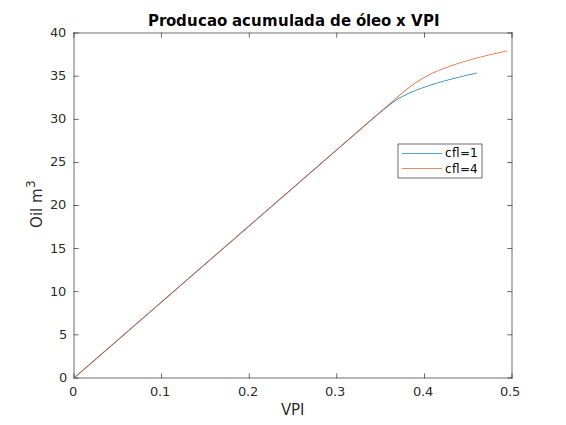
As Figuras 25 a 28 mostram os passos de tempo, quantidade de iterações no cálculo da saturação, razão água-óleo e produção acumulada de óleo ao longo do tempo com o CFL de 1 e 4. As Figuras ilustram o campo de saturação em diferentes instantes na simulação com CFL = 4.

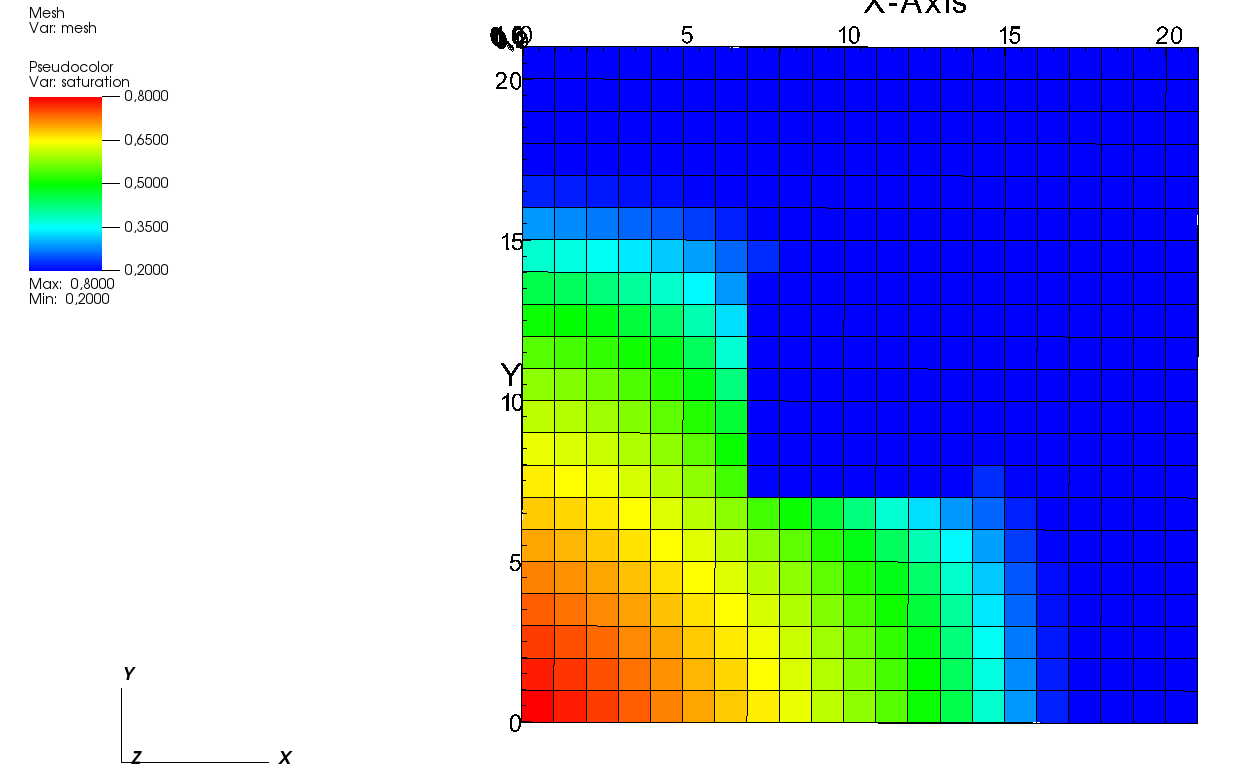
Figura 24: Campo de pressão inicial.

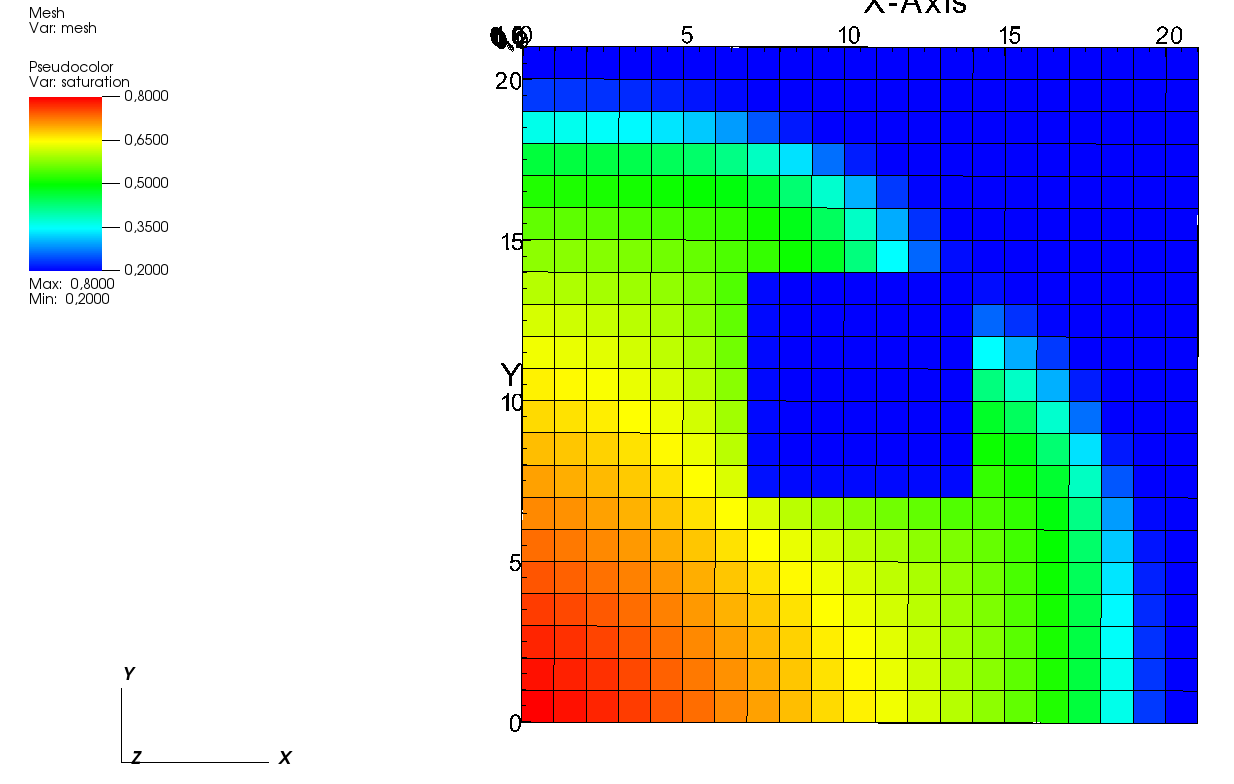
Figura 25: Passos de tempo.

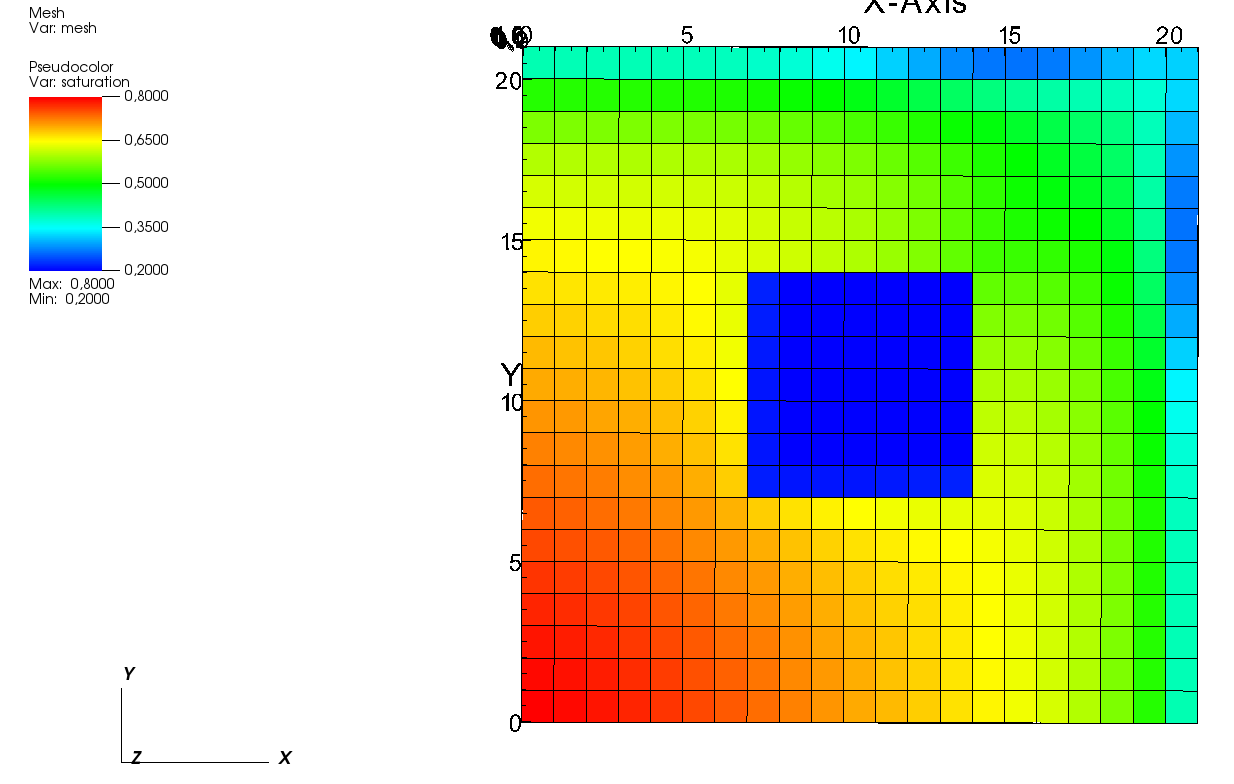
Figura 26: Quantidade de iterações no cálculo da saturação.

Figura 27: Razão água-óleo

Figura 28: Produção acumulada de óleo.

Figura 29: Saturação no vpi = 0.13.

Figura 30: Saturação no vpi = 0.23.

Figura 31: Saturação no vpi = 0.37.

# 6 Conclusões

Este trabalho tem o objetivo de utilizar a linguagem Matlab para implementar um programa capaz de simular o escoamento bifásico água-óleo em reservatórios de petróleo utilizando o método seuqencial implícito para calcular os campos de pressão e saturação e comparar os resultados com . No Problema 1, é possível notar a diferença do gradiente de pressão nas regiões com diferentes campos de permeabilidade, como era de se esperar. Nos problemas 2 e 3 nota-se que a quantidade de iterações com CFL = 4 é maior que com o CFL = 1. Isso acontece devido a distância no tempo entre as soluções: quanto mais distante, mais iterações são necessárias para atingir a convergência. No gráfico da razão água-óleo do Problema 2, observa-se que as soluções estão próximas, diferentemente do Problema 3, onde a curva com CFL = 4 atrasou o *water cut* (início da produção de água)*,* gerando um cenário otimista em relação à simulação com CFL = 1.

# Referências

CONTRERAS, F. R. L. Um Método de Volumes Finitos Centrado na Célula para a Simulação de Escoamentos Bifásicos em Reservatórios de Petróleo Heterogêneos e Anisotrópicos. Universidade Federal de Pernambuco. Recife. Dissertação (Mestrado). 2012.

CAVALCANTE, T. M. A finite volume scheme coupled with a hybrid-grid method for the 2-D simulation of two-phase flows in naturally fractured reservoirs. Universidade Federal de Pernambuco. Recife. Dissertação (Mestrado). 2019.

MARTIN F. Automatic Differentiation for Matlab (https://www.mathworks.com/matlabcentral/fileexchange/15235-automatic-differentiation-for-matlab). Acessado em 8 de Maio de 2023.